Cairo University
Faculty of Computers and Artificial Intelligence
Department of Bioinformatics

Gene Selection for cancer

Supervised by

Prof. Amr Badr
Dr. Sabah Sayed
TA. Nora Adbelhameed

Implemented by

20168001	Amira Adel Tawfik Tawfik
20178058	Aya Tarek Fouad shehada
20168005	Hager Sayed Hamdy Abdalaliem
20168004	Maya Ahmed Hussein Kamal
20178057	Omnia Emam Saad Ibrahim

Graduation Project
Academic Year 2019-2020
Final year Documentation

Table of Content

List of figures

List of Tables

List of Abbreviations

IDE: Integrated Development Environment
ML: Machine Learning
GA: Genetic Algorithms
SVM: Support Vector Machine
SVC: Support Vector Classifier
KNN: K-Nearest Neighbors
LOO-XVE: leave-one-out cross validation error
MCMC: Markov Chain Monte Carlo

Acknowledgement

Firstly, and always we thank the Mighty Allah for guiding us all to achieve what we were purposed to do in life, as Allah has brought as all together to complete our goals and bestowed the patience and diligence to finish this project as one team.

We would like to sincerely thank and gratefully acknowledge the assistance and guidance of our kind and great supervisor Professor Dr. Amr Badr Allah rest his soul and Dr. Sabah Sayed for their patience and knowledge throughout this project. Also, we appreciate their encouragement and optimism which helped us get through it. We will always respect and wish them the very best in life.

And finally, we would like to appreciate our family, friends and all the faculty members who taught us during our degree program for helping us and supported us with their constant encouragement, without them we would not have reached this final stage, so we wish them all the best of luck and thanks.

Abstract

There have been several empirical studies addressing cancer using machine learning and soft computing techniques. Many claim that their algorithms are faster, easier, or more accurate than others are. This study is based on genetic programming and machine learning algorithms that aim to construct a system to accurately differentiate between benign and malignant breast tumors. The aim of this study was to optimize the learning algorithm. In this context, we applied the genetic programming technique to select the best features and perfect parameter values of the machine learning classifiers. The performance of the proposed method was based on a classification report. The present study proves that genetic programming can automatically find the best model by combining feature preprocessing methods and classifier algorithms.

❖ Chapter One

1.1 Introduction

Microarrays are emerging technologies that allow biologists to better understand the interactions between disease and normal states, at genes level. However, the amount of data generated by these tools becomes problematic when data are supposed to be automatically analyzed (e.g., for diagnostic purposes). In this work, we present a gene selection method based on Genetic Algorithms (GAs). The proposed method uses GAs to search for subsets of genes that optimize 2 measures of quality for the clusters presented in the domain. Thus, data are better represented and classification of unknown samples may become easier.

1.2 Motivation

In this day and age, almost everything is computerized Consequently, programs are needed even for the biology field of study. In other words, this project faces topics of both the biology and computer science fields.

It is rare that a set of measurement and analytic techniques can revolutionize biomedical research and clinical practice. It is precisely because the excitement and the expectations surrounding this field are so high that we are compelled to do this project.

Here, we provide a source of challenge, problem, and dataset that will simulate basic development while furthering important goals in biological discovery. Therefore, this project will use underlying computer science technique (machine learning).

So, this brings an exciting field of study for Machine Learning researchers. In addition to this, noise and variability of the data make this domain more exciting.

This project faces what should be done when having a huge amount of data. How to choose features in data that will give you as good or better accuracy whilst requiring less data. Lastly, this will provide users with a way to identify the most effective genes in causing many diseases.

1.3 Beneficiary

Who is this intended for? Answering this question has served as our aim in this project. There are three audiences in that we have had in mind.

1. **Experienced biologists with limited experience using microarrays.**
2. **Experienced informaticians with limited experience analyzing microarray data.**
3. **Students entering the field of Bioinformatics.**

How is this beneficial for them?

- **Saves time and effort for doctors and scientist to find genes that causes cancer.**
- **Future data will be less, but more informative.**
- **Doctors can aid patients before disease progresses by knowing their gene history**.

1.4 Problem Definition

You may have noticed that squirrels in one location may be grey while in another location they are brown, even though they are the same species. All domestic dogs are the same species even though there is a huge variation in the way they look. Each of these is an example of genetic selection.

All species of living things have physical traits that are inherent to that species. Genetic selection is the process by which certain traits become more prevalent in a species than other traits. These traits seen in an organism are due to the genes found on their chromosomes. The genes code for the traits that we are able to observe.

Essentially, genes can boost their own replicative success in two basic ways. First, they can influence the odds for survival and reproduction of the organism they are in individual reproductive success or fitness. Second, genes can also influence the organism to help other organisms who also likely contain those genes—known as “genetic relatives”—to survive and reproduce (which is called inclusive fitness).

In these recent years, analysis of microarray gene expression data has become an important tool, for the genes have been found to be expressed at significantly different levels in the cell. One of- the main applications of microarrays in medicine is class prediction, which is to identify the class membership of a sample based on its gene expression profile. The process involves the construction of a statistical classifier that learns from the training set data and predicts the class membership of the test samples.

However, one key problem in this analysis is the huge number of genes in microarray data. Many of them are irrelevant or redundant to some specified disease. Thus, selecting highly discriminating genes is critical to improving the accuracy of disease classification and prediction.

This curse of dimensionality presents a challenging problem for class prediction, for it often results in high generalization error. Fewer attributes is desirable because it reduces the complexity of the model, and a simpler model is simpler to understand and explain.

The microarray dataset used for this project is that of Breast cancer. Breast cancer is a disease in which cancerous (or malignant) cells develop uncontrollably in the breast tissue. It is the most common type of cancer seen in women and the leading cause of death.

Cancer causes the cells to multiply uncontrollably. They do not die at the usual point in their life cycle. This excessive cell growth causes cancer because the tumor uses nutrients and energy and deprives the cells around it.

So, what exactly is the problem?

To summarize:

1. A large microarray with a lot of data to observe
2. How to find the useful aspects of the data used.
3. How to reduce the data, to only use the useful information it provides.
4. What to do if the data is or is not classified.
5. How accurate is our classification of this data.

1.5 Problem Objective

Gene selection addresses many problems in microarray datasets such as reducing the number of irrelevant and noisy genes data that do not contribute to the accuracy of a predictive model or may in fact decrease the accuracy of the model.

One effective solution to alleviate the problem is to perform gene selection to reduce the dimensionality of the microarray data.

Gene selection selects the most related genes to improve the classification results. A gene selection method searches for an optimal or near optimal subset of genes with respect to a given evaluation criterion.

Optimization refers to finding the values of inputs in such a way that we get the “best” output values. The definition of “best” varies from problem to problem, but in mathematical terms, it refers to maximizing or minimizing one or more objective functions, by varying the input parameters.

So, what exactly are ours goals?

To summarize:

1. To obtain a subset of features (genes) that can most efficiently classify the data.
2. Reduce the dimensionality
3. To optimize the classification results

1.6 The used tools in the project

This project does not need any hardware except ones' personal computer. This project is mostly a research project that will only require a dataset to be inserted by the user (researcher).

There are many different type of IDEs that could implement this project. Such as:

- **Python**
- **C / C++**
- **Java**
- **R**

The best IDEs for this particular project would be **python** or **R** because they include libraries that can benefit us in this project.

Therefore, it was concluded that **python** IDE would best suit the implementation for this project.

1.7 Project development methodology

Primary methods will include the following:

- **Search for microarrays that is related to our project.**
  To search common online sites such as NCBI, TCGA, …etc for datasets related to cancers. And as mentioned to the used dataset for this research is a supervised breast cancer microarray.

- **Enter the microarray in the program**
  After downloading the related data. We enter the data to our preferred IDE to start our research.

- **Pre-process the microarray data**
  To eliminate a large number of noisy and redundant genes.
  SelectKBest methods generally used as a preprocessing step.

- **The GA program runs**
  GA has the following three operators: reproduction, crossover and mutation. Genetic algorithm starts with the generation of a random population, then, the fitness of the each individual is determined using appropriate fitness function.

- **Evaluate accuracy of program through testing phase**
  To take our output data from the GA program and test its accuracy by running the output in a machine learning technique called KNN

1.8 Gantt chart

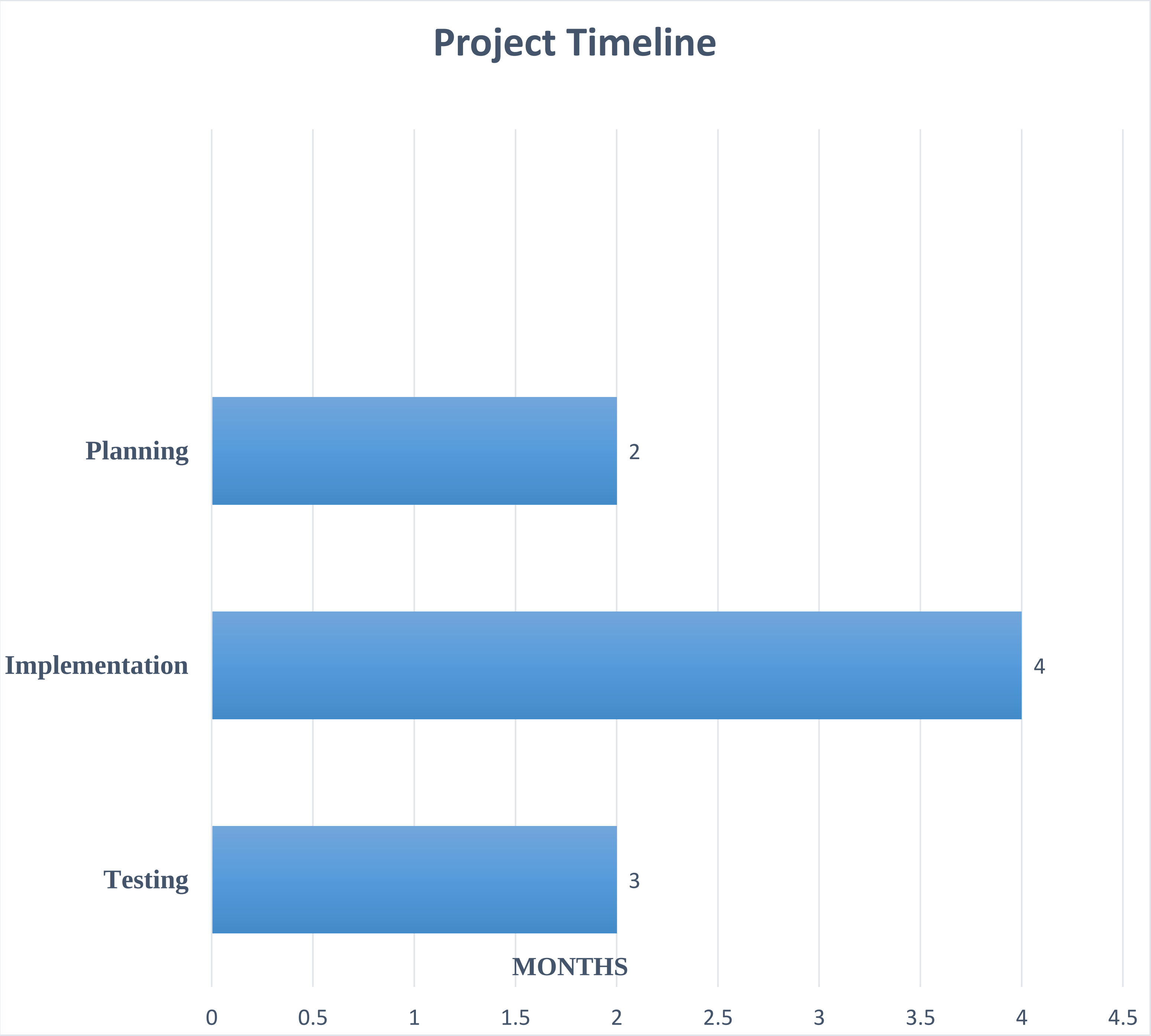


Figure 1.1: Gantt chart of project timeline

Task	**Task Title**	**Description**	**Task status**
Planning	Gene expression dataset	Finding a dataset that suits our project	Completed
Planning	Genetic algorithm (GA)	An algorithm that will determine best subset of genes	Completed
Implementation	Dataset preprocessing	Fixing and cleaning dataset to be processed in program	Completed
Implementation	Support vector machine (SVM)	An algorithm for the fitness of individual data to be used in GA	Completed
Testing	Integration	Putting all implemented parts together	Completed
Testing	Testing and verification	Testing the entire program and verifying the final output	Completed

Table 1.1: Task completion, Project timeline

1.9 Report Organization

Chapter Two: Related Work

In chapter two, we will establish other work associated with our research. There, we show that different ways to achieve our goals.
We will declare the authors of these methods, the dates they were founded and their explanation.

Chapter Three: System Analysis

In chapter three, we will clarify project specification. Where
a good project specification is a simple but complete description of a program's functionality and purpose. It contains descriptions of how the program will be used from a user perspective and performance details such as usability, reliability and stability. In addition, illustrate a use case diagram that emphases our program.

Chapter Four: System Design

In chapter four, we will portray our research diagrams.
The purpose of a component diagram is to show the relationship between different components in a system (the program’s algorithm in our case) . The term "component" refers to a module of classes that represent independent systems or subsystems with the ability to interface with the rest of the system. Also, a sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together.

Chapter Five: Model proposed

In chapter Five, we will demonstrate step by step our implementation methods. We will give an explanation beginning from how our data looks like, how it is represented and constructed. Also, how it is preprocessed.

In addition, we will demonstrate our program functions, how it affects our data and such.

Moreover, we will explain the machine learning technique that was used in the model.

Chapter Six: Model Setup and Results

In chapter Six, we will show our parameter settings and describe it. We will also show our model results accordingly. And lastly, give a conclusion to our research.

❖ Chapter Two

Related Work

In this chapter, we will explain some topics that are related to our project. Our project consists of a GA program with the use of Support Vector Machine (SVM) ML technique.

2.1 Gene selection using Bayesian variable selection approach

In this approach, (Kyeong Eun Lee, Naijun Sha, Edwar R.Dougherty, Marina Vannucci, Bani K.Mallick ,2003) propose a hierarchical Bayesian model for gene (variable) selection. They employ latent variables to specialize the model to a regression setting and use a Bayesian mixture prior to perform the variable selection.

They control the size of the model by assigning a prior distribution over the dimension (number of significant genes) of the model. The posterior distributions of the parameters are not in explicit form and we need to use a combination of truncated sampling and Markov Chain Monte Carlo (MCMC) based computation techniques to simulate the parameters from the posteriors. The Bayesian model is flexible enough to identify significant genes as well as to perform future predictions.

The Main Difference between our approach and , (Kyeong Eun Lee, Naijun Sha, Edwar R.Dougherty, Marina Vannucci, Bani K.Mallick ,2003) approach is the main technique (Kyeong Eun Lee, Naijun Sha, Edwar R.Dougherty, Marina Vannucci, Bani K.Mallick ,2003) use Bayesian variable selection approach as the main technique, we prefer to use Genetic Algorithm approach, which compute a set of solutions instead of a single solution, and it avoids becoming trapped in a local optimum, which may happen in other optimization techniques.

2.2 Gene selection and classification using random forest

In this approach, (Ramón Díaz-Uriarte and Sara Alvarez de Andrés, 2006) investigate the use of random forest for classification of microarray data (including multi-class problems) and propose a new method of gene selection in classification problems based on random forest.

Using simulated and nine microarray data sets they show that random forest has comparable performance to other classification methods, including KNN, and SVM, and that the new gene selection procedure yields very small sets of genes (often smaller than alternative methods) while preserving predictive accuracy.

The Main Difference between our approach and (Ramón Díaz-Uriarte and Sara Alvarez de Andrés, 2006) approach is the classification method while (Ramón Díaz-Uriarte and Sara Alvarez de Andrés, 2006) use the random forest as a classification method, we prefer to use the support vector machine.

2.3 Gene selection using hybrid Genetic Algorithm/support vectors machine

Our project will be based on this approach,

In this approach, (Shutao Li, Xixian Wu and Xiaoyan Hu, 2008) present a gene selection method based on genetic algorithm (GA) and support vector machines (SVM). First, the Wilcoxon rank sum test is used to preprocess the original data, then a hybrid GA/SVM selects different gene subsets on different training sets and the number of each gene appearance in the different selected gene subsets is recorded. Finally, the genes with the highest selected number are used to form a final subset for tumor classification.

The Main Difference between our approach and (Shutao Li, Xixian Wu and Xiaoyan Hu, 2008) approach is the preprocess step while (Shutao Li, Xixian Wu and Xiaoyan Hu, 2008) use Wilcoxon rank sum test to eliminates a large number of noisy and redundant genes. Which greatly reduces the searching time of GA, we use SelectKBest Method, regardless the difference between preprocess step our approach and (Shutao Li, Xixian Wu and Xiaoyan Hu, 2008) approach are the same.

2.4 Feature Selection using Genetic Algorithms

In this approach, (Vandana Kannan, 2018) use GA to select features for different applications, and develop a solution that uses a reduced feature set (selected by GA) to classify images based on their domain/genre, In this approach (Vandana Kannan, 2018) explore 3 classification algorithms – Random Forest (RF), Support Vector Machine (SVM), and Neural Networks (NN), and perform 10-fold cross-validation with all 3 methods. The idea is to evaluate the performance of each classifier with the reduced feature set and analyze the impact of feature selection on the accuracy of the model

The Main Difference between our approach and (Vandana Kannan, 2018) approach is the classification algorithms while (Vandana Kannan, 2018) explore 3 classification algorithms – Random Forest (RF), Support Vector Machine (SVM), and Neural Networks (NN), we only use Support Vector Machine (SVM) as a classification algorithm because the breast cancer dataset that we working on it give appropriate accuracy of the (SVM) model with no need to use another classification algorithms .

❖ Chapter Three

System Analysis

3.1 Project specifications

In chapter two, we will clarify a simple but complete description of the program's functionality and purpose and will explain how the user will interact and the output of the program.

3.1.1 Functional requirements

The aim of our research is to find a subset of genes from a large dataset that will determine if a person is classified with cancer or not. This final subset is obtained by analyzing the frequency of appearance of each gene in the different gene subsets. In other words, we want to do a feature selection.
We use the algorithm of genetic algorithm to do this.

3.1.2 Non-functional requirement

- **Performance:**

Feature (Genes) subset selection works by removing features that are not relevant or are redundant. The subset of features selected should give the best performance according to some objective function. In this proposed method, we examine the use of a Genetic Algorithm to do so.

The Genetic Algorithm is an example of the Wrapper approach which measures the “usefulness” of features based on the classifier performance (SVM classifier). Thus, wrapper methods are essentially solving the “real” problem (optimizing the classifier performance), but they are also computationally more expensive compared to filter methods due to the repeated learning steps and cross-validation.
So, when thinking about the performance we need to take into account that there are limitations to GA.

- **Limitations of GA**

Like any technique, GAs also suffers from a few limitations. These include:

- GA is not suited for all problems, especially problems which are simple and for which derivative information is available.
- Fitness value is calculated repeatedly which might be computationally expensive for some problems.
- Being stochastic, there are no guarantees on the optimality or the quality of the solution.
- If not implemented properly, the GA may not converge to the optimal solution.

- **Usability**

The use of machine learning techniques to automatically analyze data is becoming increasingly widespread. However, the size of the data to be processed has increased the past 5 years and therefore feature selection has become a requirement before any kind of classification takes place. Our model makes it easier for users to accurately efficiently classify breast tumors (benign or malignant).

In addition, with some tuning to this program. It shows that users can use other types of raw data with the same implemented program and produce results. Other users can acquire results using the methods explained in this research.

- **Reliability**

Unlike feature extraction methods, feature selection techniques do not alter the original representation of the data.

After applying a breast cancer dataset with the resulted subset of genes to a classifier (e.g., KNN classifier) the results are supposed to show that the proposed method has excellent selection and classification performance which can yield 100% classification accuracy using only a small number of genes.

- **Stability**

The process of GA was repeated 10 times to observe the stability of the program, and there was a non-neglectable numbers of repeated genes in 10 runs , demonstrating high stability.

3.2 Use case diagram

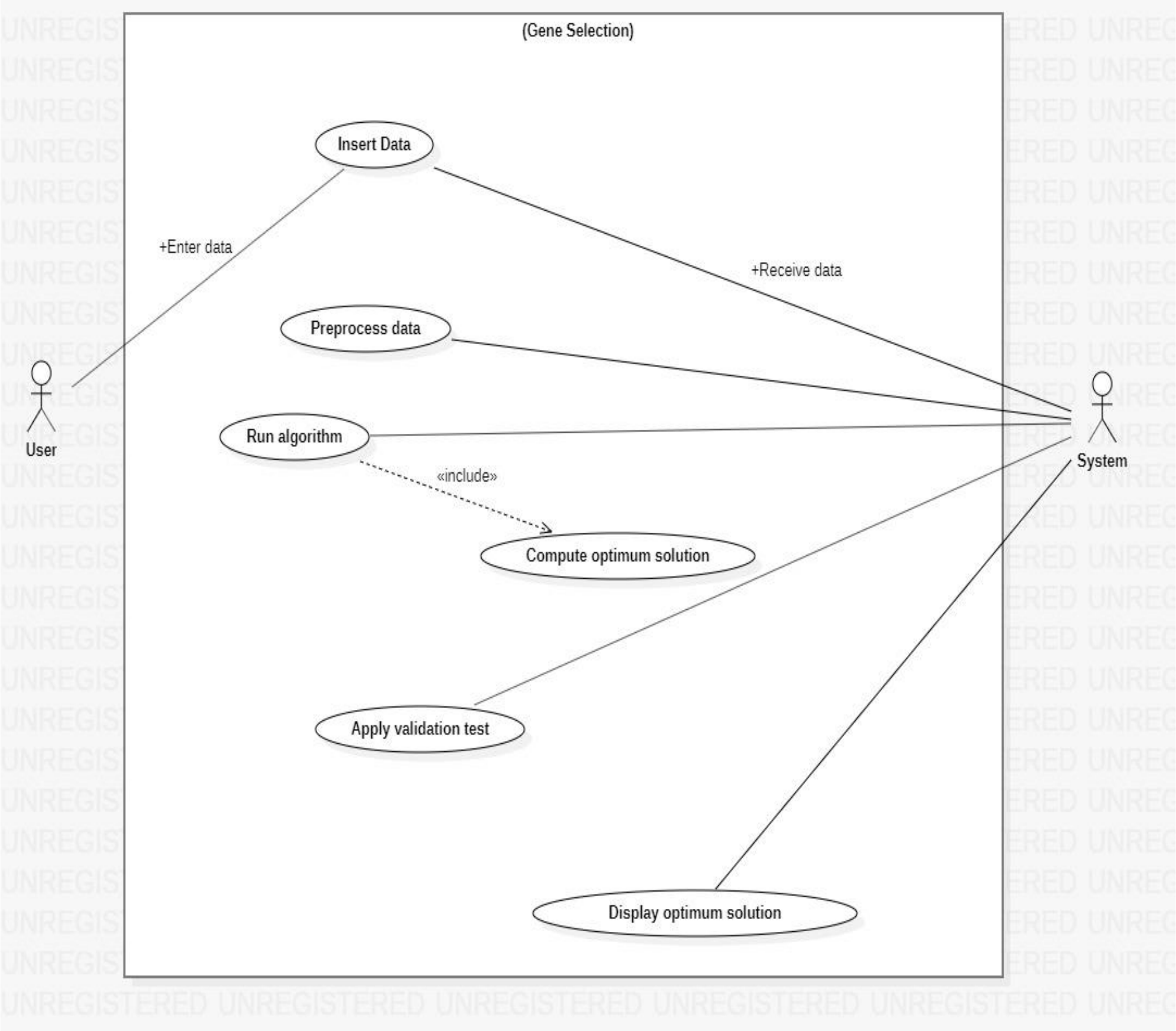


Figure 3.1: Use case diagram

❖ Chapter Four

System Design

4.1 System Component Diagram

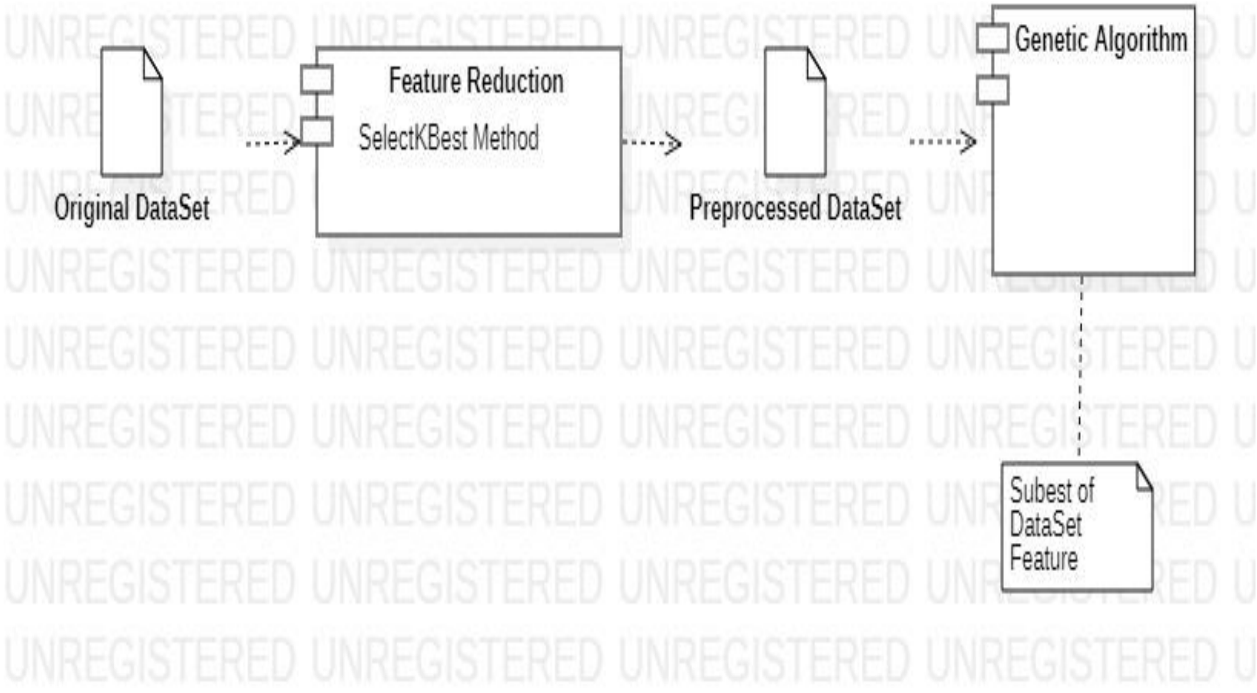


Figure 4.1: System component diagram

4.2 Sequence Diagram

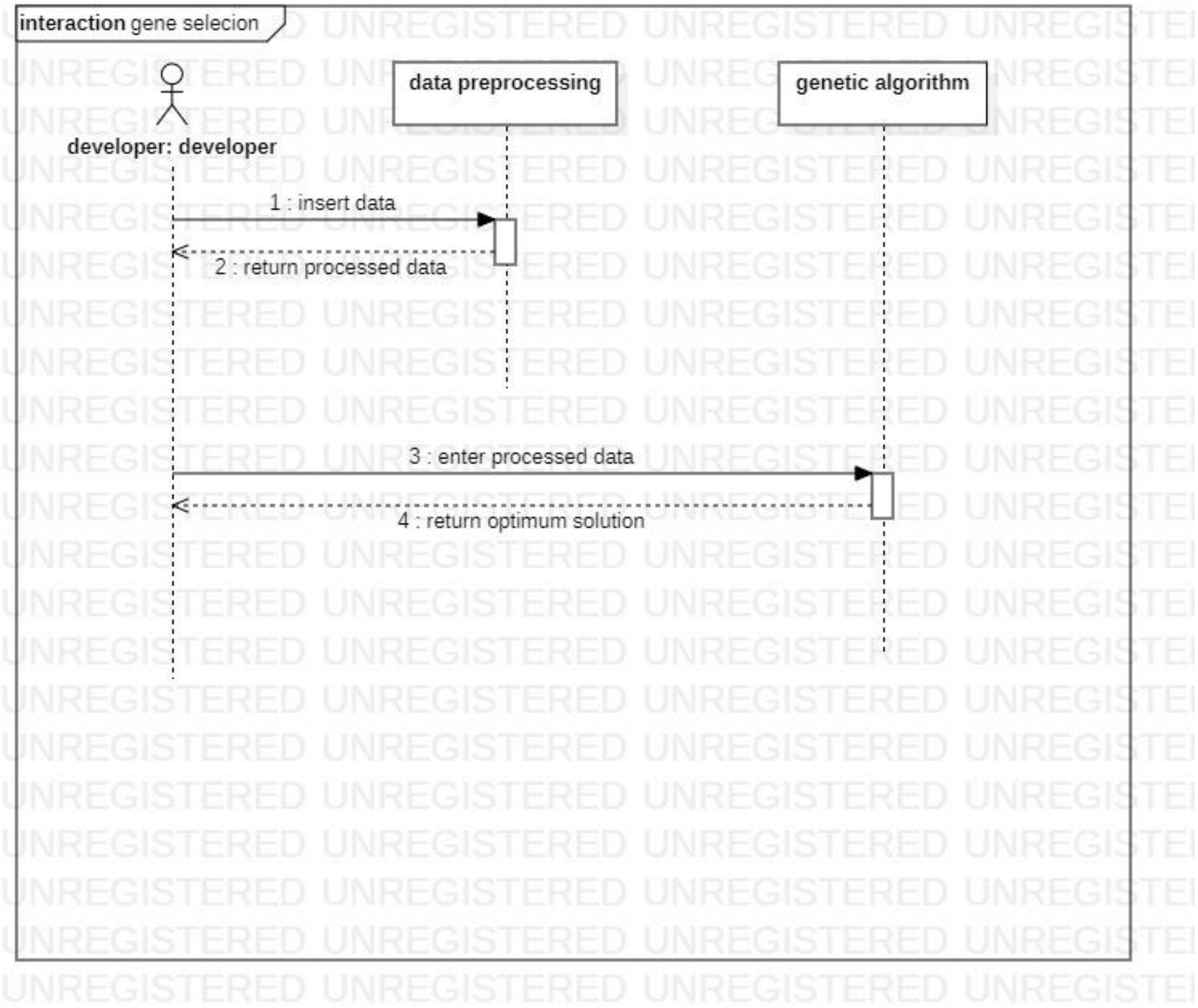


Figure 4.2: Sequence Diagram

❖ Chapter Five

Model Proposed

5.1 Data Preprocessing

5.1.1 Preprocessing of file components

First we had our Dataset file preprocessed, removing all the unneeded information attached to the dataset in file.

5.1.2 Preprocessing of Dataset

The second step taken is that of preprocessing our data, where a feature selection technique is applied to reduce the amount of input variables that are to be used in the GA.
The aim of preprocessing step is that it generally eliminates a large number of noisy and redundant genes in original dataset as shown in **tables 5.1,5.2** respectively. This step greatly reduces the searching time of GA, The method we follow is that of **SelectKbest technique**.

	#Cases.	#Features	#Class (0)	#Class (1)
Breast cancer Dataset	58	18,383	27	31

Table 5.1: Original DATA SET CHARACTERISTICS.

	#Cases.	#Features	#Class (0)	#Class (1)
Breast cancer Dataset.	58	1000	27	31

Table 5.2: Preprocessed DATA SET CHARACTERISTICS.

SelectKBest is a feature selection technique that scores the features against the target variable using a function (in our case **f_classif**) & then retains the most significant features, as shown in **table 5.3**.

	Score_func.	**K**
Values assigned	F_classif	1000

Table 5.3: SelectKBest function parameters.

5.2 Genetic Algorithm

- **Flowchart**

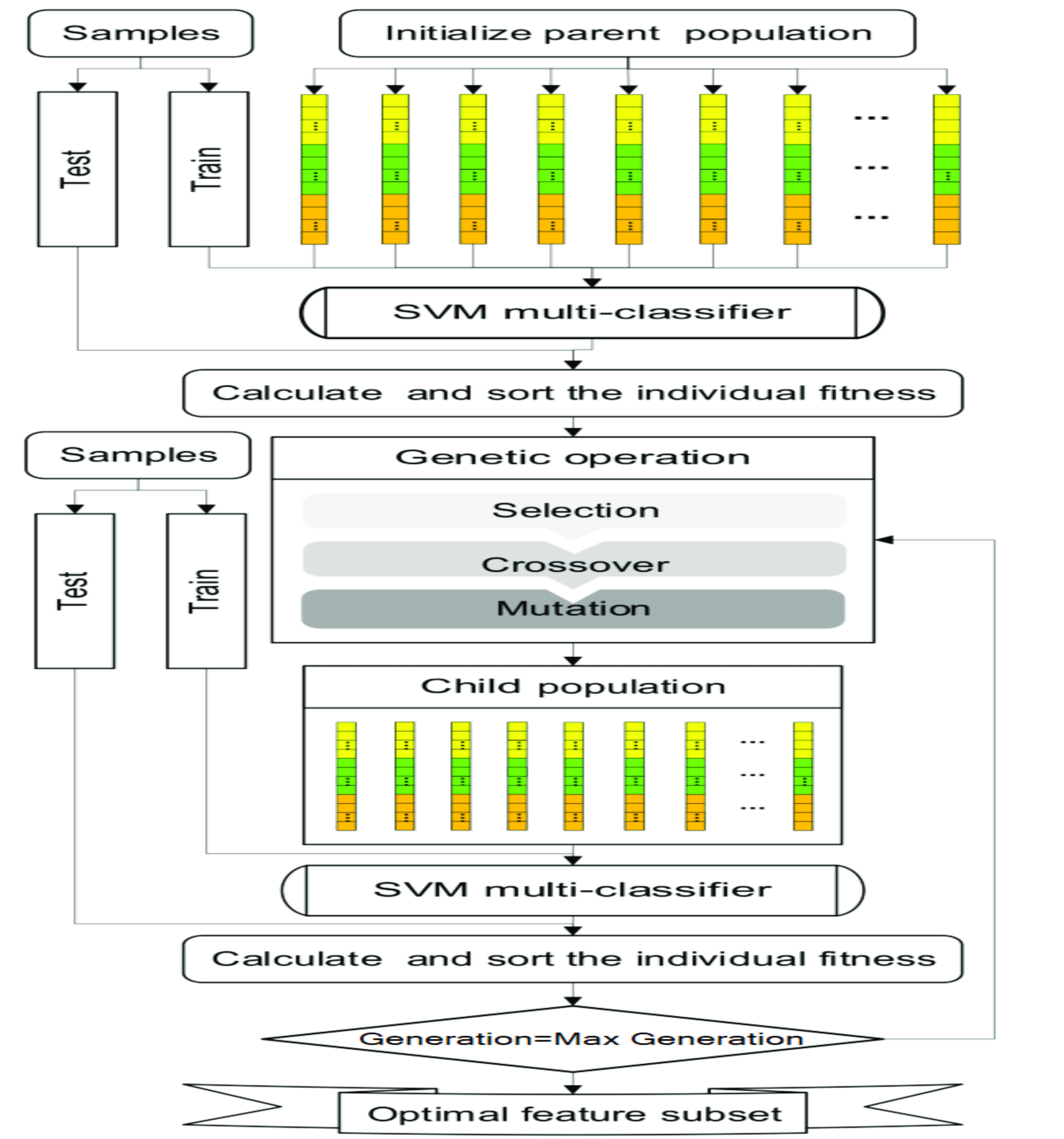


Figure 5.1: Flowchart

Algorithm–

```
Step1: Initialize subsets (parent) population
For each subset:
    Build classifier using support vector machine.
    Calculate fitness of each subset.
    Sort parents according to their fitness.

Step2: From 1 to Maximum Generation:
    Do:
        Selection.
        Crossover.
        Mutation
        For each new subsets (child):
            Build classifier using support vector machine.
            Calculate fitness of each subset.
        Replace old Generation with new Generation
```

- Genetic algorithms (GA), is a general adaptive optimization search methodology which supports an analogy of Darwinian natural selection and genetics biological systems, could be a promising alternative to standard heuristic search.

- GA works with a collection of candidate solutions referred to as a population. Based on the Darwinian principle of survival of the fittest, the GA gains the optimum solution when a series of repetitive computations are applied. GA generates successive populations of alternate solutions which are represented by chromosomes, i.e. an answer to the problem, till acceptable results are obtained.

- A fitness function assesses the standard of a solution in the analysis step.

- The crossover and mutation functions are the units that impact the fitness value.
- For reproduction, chromosomes are selected by evaluating the fitness value. The fitter chromosomes have higher chance to be elected into the recombination pool using the roulette wheel or the tournament selection methods.

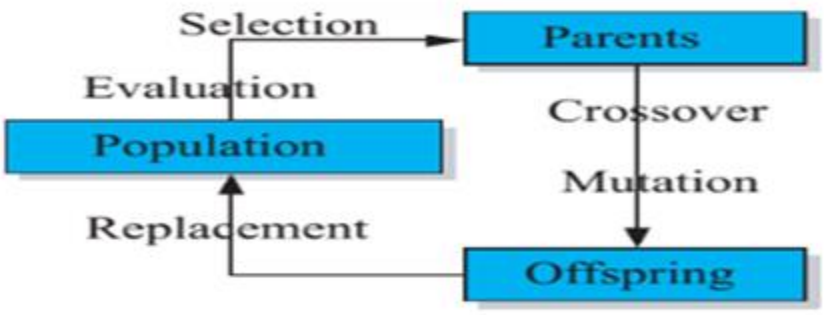


Figure 5.2: Evolutionary cycle

- **Main steps of a GA:**

1. Construction of the first generation
2. Fitness function computation
3. Selection While stopping criteria not met do
4. Crossover
5. Mutation
6. Selection End

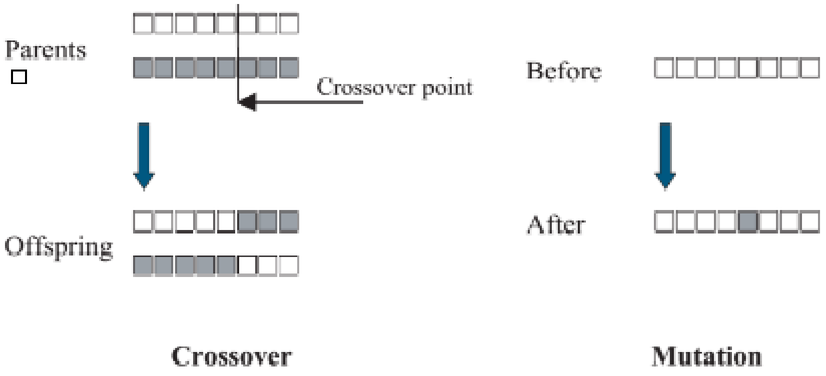


Figure 5.3: Genetic crossover and mutation operation

1. Construction of Generation

- Populations will consist of chromosomes whose genes where randomly assigned the values ones and zeros , for ‘ones’ to be representing active genes and ‘zeros’ representing inactive ones, as shown in **figure 5.4,**

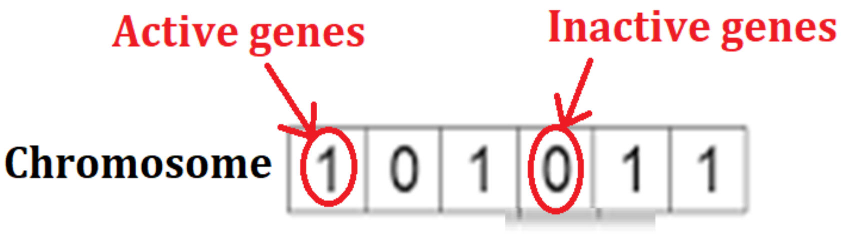


Figure 5.4: chromosome

2. Fitness function

- After chromosomes are initialized they are passed to the fitness-evaluation function, after which each chromosome will be assigned with their fitness. This is the first main step in GA.

- Fitness of chromosomes is obtained by using the SVM machine learning technique.

▪ SVM model

- **S**upport **V**ector **M**achine is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model sets of labeled training data for each category, they’re able to categorize new text.

- **SVM** belongs to a family of **generalized linear classifiers.** In another terms, Support Vector Machine (SVM) is a classification and regression prediction tool that uses machine learning theory to maximize predictive accuracy while automatically avoiding over-fit to the data.

- They were first developed to solve the classification problem, but recently they have been extended to solve regression problems.
- SVM models are constructed to seek a decision surface (hyper plane) that may separate the data points into two categories with a maximal margin between them, i.e., the distance from the hyper plane to the nearest examples from each of the classes, as shown in **figure 5.4**.

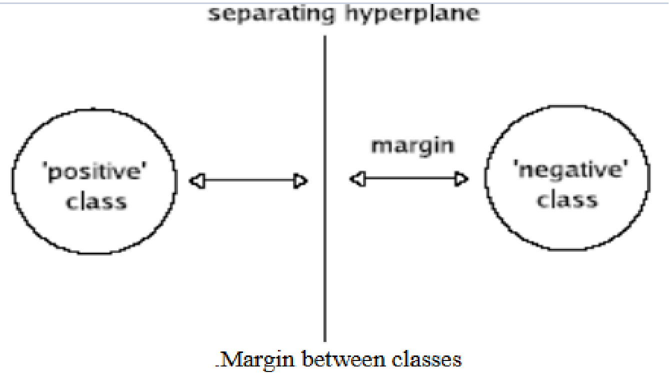


Figure 5.4: SVM

- Now, since we have generally discussed the main idea of the SVM technique, we need to know the classifier's different types; basically and most commonly discussed method is the simple **linear support vector classifier**.

  - In the simplest case, compounds from different classes can be separated by linear hyper plane; such hyper plane is defined solely by its nearest compounds from the training set. Such compounds are referred to as **support vectors**, giving the name to the whole method.

- For calculating the SVM we see that the goal is to correctly classify all the data. For mathematical calculations we have,

  [a] If Yi= +1; $wxi + b \geq 1$
  [b] If Yi= -1; $wxi + b \leq 1$
  [c] For all i; $yi\ (wi + b) \geq 1$

- Another category of methods to be discussed are the **Kernel** methods; These methods owe their name to the use of **Kernel Trick** ,which enable them to operate in a high dimensional, implicit **feature space** without ever computing the coordinates of the data in that space.

- **Kernel:** If data is linear, a separating hyper plane may be used to divide the data. However it is often the case that the data is far from linear and the datasets are inseparable. To allow for this **kernels** are used to **non-linearly** map the input data to a high-dimensional space. The new mapping is then linearly separable, This mapping is defined by the Kernel:

$$K(x, y) = \phi(x) . \phi(y)$$

- **Feature Space:** Transforming the data into feature space makes it possible to define a similarity measure on the basis of the dot product. If the feature space is chosen suitably, pattern recognition can be easy.

- The idea of the kernel function is to enable operations to be performed in the input space rather than the potentially high dimensional feature space.

- **Kernel trick:** The training set is not linearly separable in an input space. The training set is linearly separable in the feature space.

- **Examples on different Kernel Functions:**

  - Different SVM algorithms use different types of kernel functions. For example **linear, nonlinear, polynomial, radial basis function (RBF), and sigmoid**.

1. **Linear:** used when the data is linearly separable, that is, it can be separated using a single Line. This is the most common kernels to be used, equation:
   K(xi, xj) = xi . xj

2. **polynomial**: A popular method for non-linear modeling, preferable equation:
   K(xi, xj) = (xi . xj +1)^{d}
3. **Gaussian radial basis function (RBF):** Radial basis functions most commonly with a Gaussian form:

   $K(xi, xj) = \exp(-\gamma||xi - xj||^2)$ for $\gamma > 0$, sometimes parameterized using:

   $\gamma = 1/2\ \sigma^2$

- The most used type of kernel function is **RBF (**the one used in our case**).**
- As shown in **table 5.4** below, SVM classifiers using an RBF kernel has two parameters: **gamma** and **C**.

	kernel	C	gamma
Values assigned	‘rbf’	1000000	0.001

Table 5.4 SVC-RBF parameters

i. **<u>Gamma</u>:** can be thought of as the ‘spread’ of the kernel and therefore the decision region. When gamma is low, the ‘curve’ of the decision boundary is very low and thus the decision region is very broad. When gamma is high, the ‘curve’ of the decision boundary is high, which creates islands of decision-boundaries around data points.

ii. **<u>C</u>:** is the ‘penalty’ for misclassifying a data point. When C is small, the classifier is okay with misclassified data points (high bias, low variance). When C is large, the classifier is heavily penalized for misclassified data and therefore bends over backwards avoid any misclassified data points (low bias, high variance).

- While applying SVM function, a **cross validation** method should be used to evaluate SVM model on a limited data sample in order to estimate the skill of the model on unseen data and to avoid overfitting; more efficient use of data as **every observation is used for both training and testing**.

- Cross Validation method used in our case is **K-Fold C**ross **V**alidation.

- This procedure has a single parameter called **k** that refers to the number of groups that a given data sample is to be split into.

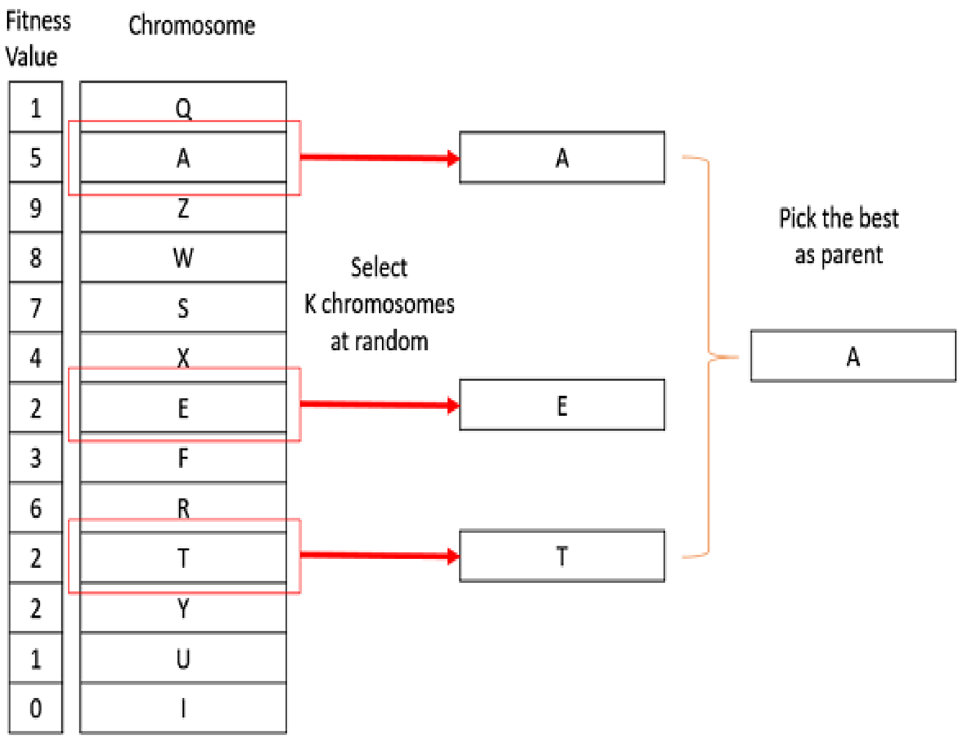
  - **Steps**

    1. Shuffle the dataset randomly
    2. Split the dataset into k groups
    3. For each unique group:
       i. Take the group as a hold out or test data set
       ii. Take the remaining groups as a training data set
       iii. Fit a model on the training set and evaluate it on the test set
       iv. Retain the evaluation score and discard the model
    4. Summarize the skill of the model using the sample of model evaluation scores

iii. Selection

- After the fitness function is computed, another function named selection is called where based on the fitness of chromosomes, parents are to be selected.
- The selection function can be implemented in different ways. The way we used is called **Tournament Selection**.

- In K-Way tournament selection, we select K individuals from the population at random and select the best out of these to become a parent. The same process is repeated for selecting the next parent. Tournament Selection is also extremely popular in literature as it can even work with negative fitness values.



Fitness Value	Chromosome
1	Q
5	A
9	Z
8	W
7	S
4	X
2	E
3	F
6	R
2	T
2	Y
1	U
0	I

Figure 5.5: Tournament Selection

Algorithm --

1. Select k individuals from the population and perform a tournament amongst them

2. Select the best individual from the k individuals

3. Repeat process 1 and 2 until you have the desired amount of population

4. Crossover

- After the parents are selected from the selection function .they are passed to a **crossover** function.

- Crossover is the critical genetic operator that enables new solution regions within the search space to be explored; it is a random mechanism for exchanging genes between two chromosomes.

- The objective of this step is to gather interesting features of several solutions in new individuals by making combination of the previously retained solutions.

- It‘s necessary to notice that this step is independent from the optimization, that's to say, a crossover can produce good and worse solutions equally. Only the selection step is used to eliminate bad solutions.

- Crossover occurs during evolution according to a user-defined crossover probability (Pc).

- Just like the selection function crossover have different ways to be implemented. Here, we had used the One-Point crossover.

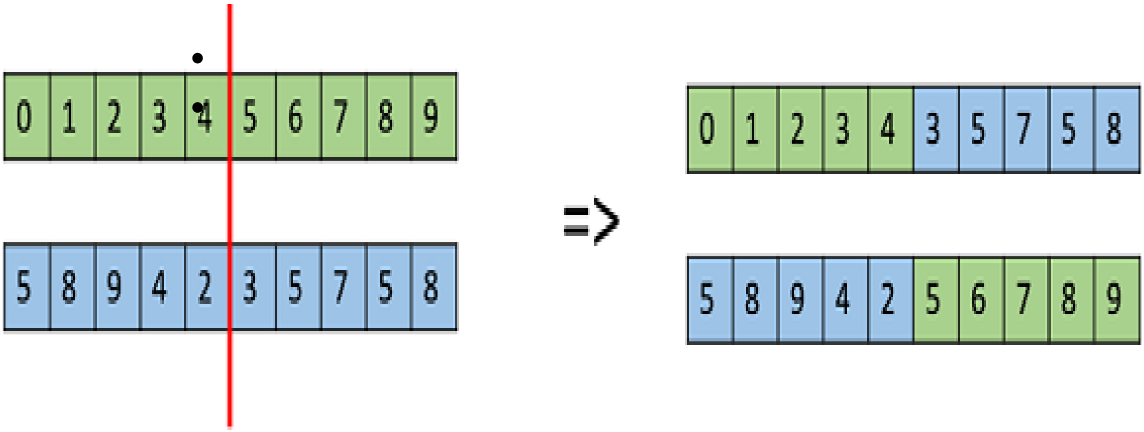


Figure 5.6: Crossover

- In this one-point crossover, a random crossover point is selected and the tails of its two parents are swapped to get new off-springs.

```
Algorithm – –

Generate random number R1 ∈ [1, chromosomeLength - 1]
Let crossover point = R1
Generate random number R2 ∈ [0, 1]
 If R2 ≤ crossover probability Pc    then
                    For i = 1 to R1    do
                            Offspring1 [ i ] = parent1 [ i ]
                            Offspring2 [ i ] = parent2 [ i ]
                    For j = R1 +1 to chromosomeLength  do
                            Offspring1 [ j ] = parent2 [ j ]
                            Offspring2 [ j ] = parent1 [ j ]

 Else  If  R2 > Pc      then
                    Offspring1 = parent1
                            Offspring2 = parent2
```

5. Mutation

- After the crossover, there is another function called **mutation**.

- In mutation the genes could often be altered, i.e. in binary genes change genes code from 0 to 1 or vice versa

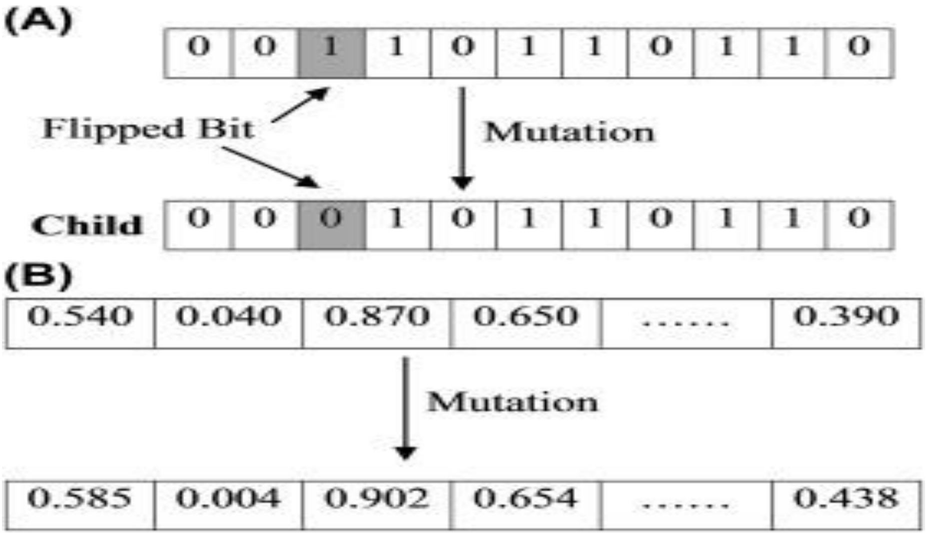


Figure 5.7: Mutation

```
Algorithm —

For i = 1 to chromosomeLength  do
        Generate random number R ∈ [0, 1]
        If  R  ≤  mutation probability Pm     then
         flip  bit [ i ]
        else if  R > Pm then
         leave bit [ i ]
```

6. Replacement

- Offspring replaces the previous population using the **Diversity Replacement** strategy and forms a replaced or new population in the next generation. The evolutionary {biological process} process operates several generations till termination conditions satisfy.

```
Algorithm —
Replace (P,S,n) :

    //S is set of chromosomes generated as offspring
    //P is parent generation of chromosomes
    //n is population size

    P:=S
    End
```

5.3 Validation Function (KNN)

- Finally, after obtaining the subset of genes that have been worked on to be the best discriminative genes among cancer traits, a validation step needs to be taken, that is passing our list of genes obtained to a validation function, to test the actual efficiency of these genes discrimination power across some given classified-dataset.
- Validation function used in our case is K-Nearest-Neighbors (KNN) classifier function; a supervised machine learning model.
- K-NN models work by taking a data point and looking at the ‘k’ closest labeled data points. The data point is then assigned the label of the majority of the ‘k’ closest points.

Algorithm —

```
-  Let m be the number of training data samples. Let p be an unknown point.
1. Store the training samples in an array of data points arr[]. *This means each element of this array represents a tuple (x, y).*
2. for i=0 to m:
3. Calculate Euclidean distance d(arr[i], p).
4. Make set S of K smallest distances obtained. Each of these distances corresponds to an already classified data point.
5. Return the majority label among S.
```

❖ Chapter Six

Model Setup and Results

5.1 Parameter settings

The setting of GA parameters is very important in the GA/SVM method. For GA parameters, if the population size is too small, it is difficult to get the best resolution and too big a population size will require a long convergence time.

Thus, the size is normally 50–100. If the crossover Pc is too low, it is difficult to search forward and a Pc value too big will damage individuals with high adapting value. Therefore, the Pc is normally 0.3–0.9. If the mutation rate Pm is too low, the new individual is hard to produce and too high a Pm would make the GA simple search at random. Thus, the Pm is normally 0.01–0.2. The final parameters settings are shown in **Table 6.1**.

GA Parameters	**Breast Cancer**
Population	100
Chromosome Length	1000
Generation	250
Crossing rate	0.3
Mutation rate	0.09

Table 6.1: Genetic algorithm (GA) parameter settings

5.2 Model Results

- The process of GA was repeated 10 times and the most repeated genes in 10 runs is considered as the final result, the breast cancer dataset, the highest accuracy is 98% with only four genes selected as shown in **Table 6.2**.

- As a validation step we ran the K Nearest Neighbors on the final gene subset and it gave 94% accuracy

Gene_ID	OFFICIAL GENE SYMBOL	Description
1552519_at	ACVR1C	ACVR1C is a type I receptor for the TGFB family of signaling molecules, Authors report that the TGFss superfamily receptor ALK7 is a suppressor of tumorigenesis and metastasis, as revealed by functional studies in mouse models of pancreatic neuroendocrine and luminal breast cancer
1554044_a_at	MRAP	This gene encodes a melanocortin receptor-interacting protein. Mutations in this gene have been associated with familial glucocorticoid deficiency type 2
1555758_a_at	CDKN3	The protein encoded by this gene belongs to the dual specificity protein phosphatase family. This gene was reported to be deleted, mutated, or overexpressed in several kinds of cancers.

1555778_a_at	POSTN	This gene encodes a secreted extracellular matrix protein that functions in tissue development and regeneration, this protein plays a role in cancer stem cell maintenance and metastasis.

Table 6.2: The top four important genes selected from the breast cancer dataset

5.3 Conclusion

We proposed a hybrid GA/SVM method for gene selection, and the results show that a perfect prediction is obtained. At the same time, the effects of different SVM parameters and GA parameters and SVM classifiers on classification accuracy are analyzed.

Finally, the top important K genes are listed and their expression levels in different samples shows clear separation between the two classes, and their descriptions in NCBI and GeneCards websites confirm that their mutation cause cancers.

❖ References

- **Li2008_Article_GeneSelectionUsingGeneticAlgor.pdf**
- **Yang-Honavar1998_Chapter_FeatureSubsetSelectionUsingAGe.pdf**
- **Lessmann, Stahlbock, Crone (2006) Genetic Algorithms for Support Vector Machine Model Selection - WCCI06 01716515.pdf**
- **https://www.cancercenter.com/cancer-types/breast-cancer/symptoms**
- **https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-7-3**
- **https://www.ncbi.nlm.nih.gov/bioproject/PRJNA107401**

- Kyeong Eun Lee, Naijun Sha, Edward R. Dougherty, Marina Vannucci, Bani K. Mallick, Gene selection: a Bayesian variable selection approach, Bioinformatics, Volume 19, Issue 1, January 2003. 10.1093
- Ramón Díaz-Uriarte, Sara Alvarez de Andrés BMC Bioinformatics. 2006; 7: 3. Published online 2006 Jan 6. 10.1186
- Shutao Li , Xixian Wu , Xiaoyan Hu College of Electrical and Information Engineering, Hunan University, Changsha, Hunan, People’s

Republic of China. Published online: 22 January 2008. 12:693–698

- **Introduction references:**

https://study.com/academy/lesson/genetic-selection-definition-pros-cons.html#:~:text=Genetic%20selection%20is%20the%20process,and%20come%20in%20various%20forms.

https://www.sciencedirect.com/topics/biochemistry-genetics-and-molecular-biology/genetic-selection

https://course.oeru.org/ipsy103/learning-pathways/evolutionary-psychology/gene-selection-theory/

https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0212333#:~:text=Gene%20selection%20is%20to%20select,%2C%20wrapper%2C%20and%20embedded%20methods.

https://bmcmedgenomics.biomedcentral.com/articles/10.1186/s12920-018-0447-6

https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-7-95

- **Design of chromosome references:**

https://www.sciencedirect.com/topics/engineering/chromosome-representation

https://pdfs.semanticscholar.org/7af8/2b50abf32c7620f0437c0dfb7af7443180e4.pdf

http://biology.kenyon.edu/slonc/bio3/AI/GEN_ALGO/gen_algo.html#:~:text=AI%3A%20Genetic%20Algorithms&text=Genetic%20algorithms%20provide%20co

mputers%20with,upon%20implementations%20of%20evolutionary%20processes.&text=The%20computer%20program%20first%20creates,then%20tests%20their%20%22fitness%22.

- **Fitness function references:**
  https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_fitness_function.htm#:~:text=The%20fitness%20function%20simply%20defined,it%20should%20be%20sufficiently%20fast.
  https://www.cs.cmu.edu/~schneide/tut5/node42.html#:~:text=Leave%2Done%2Dout%20cross%20validation,is%20made%20for%20that%20point.
  https://www.researchgate.net/figure/Diagram-of-k-fold-cross-validation-with-k-10-Image-from-Karl-Rosaen-Log_fig1_332370436

- **Selection references:**
  https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_parent_selection.htm
  https://arxiv.org/ftp/arxiv/papers/1203/1203.3099.pdf

- **Crossover references:**
  https://link.springer.com/article/10.1007/s11633-014-0870-x
  https://www.tutorialspoint.com/genetic_algorithms/genetic_algorithms_crossover.htm

- **mutation references:**

https://www.sciencedirect.com/topics/engineering/mutation-operator
https://www.geeksforgeeks.org/mutation-algorithms-for-real-valued-parameters-ga/
http://www.neurodimension.com/genetic/documentation/OptiGenLibraryforCOM/GeneticServer/Uniform_Mutation.htm
http://www.neurodimension.com/genetic/documentation/OptiGenLibraryforCOM/GeneticServer/Non-Uniform_Mutation.htm

- **replacement references:**

http://shodhganga.inflibnet.ac.in/bitstream/10603/32680/17/17_chapter%207.pdf
https://www.cs.unm.edu/~neal.holts/dga/optimizationAlgorithms/steadyStateGA.html
https://www.researchgate.net/post/What_is_meant_by_the_term_Elitism_in_the_Genetic_Algorithm#:~:text=Elitism%20involves%20copying%20a%20small,unchanged%2C%20into%20the%20next%20generation.&text=Candidate%20solutions%20that%20are%20preserved,remainder%20of%20the%20next%20generation.