Faster Markov Blanket with Tabu Search For Efficient Feature Selection of Microarray Cancer Datasets

By

Hardik Ghevariya

A thesis submitted in partial fulfilment
of the requirements for the degree of
Master of Science (MSc) in Computational Sciences

The Faculty of Graduate Studies
Laurentian University
Sudbury, Ontario, Canada

TABLE OF CONTENTS

TOPIC **PAGE NO.**

List of Tables

List of Abbreviations

Deoxyribonucleic Acid	(DNA)
k-Nearest Neighbor classifier	(k-NN)
Minimum Redundancy Maximum Relevance	(mRMR)
Naïve Bayes	(NB)
Support Vector Machine	(SVM)
Genetic Algorithm	(GA)
Differential Evolution	(DE)
Deoxyribonucleic Acid	(DNA)
Markov Blanket	(MB)
Learning Tabu Search	(LTS)
Directed acyclic graph	(AUC)
Ant Colony Optimization	(ACO)
Markov Random Fields	(MRF)
Artificial Neural Network	(ANN)
Machine Learning	(ML)

Abstract

In the field of medical science, particularly for diseases caused due to genetic reasons, the proper classification of genes is necessary to prescribe a cure for the same. Genes are required to be classified as per any particular characteristic that influences the cancer. Feature selection methods have been recognized as being important in this domain. This process assumes more importance wherein datasets containing a large number of variables (genes) are considered. In this thesis, we focus our work on the Tabu search technique combined with the Markov Blanket algorithm for feature selection and classification of microarray gene expression data. The HITON implementation of the Markov Blanket algorithm for feature selection was implemented and compared with the HITON plus Tabu search method on microarray gene expression datasets. We propose HITON with Tabu search for feature selection algorithm to obtain high classification performance for high dimensional microarray cancer datasets. Higher accuracy was achieved by applying Tabu search with HITON algorithm when tested with three classifiers - KNN, SVM, and NN. The proposed algorithm HITON + Tabu with SVM achieved 99.40% classification accuracy for the Prostate dataset, whereas HITON with SVM gave 98.04% classification accuracy, an increase by 1.36% accuracy using HITON + Tabu algorithm. For Leukemia dataset, HITON + Tabu with KNN achieved 99.83% classification accuracy, whereas HITON with KNN gave 98.81% classification accuracy, an increase by 1.02% accuracy using HITON + Tabu algorithm. For the Lung Cancer dataset, HITON + Tabu with SVM achieved 92.36% classification accuracy, whereas HITON with SVM gives 89% classification accuracy, an increase by 3.36% accuracy using HITON + Tabu algorithm. In addition, the proposed algorithm can be generalized to solve various other optimization problems.

Keywords: Feature Selection, Microarray Data, Markov Blankets, Wrapper Methods, HITON, Tabu Search, Fitness Function, Crossover, Mutation, Cancer Classification, Support Vector Machine, Neural Network, Gene Selection.

Acknowledgments

Foremost, I would like to express my sincere gratitude to my supervisor Dr. Kalpdrum Passi for the continuous support of my Master's study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me at all times of research and writing thesis. Dr. Passi's office was always open whenever I ran into a trouble spot or had a question about my research. He gave me the confidence to complete this work.

Besides my supervisor, I am deeply grateful to all the staff of the Computational Science program. Last but not least, I would like to thank my family, my parents Kantibhai and Vimlaben, and brother Rajni for supporting me spiritually throughout my life.

I also extend my hearty thanks to all those who have directly or indirectly helped me in the adventurous journey of my research.

Thank you.

Chapter 1
Introduction

1. Introduction to gene expression and Microarray Technology

Microarray is one such technology that allows investigation of non-traceable issues and offers new apparatuses that change the conduct of scientific experiments. The preferred primary standpoint of microarray advancements is one of scale in contrast with customary strategies. Rather than experiments dependent on results from one or a few genes, microarrays permit a great many genes or entire genomes to be analysed at the same time.

Microarrays are microscope slithers containing the ordered sample series like DNA, RNA tissue and protein. The study of genes relies on the investigation of a large amount of data that have been obtained from various biological tests [1]. A particular type of such study involves scrutinizing the features of a large number of genes at the same time under a particular set of conditions. The study of such magnitude is made possible due to microarray techniques where the magnitude of data produced is huge. This technology is the latest addition to techniques used in such studies and has become one of the most important methods used by biologists to study the characteristics of genes in an organism. A microarray is a solid surface wherein the DNA are fixed at particular spots in a certain particular pattern. These patterns are then used to study gene characteristics using various methods [2].

In order that different types of diseases are treated properly, it is important to have a correct diagnosis of the disease. Knowledge of inter and intracellular activity is key to this. It therefore requires that the intricate metabolic processes are properly understood before trying to develop methods that would be able to classify ailments and provide necessary medicines that would eliminate the problem. DNA microarray technology is a promising method that can manifest the features of large volumes of genes present in an organism [3]. The technique can concurrently investigate and detect a large number of

genes through quantification of gene sequencing, hybridization, and DNA transcription. Gene expression microarray also finds application in choosing genes with increased or decreased functions [2]. Although every individual has identical genes, those present in organs prone to diseases show certain different features than genes found in normal organs. This feature is referred to as gene expression. Study of the different gene expressions and classifying them to represent various diseases together with the identification of those genetic expressions within an organism is where biomedical research is thriving.

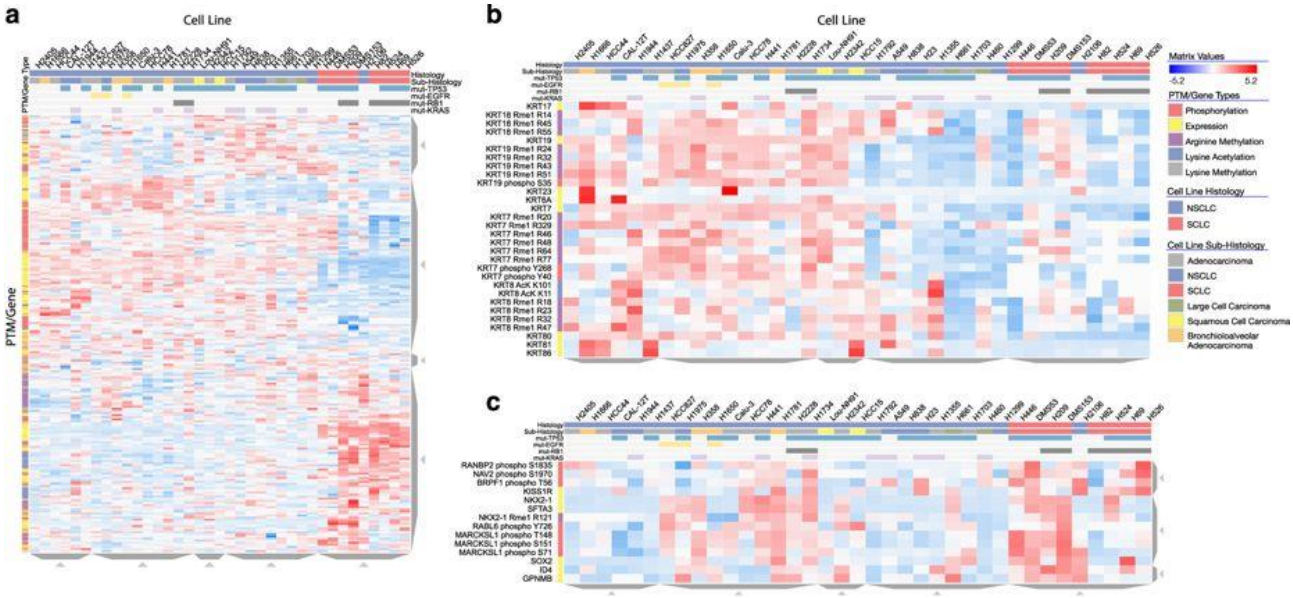

Figure 1.1: Shows the clustering of lung cancer lines using Microarray data [1]

Figure 1 describes the Lung cancer cell lines (columns) clustered based on a combination of Post-translational modifications (PTMs) and Messenger Ribonucleic acid (mRNA) expression data [1]. The microarray type depends on the material on the slide: DNA, DNA microarray; RNA, RNA microarray; protein, a microarray of protein; tissue, a microarray of tissue. As the samples are organized in a systematic way, records from the microarrays can be found back to either sample. This means that microarray genes are traceable. The number of samples ordered can be hundreds of thousands on a microarray.

During studies concerning genetic expression, students of biological research aim at finding similarities and dissimilarities present within the genetic composition. Similar genes would imply that the genes adhere to the same set of principles and have a probability of performing the same set of functions. Dissimilar genes would however not follow the conditions above. If various genes are matched, one can derive an idea regarding the features of genes in different circumstances [2].

The thousands of genes represent high-dimensional data. Out of which, few are more important and few less important for a specific experimental work. More stress is therefore needed to extract more relevant genes from data on gene expression. DNA microarray classification suffers from serious impediments, which include the presence of a relatively small number of samples compared to a large volume of genes present in the microarray dataset. This is also referred to as the curse of dimensionality. Feature selection approaches in DNA microarray datasets allow for the removal of such unnecessary and unrelated characteristics from any microarray dataset. This technique provides a solution for the most critical problem crippling biomedical research [2].

Microarrays of DNA are used to identify:

1. The level of expression of genes in a sample often referred to as expression profiling.
2. The gene sequence in a sample, commonly referred to as mini sequencing for short nucleotide reading, and the mutation or Single nucleotide polymorphisms analysis for single nucleotide reading [4].

1.1 Workflow summary of Microarrays

Microarrays can be utilized from numerous points of view to gauge gene expression, yet a standout amongst the most well-known applications is to think about the declaration of an arrangement of cell genes kept up in a specific (condition A) with a similar arrangement of genes from a reference cell kept up under typical conditions (condition B). A microarray is usually a glass slide on which DNA molecules at different functional locations called spots (or features) are fixed in an orderly manner [4].

A microarray can contain thousands of spots and each spot can contain several million copies of similar DNA molecules that are unique to a gene. Microarrays can be used in many ways to measure gene expression, but one of the most popular applications is to compare the expression of a set of genes from a cell held in a specific condition (condition A).

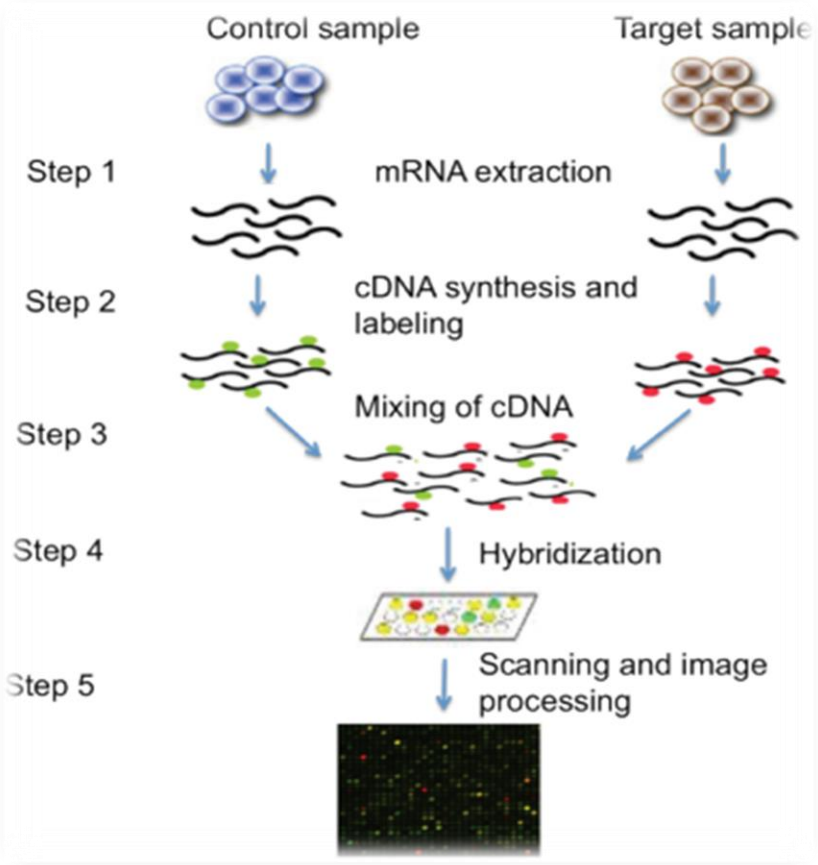

Figure 1.2: Overview of Microarrays Experimental Steps [5]

Figure 1.2 provides an overview of the experimental steps involved. RNA is extracted from the cells first. Next, the extracted RNA molecules are reversed to cDNA using an enzyme reverse transcriptase and nucleotides labelled with various fluorescent colours. For example, cDNA from cells grown in condition A can be marked with red colour and green colour from cells grown in condition B. Once the samples have been labelled differently, they can be hybridized on the same glass slither. Any cDNA arrangement in the example is hybridized to explicit spots on the glass slither containing its correlative sequence at this point. The measure of cDNA connected to a spot is straightforwardly relative to the quantity of RNA particles present in the two examples for this gene [4].

In the case where gene was expressed in the two conditions to a similar extent, the spot would be yellow; the spot would be green if particular gene is highly expressed in normal cell; the spot would be red if particular gene is highly expressed in diseased cell and the spot would be grey if the gene were not expressed in the two conditions as seen in Figure 3. Toward the finish of the exploratory stage, accordingly, a picture of the microarray is seen, in which each spot comparing to a gene has a related fluorescence value demonstrating to the comparative expression level of that gene.

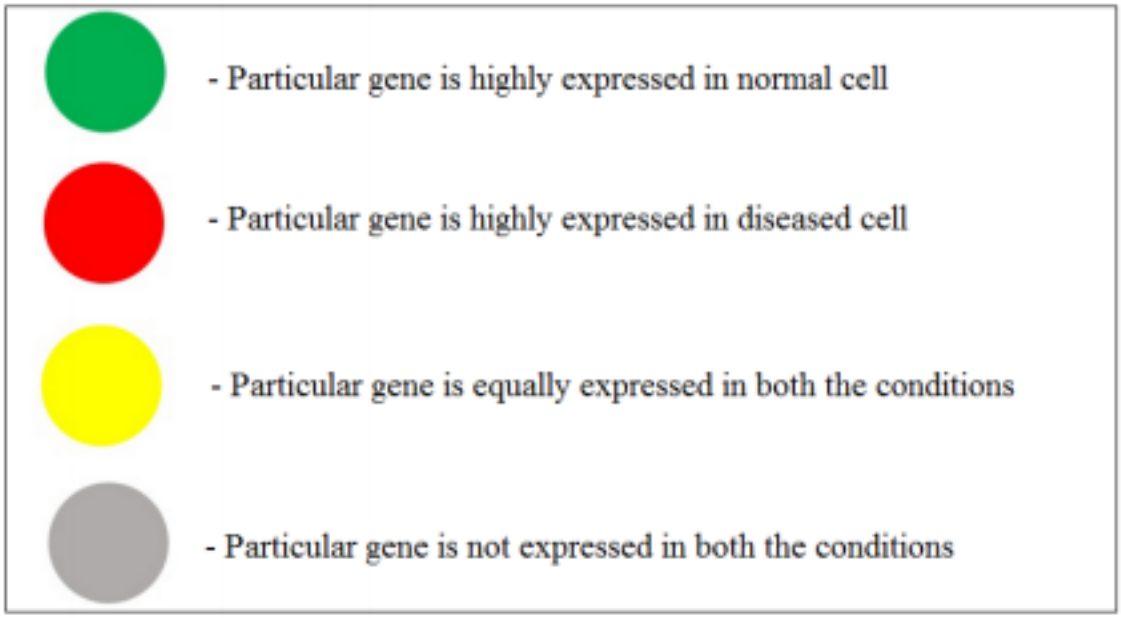

Figure 1.3: Expression level of genes [6]

1.1.1 DNA Microarray: Advantages and Disadvantages

Microarrays that are commercial are of high quality, good density and are available for the most widely studied organisms including humans, mouse, rats and yeasts. A DNA microarray is a study in which thousands of genes can be compared at once. In science, this technology has reduced a lot of time, but now researchers need to think about the process of extracting meaningful information from the raw data using microarray data.

Advantages of the DNA Microarray: [3]

- Provides information for a huge number of genes
- Different fragments of DNA can be utilized to think about gene expression
- A great step forward to finding solutions for cancer and other diseases
- One experimentation rather than many
- Fast and simple to get results

Disadvantages of the DNA Microarray: [3]

- Just on the grounds that mRNA is "turned on" doesn't mean proteins are made
- Very little information is accessible about numerous genes
- Will the discoveries prompt exploitative medical methodology?
- Scientists have no institutionalized method to disseminate results
- Correlations in results don't mean causation

1.2 Introduction to Bioinformatics

Bioinformatics is a science and focuses on understanding biological data using information technology. The development of bioinformatics dates from the 1960s. It was consistent with the development of protein sequencing methods from a range of organisms and the availability of protein sequences after the insulin sequence was determined by Frederick Sanger in the early 1950s [7]. Research on bioinformatics received a boost when machine learning techniques were utilized for the same. These techniques possess various properties which include adjustment and ability to locate faults. Machine learning techniques classify the network and are required to study and adjust itself to changing conditions, thereby enhancing the efficiency of the machine. Machine learning techniques comprise a variety of processes including artificial neural networks, genetic algorithms, and fuzzy systems besides other processes which are a combination of two or more processes [8]. The utility of machine learning methods is in training the network to give accurate results and also to eliminate

negative values. Machine learning has been defined as the capacity of the computer to enhance its results in relation to past performances.

The Dayhoff team successfully organized protein sequences into separate groups and subgroups on the basis of sequence similarity and a point accepted mutation (PAM) matrices [9]. This was published as protein sequences atlas [10], which was widely used in alignment of protein sequences and searches for database similarity [11]. These were the pioneering methods of alignment of protein sequences and molecular evolution [10].

To evaluate and compare a large number of protein sequences of different organisms, innovative software methods are required because the manual handling of many amino acid sequences was unfeasible. This led Margaret Oakley Dayhoff and her colleagues at the National Biomedical Research Foundation [11] to compile the first “Protein Information Resources” [12].

1.2.1 Applications of Bioinformatics

Bioinformatics is the science in which biological sequences and molecules store, extract, organise, analyse, interpret and use information. It was mainly driven by progress in the sequencing and mapping of DNA. In recent decades, rapid developments in genomics and other molecular research technologies and developments in information technologies have united to produce a great deal of molecular biological information. Bioinformatics' primary goal is to improve understanding of biological processes [2].

Genome Annotation:

Annotation is the process of marking the genes and other biological characteristics in a DNA sequence in the context of genomics. Dr. Owen White designed the first genome annotation software system in 1995 [13].

Genome annotation is the process of sequence attachment of biological information. It comprises of three key steps:

1. Recognizing parts of the genome that don't code for proteins.
2. Distinguishing components on the genome, a procedure called gene prediction.
3. Assigning organic data to these components.

Gene Expression Analysis

The expression values of numerous genes can be dictated by measuring mRNA levels with various strategies, for example, microarrays, expressed cDNA sequence tag sequencing, serial analysis of gene expression tag sequencing, massively parallel signature sequencing (MPSS), or different uses of multiplexed in-situ hybridization and so forth. All these techniques are highly noisy and subject to biological measurement bias [13]. In high throughput gene expression studies, the main research area involves the development of statistical tools to separate data from noise.

Mutations Analysis in Cancer

The affected cell genomes are rearranged in cancer in complicated or even random ways. Massive sequencing attempts are employed in many cancer genes to identify previously unknown point mutations. By creating advanced automated systems and developing new algorithms and computer programs to compare the sequencing results with the increasing collection of human genome sequences and germ line polymorphisms, computational biology endure achieving the sheer volume of generated sequence data. The study of lesions found to be recurring in many tumors is another type of data that involves the development of new information technology [14].

1.3 Biological Data Mining

Biological data mining includes gene discovery, protein function area detection, function motive detection, protein function deduction, disease diagnosis, disease prediction, disease treatment streamlining, protein and gene interaction network reconstruction, subcellular protein data cleaning and prediction [15].

Machine learning can be used to classify the peptide by mass spectroscopy. To reduce stochastic variations in peptide detection by searching for databases, the correlation between fragment ions in a

tandem mass spectrum is important. It is highly desirable to have a proficient scoring algorithm that takes into account the correlative information harmoniously and comprehensively [16]. The two key "high-level" data mining objectives are forecasting and explanation in practice. The key tasks for data mining are classification, estimation, prediction, association, clustering, description & visualization [15].

Classification: Classification is a process where a knowledge object is mapped (categorized) into one of several predefined groups.

Estimation: Some input data give an unspecified continuous variable value.

Prediction: Same as classification except that the values are classified according to some future behaviour or future value predicted.

Association: It is also called dependency modelling to decide how things work together.

Clustering: Population segmentation into a number of subgroups or groups.

Description and visualization: Data representation using visualization methods. Data learning is divided into two categories: directed ("supervised") and undirected ("unsupervised").

The first three functions of classification, estimation and prediction are examples of supervised learning. In unsupervised learning, no variable is identified as the target; the aim is to establish a relationship between all variables [15].

Advancing new tools for data processing and knowledge disclosure is a subject of dynamic research. One inspiration driving the advancement of these tools is their potential application in current molecular science [15].

1.4 Feature Selection methods for better prediction

Feature selection algorithms can reduce the number of features to construct a machine learning model by checking various attribute combinations in a dataset. Feature selection is a common terminology in the realm of mathematical and statistical science. Feature selection involves the process of eliminating unnecessary or redundant features from the parent data set. Data generated from various sources contain a host of attributes. This is a significant drawback in the process of classifying the

data. The presence of irrelevant data tends to interfere in the training process of any algorithm and can give invalid output. This makes the study of the results very difficult and may lead to the wrong diagnosis. As this study would reflect upon, a feature selection procedure helps in the correct assessment of data leading to drawing correct conclusions. There are various feature selection methods, common among these being principal component analysis (PCA), Correlation Feature Selection (CFS), Gain Ratio, Relief, non-negative matrix factorization (NMF), linear discriminant analysis (LDA), auto-encoders, etc.

In our research, we have explored the use of Markov Blanket for feature selection and optimizing the predictive power of the classifiers on cancer microarray data. A well-known implementation of the Markov Blanket called the HITON [17] was studied. To further enhance the performance of the HITON algorithm, a hybridized algorithm is proposed in this research by combining HITON with Tabu Search for finding the best features affecting the gene expression on large cancer microarray datasets. The results obtained have been summarized in relevant sections, and a comparison has been made to get an idea of the performance of individual methods with a view to upscale the same for future use. Experimental results show an improved performance of prediction of gene expression in five microarray cancer datasets by the proposed algorithm.

1.5 Classification Techniques

Gene expression is predicted by applying various classifiers on the subset of features selected using feature selection techniques. Once the most predictive and informative genes are selected using our proposed HITON (implementation of Markov Blanket algorithm) with Tabu search (HITON + TABU) algorithm, we input those selected genes into a classifier to get more accurate prediction results. There are two main goals of microarray studies. The first is to identify molecular signatures associated with known classes and second is to discover new classes. To achieve these goals in pattern recognition [18], two different methods have been suggested: Supervised methods and unsupervised methods. Supervised methods of analysis are predominantly used to identify the differences at the molecular level between the known classes and to diagnose or predict the class of the new sample.

This technique is also known as classification. In unsupervised methods, data is organized without any knowledge of external classification information. The data is organized into different groups such that samples within the group are similar and differs from samples in the other group in some sense. It can be used for discovering new classes within a given data. This technique is also known as clustering. As a part of this study, we are using the classification techniques to compare the differences among them. The classification techniques used in this research are:

1. K - nearest neighbour (KNN)
2. Support Vector Machine (SVM)
3. Neural Network (NN)

Classification is a supervised learning approach for a division of multivariate data into various sources of populaces. It has been playing a substantial role in bioinformatics by class prediction or pattern identification from datasets. Support Vector Machine (SVM) is a powerful tool for the classification of genomic cancer. Guyon et al. [19] used SVM for the ranking of features to determine the significance of a feature by its weight. In the field of gene selection, SVM- RFE (Recursive Feature Elimination) has an important role, as iteratively eliminates less significant genes by considering the weights of the support vectors. Zhang et al. [20] suggested the construction of a classifier distance matrix in order to ease the time convolution of wrapper-based subset selection (FSS). The proposed method periodically updates the matrix to calculate the relevance criteria of each feature. The authors have embedded the classifier K-Nearest Neighbour (K-NN) with a subset selection based on the Wrapper [20]. The main genes can be identified by extracting important samples that can be classified by neural network or SVM. Bishop et al. [21] proposed a method known as Neural Networks for Pattern Recognition technique to improve the accuracy of classification.

1.6 Objectives of the Study and Outline of the Thesis

The main objective of this research is to increase the prediction accuracy by the enhanced Markov Blanket algorithm for feature selection on cancer microarray datasets. Five cancer microarray datasets were used for the experiments: Lymphomas, Prostate, Leukemia, Brain Tumor, and Lung Cancer. The proposed hybridized HITON algorithm with Tabu search has been compared with HITON algorithm for better prediction accuracy with three different classifiers. The three different classifiers used in this research are K - nearest neighbour (KNN), Support Vector Machine (SVM), and Neural Network (NN). The algorithms were implemented in MATLAB.

The thesis is organized as follows:

Chapter 2 presents the literature survey highlighting some existing literature on microarray data fields.

Chapter 3 discusses feature selection techniques and optimization methods in detail.

Chapter 4 introduces the Markov Blanket algorithm and the proposed enhancement for efficient feature selection.

Chapter 5 presents machine learning techniques and tools.

Chapter 6 presents the results and discusses the performance of the proposed algorithm, HITON + Tabu search with that of HITON for the five cancer microarray datasets using three classifiers.

Chapter 7 presents the conclusions and discusses the future work that can be performed.

Chapter 2

Literature Review

Of late, there has been a significant amount of study in the field of feature selection. In order to model gene and protein interactions and build a knowledge base of biological processes, the analysis of different data sources is required. Microarray data analysis or Microarray gene expression analysis enables the most significant genes for a target disease and gene group with related patterns to be identified under diverse experimental circumstances. The choice of features with classification or clustering algorithms is the most commonly used approach to deal with these problems. The basic principle of these techniques is to use computer resources to solve problems on scales of magnitude that are far too large for human judgement. Computational biology research often overlaps with system biology. In this field, major research efforts include alignment of sequence, gene finding, genome assembly, alignment of protein structure, prediction of protein structure, gene expression prediction and protein-protein interactions and evolution modelling. The enormous amount of data in these areas of research makes the use of data mining and artificial intelligence very exciting and promising. From many sources, such as the results of high throughput experiments or clinical records, these techniques aim to reveal previously unknown knowledge and relationships [22].

The primary motive of feature selection lies in decreasing the operational hardships with no negative impact on the performance by way of removal of unnecessary data [23], [24]. Stress is laid to prepare a classification based upon high levels of necessary data and low levels of less important ones. High-level classification of necessary data and low-level classification of less important ones can be achieved through specific algorithms like minimum redundancy maximum relevance (mRMR) algorithm [25] and fast correlation-based filter (FCBF) algorithm [26]. Various sources of information became available late. For example, DNA microarray studies, produce a huge number of gene expression and provide a clear method for collecting significant knowledge measurements in a short span. They are used to gather tissue and cell data on variances of gene expression. They are

utilized to gather data from tissue and cell with respect to gene expression variances. Techniques trusting on gene expression profiles are more objective, exact and reliable if it is based on the morphological appearance of the Tumor Compared to standard methods of Tumor diagnosis [27].

Analysing the most important parts of research papers and performing on-demand data integration to impart new knowledge and validation purposes in many biological studies is a key Problem. Given the huge amount of information, having improved mechanisms to effectively identify and represent the most relevant parts of textual documents has become increasingly important. In order to model gene and protein interactions and build a knowledge base of biological processes, it is important to examine different data sources. Microarray data analysis or Microarray gene expression analysis enables the most significant genes for a target disease and gene group with related patterns to be identified under diverse experimental circumstances. Choosing features with classification or clustering algorithms is the most common approach used to address these issues. Clustering is a valuable exploratory method for data on gene expression as it bunches similar objects together and helps the scholar to recognize potentially important gene connections. The genes of the same cluster typically perform related functions and are often co-regulated. Similar genes can, therefore, be grouped in order to understand the functions of genes for which information has not been beforehand accessible. Another use of clustering algorithms is to recognize the group of redundant genes and then select only one representative to reduce dimensionality for each group [4] [28].

The aim of this research is to exploit optimization methods and classification techniques on gene expression data to answer explicit biological issues, i.e., i) recognizing the best significant genes associated to a target disease, and ii) combining genes together which indicates a similar behaviour under different conditions. Especially, a new HITON (implementation of Markov Blanket) with the Tabu search technique has been created to distinguish the most significant genes and to enhance the exactness of forecast models for test characterization. Also, we apply HITON with the Tabu search method to new datasets and validate the results.

“Markov Blanket” was first introduced by Koller and Sahami [29] for feature selection defined by feature relevance. As far as the class attribute is concerned, an input function can be categorized as a highly relevant, irrelevant redundant or non-redundant feature, and a Markov blanket should include highly relevant and non-redundant features [30]. Although, using the specific feature to evaluate Markov blankets accurately is very difficult due to the noisy data and limited sample size [29]. To address limited sample size and noise in data issues, Tsamardinos and Aliferis [17] provided theoretical results relating feature selection principles with Markov blankets in Bayesian networks. Their theoretical results have shown that if a probability distribution can be faithfully represented by a Bayesian network, then the class attribute Markov Blanket in the Bayesian network is not only unique but also provide the solution for selecting features [17]. Proposal for the creation of a framework for a culmination of features representing unnecessary data has been put forward by Yu et al. [31]. The other noteworthy advancements in this field include grouping together of two decades of work to a level of optimum probability by Brown et al [32], linkage of Markov blankets in Bayesian networks with requisite characteristics as enumerated is followed by transfer of the same leading to Markov blanket spread over Bayesian network has been done by Tsamardinos and Aliferis [17]. The aftermath of this study regarded the Markov blanket selection method as the best in filter selection methods.

The development of the first sound algorithm and grow/shrink (GS) algorithm is attributed to Margaritas and Thrum [33]. The motive behind this method is finding Markov blanket for enhancement in the speed of the global Bayesian network. GS algorithm relies upon the large size of occurrences and is found unsuitable for sets having lower volumes of data. This impairment in the GS algorithm has been rectified when Tsamardinos and Aliferis [17] developed the IAMB (Incremental Association Markov blanket) algorithm as a new form of the GS algorithm. This method, besides being more efficient than its predecessor, provides a reliable way to find the Markov blanket. IAMB (Incremental Association Markov blanket) method still suffers from reducing the number of conditional independence tests. To do away with the problems resulting from the nature

of data, HITON-MB (Markov Blanket of Target) and MMMB (Max-Min Markov Blanket) models have been developed [17]. The variable selection method used for classification, regression and prediction is called HITON. The HITON-MB returns Markov blanket of any target node in a faithful Bayesian network. HITON-MB and MMMB adopt a two-pronged approach to arrive at the Markov blanket. The first approach tries to identify the parent and the offspring of the target, while the second finds their spouses. This job was further simplified by the following algorithms HITON-PC (Parent-Children, only discovering parents and children of a target feature) and MMPC (learning technique to select the correlated features for each strongly relevant feature). MMMB was also successful in paving the way for PCMB (Parent-Children Markov Blanket) to cater to the problem. The underlying characteristic of these algorithms lies in their ability to select one Markov blanket, assuming that the probabilistic distribution adheres to the Bayesian network.

Pena et al. [34] came out with a stochastic Markov blanket algorithm, also known as KIAMB (stochastic Markov blanket algorithm), which if allowed to run a number of occasions can give the desired results and KIAMB can find many Markov blankets by using a stochastic heuristic search. Considering recent developments, Statnikov et al. [17] Developed the target information equivalence (TIE*) algorithm having the ability to explore all Markov blankets in the distribution of data. This algorithm outsmarts KIAMB. TIE* utilizes the method of referencing the findings of many Markov blankets and does not miss any variables. However, with the growing number of features, Markov blankets would grow, making TIE* complicated.

Xue Bai [35] suggested the Tabu search method for feature selection. This method, when compared with other methods, showed better results within the scope of giving the best solution at the shortest possible time. Results obtained through the integration of a Tabu search algorithm and learning mechanism have been studied by Hongbin Zhang [36]. In his research, a Tabu search algorithm specially built for the solving optimal feature selection problem. Results show that learning Tabu search fared better than its classical counterpart and Tabu search not only has a high chance of obtaining the optimal solution or close to it, it also requires less computational time than branch and

bound method [36]. The reason attributed is the use of estimation function, which restricts several processes leading to shorter run time. Oduntan et al. [37] went a step further when he designed a new feature selection method - multilevel feature selection which finds use in medical data using a multilevel search method. The performance of this technique is put to the test in terms of the nature of its solution by comparing the results with other known algorithms. To arrive at a common basis for comparison, studies have been conducted using equal volume and variety of data. The multi-level feature selection technique combines a hierarchical search framework with a search method. The framework, which allows the methodology to be easily adapted to different forms of biomedical databases, consists of increasingly coarse versions of the original set of features systematically explored by the search method. This algorithm creates the most favourable subsets, thus enabling a high level of accuracy in classification as compared to other methods [37].

The year 1986 witnessed the development of Tabu search [38], which proved to be regarded as the best local search algorithm with a capacity to find solutions even with large and varied data. The functional scope of this algorithm is in finding preliminary results and further assessment of the fitness values of these results. A realistic set of solutions can then be found through minor changes in the primary results. These minor changes are referred to as moves. The principle of this method is to include the best values of the neighbours in the list of current solutions if it does not form a part of the Tabu list. The function of the Tabu list is maintaining a record of solutions searched in the past so that TS does not require to consider them for any further searching, thereby eliminating filling up of space. The purpose of creating a movement is to enhance variety without taking into consideration the quality of solutions. Simultaneously, the Tabu list also incorporates the best local neighbours so that the same do not pass through the process of future scrutiny. When a Tabu list is developed using such realistic solutions, these are subjected to evaluation. The excellent results obtained will form a part of future results. This loop culminates when stopping standard is achieved.

In Classical feature search algorithms like Tabu search and the dataset has a large volume of data with a large number of attributes, which implies that the entire neighbourhood can’t be studied at

each repetition. To overcome this problem, David Schindl and Nicolas developed the Learning Tabu Search (LTS) [39]. In this method, the neighbourhood is searched in two stages - (i) estimation of the quality of all neighbours and (ii) checking the Q best neighbours. The basis of LTS lies in estimating the quality of neighbourhood results. The idea behind this process is the fact that if certain characteristics are found to be a part of the best solutions, these features require to be an integral part of finding future solutions. The characteristics mentioned above are calculated from the results where such combinations are found as such LTS needs sufficient memory to store such good combinations for future reference. This memory is the defining attribute of LTS. The process is similar to ant colony optimization (ACO) algorithms [40]. Trail value is defined as the quality of any such amalgamation. Trail value assumes importance due to the fact that trail value is in direct proportion to the quality of such combination. In tune with ACO, updating of memory is necessary to enhance quality combinations and eliminate bad combinations. Elimination of combinations of inferior quality is done using the process of evaporation. The process of updating, in LTS, is done periodically and referred to as cycles. In each such cycle, the best solution provides the basis for updating of trail values. The dimension of such cycles has a direct bearing on how a learning process performs.

Chapter 3

Feature Selection Methods

3.1 Feature Selection

The advent of new technology has ushered in an era where information is at the fingertips. The recent years are witness to the fact that huge data is being generated globally at a rapid rate. These data also possess the property of variation. This makes the analysis of data very difficult, which results in no viable conclusion. In order that the vast volume of varied data is processed in a quick time, artificial intelligence is used. Machine learning, which a component of artificial intelligence, tries to create algorithms that can reduce the burden of analysing these data. Designing a robust algorithm to do away with all bottlenecks requires the removal of all unnecessary data. Feature selection, as the name suggests, is a method employed to get rid of unnecessary data which can prove detrimental in generating results due to the curse of dimensionality. The curse of dimensionality was coined to indicate that the number of samples needed to estimate an arbitrary function with a given level of accuracy grows exponentially with the number of variables (i.e., dimensions) that it comprises [41]. The methodology used is based on defining certain features and classifying data based on these features. Feature selection results in the most relevant features for classifying the data to the most accurate labels. This process not only reduces the run time of such machine learning algorithms but also increases the accuracy of the results.

There are three types of feature selection algorithms namely filters, wrappers and embedded types. In filters, every characteristic is gauged through a process that is not dependent on the classifier, and the superior characteristics are taken [41], which makes results more accurate. The wrapper technique compartmentalizes the characteristics and checks the output of the classifier for each compartment and thereafter selects those compartments which have the best output. This makes wrappers rely on classifiers. In spite of the fact that compartmentalizing is a difficult task [42], wrappers can be depended upon. However, wrappers have a large runtime which tends to increase calculation cost,

but the same can be decreased by using certain experimental algorithms. Regarding the embedded process, it is regarded as a process that uses the best of both filter and wrapper techniques. Feature selection is performed by embedded processes in the same way as artificial neural networks. Various studies have been conducted on the scope of these techniques [43]. Results show that the filter method is preferred over wrapper in many instances.

3.2 Types of feature selection techniques

As enumerated above, feature selection involves a reduction in the facets of data, thereby reducing unnecessary features while doing an analysis of data. This makes predictions more accurate and reliable. This aspect assumes importance in the light of the fact that machine learning mechanisms follow the policy of 'garbage in – garbage out.' Needless to mention that results or output would depend on the quality of data fed into the model. As such, it is important to feed the model with specific and relevant data.

Various studies have been conducted to design a technique that gives the best result in minimum time at a low computational expense. The various types of feature selection techniques that are generally being employed have been classified as a filter method, wrapper method, and embedded method though other methods are also used. This study restricts itself to the three former techniques. The features of each method, along with their importance, have been detailed below.

3.2.1 Filter Method

This method establishes itself on the relation between the features of any data and its class attributes. The characteristics of each data are ranked on the basis of its importance to the class attributes. The selection of data to be fed into the model depends upon it attaining a minimum value [44] which is predetermined by the user. Various filter algorithms are used in feature selection including chi-squared test [45], One-R [46], Relief-F [47], Gain Ratio [48], Information Gain [49], Naïve Bayes [50], Multilayer Perceptron (MLP) [51], J48 decision tree [8], etc. The accomplishment of the classification procedure on the result of a feature selection process renders the prediction to be more

accurate and saves time besides reducing the use of memory and processor and making the data understandable.

Advantages [52]

- It can easily scale to very large datasets.
- This method is simple, computer fast and independent of an algorithm for classification.
- The choice of genes should just be completed once, and after that diverse classifiers can be assessed.

Disadvantages [52]

- They ignore the classifier interaction.
- They often vary uniformly or slightly.

3.2.2 Wrapper Methods

This is one of the better alternatives in the feature selection process as it takes into consideration specific preconceptions of the algorithm. The wrapper methods gauge the performance of data based on certain pre-designed machine learning algorithms. The modus operandi of this process lies in selecting characters that are considered best for the machine learning algorithm and tries to improve its performance. For algorithms having low complexity, wrapper methods need to run a large number of processes resulting in huge calculation cost. The wrapper selection method can be used over a set of data using various classification algorithms. Studies suggest that with Naïve Bayes, the wrapper method fares better. The same conclusion can also be drawn with Bagging. In general, we can say that you have to find answers to the following in order to build a wrapper algorithm [54]: How to find out all possible subsets of features/genes? How can we be satisfied with our classifier's classification performance to guide the search and when to stop the search? What predictor should be used? Although a thorough search can be carried out in the cases, the number of genes is not too large.

The wrapper is a blind search where we tend to find a subset and search randomly for the perfect subset that cannot be guaranteed unless all possible subsets are found. The problem with selecting a gene subset using a wrapper algorithm is therefore recognised to be NP-hard and the search with each iteration tends to be intractable to the user and sometimes takes hours to complete even on a fast computer [54] [55].

Wrapper selection methods can be seen in the following examples: forward feature selection, backward feature elimination, recursive feature elimination, and others [56] [57]. In this thesis the Markov Blanket was used which is a type of forward feature selection technique.

Forward feature selection: Beginning with no characteristics, this process determines the best characteristics and includes these in the set before the process is run. The process is repeated and further features are added.

Backward feature elimination: This process is the reverse of the previous process. In this process, all features are considered, to begin with, and after that, eliminating the bad ones.

A mixture of both forward selection and backward elimination: In this process, the model automatically ranks the features and adds the good ones while eradicating the bad ones.

Recursive feature elimination: In this process, the model runs with all the characteristics individually and analyses the results which forms the basis of its grading the characteristics into good and bad. It then retains the good characteristics of rejecting the bad ones.

In general, the wrapper method performs well even though these are relatively slower and less flexible. Considering the operating aspect, wrapper methods are more expensive. This is due to the fact that wrapper methods examine subsets of characteristics as defined by the algorithm. This can be understood by the fact that wrappers can predict the results of adding characteristics to any particular subset based upon which required additions may be decided. The benefit that wrappers have is the fact that wrapper methods get reliable results with smaller subsets.

The wrapper method shows the ability towards providing accurate choice even though there may be certain differences in their performance due to different classifiers. In spite of their superior performance, wrapper methods suffer from the problem of over-fitting. The over-fitting occurs when there is insufficient data to train the classifier and the data does not fully cover the concept being learned [58]. This can be dealt with using a variety of procedures.

Advantages [49] [50]

- It always chooses an almost perfect subset.
- This method's error rate is less compared to other methods.

Disadvantages [49] [50]

- The risk of over-fitting is higher than filtering techniques.
- Compared to other methods it is very computationally intensive.
- This is intended for the specific learning machine on which it was tested

The wrapper method comprises two phases, the decrease phase, which selects highly relevant features and the increase phase, adding weakly relevant features.

3.2.3 Embedded Methods

These methods are an amalgamation of both filter and wrapper methods. The algorithms, in this case, have built-in feature selection methods, e.g. LASSO, RIDGE [58], etc. Various embedded models have been launched over the last few years. The hallmark of these models lies in the fact that these models utilize regression methods as an optimizer whose losses are predetermined. The methodology employed has been broadly classified into various groups. One of the groups makes use of the L2 norm penalty, also called the ridge penalty, e.g. recursive feature elimination (RFE) is based on the L2 norm [59], [60]. The other group uses the L1 norm penalty, also known as the lasso penalty, e.g., feature selection using a solution of the L1 norm and penalized least-square on regression coefficients uses the L1 norm. There is yet another group that uses a combination of both L1 and L2 norm penalty,

e.g. feature selection using double SVM formulation [2], penalized least squares with an elastic net penalty. The fourth group uses the L0 norm penalty, e.g. feature selection using L0 norm formulation [60]. The use of other penalties like smooth clipped absolute deviation penalty is also known.

The Embedded methods tend to have a higher capacity than filter processes and are therefore more appropriate. With small amounts of training data, filter methods perform better as the number of training points increases. [61] [57]

Advantages [36-39]

- Computationally less intensive than methods of wrapping.
- The interaction with the classification model is included.
- They use the available data better if they do not need to divide the training data into a training and validation set.
- They achieve an answer quicker by avoiding the retraining of a predictor for each variable subset examined from scratch.

Disadvantages [36-39]

- Specific to a learning machine.
- Over-fitting problem compared to filters.

An important strategy for dealing with classification complexity can be solved by a mix of experts (MoE) who work with a common set of classifiers with expertise in various regions. Billy Peralta and Alvaro Soto [61] provided a regularized MoE variant that integrates an embedded process for the selection of local features using L1 regularization. Meihong Zhu et al. [57] submitted a feature selection criterion that can be applied to the multiple criteria linear programming classification using an embedded backward feature selection procedure.

3.3 Rank based Gene Selection Approach

SVM-RFE (Support Vector Machine - Recursive Feature Elimination) plays an important role in the field of gene selection, as it iteratively removes less important genes by taking into account the

weights of aid vectors. Due to the scarcity of microarray data, different methods produce different results. To boost gene selection capabilities and minimize software costs, it is possible to identify two strategies, such as (a) SVM-RFE weight redefinition [62], (b) SVM-RFE results with multiple gene removal policies. Because of the concluding gene set's previously known cardinality [63], the variants of SVM-RFE gene selection may lead to a question of subset bias. The rating score in the suggested method can be determined using statistical analysis. A Gene Ontology (GO)-based similarity test was used to verify the selected subset. Granular SVM is a granular computer-based learning system combining statistical learning theory and granular computing theory [64].

The main challenge in the field of gene expression data identification is the smaller sample number compared to the higher gene number. To address this problem, researchers have discussed various feature selection algorithms in the literature to reduce the number of genes. Recent studies include the use of semantic similarities between genes in GO to improve the selection of functions [65]. Christopher E. Gillies et al. [66] developed a new simulation that generates binary class data sets with differentially articulated genes amongst two classes.

3.4 Current Challenges in Gene Expression Data Classification

According to the above discussion, the challenges in the classification of gene expression data are discussed next.

Data from microarray experiments contain noise and are not very accurate. Therefore, before analysing the data, we need to use sophisticated noise omission and data normalization techniques [54]. Another major challenge is to create a distinct classifier to classify all types of gene expression data into a large number of classes.

The Microarray data set variables are classified with a huge number of mathematical techniques that helps to identify particular expression. To overcome this and obtain a strong performance of the classification much by selecting the minimal relevant subset features to identify cancer. Therefore, the future need is to train a decision network for identifying the utmost predictive and informative genes for gene expression data classification. Sadly, training Bayesian Networks is highly expensive

in terms of computation because in Bayesian Network each node (feature) needs to have prior knowledge of the network structure. In addition, Bayesian networks tend to have poor performance on high-dimensional data, which leads to finding the minimum relevant subset of genes intractable. Lastly, Bayesian network models can be difficult to interpret as well as require separating effects between different sections of the network [67]. From this point of view, we can use HITON (Markov Blanket algorithm) [17] for obtaining a strong performance of the classification much by selecting the minimal relevant subset features. Using the Markov Blanket algorithm, we can recognize the best significant genes associated with target disease and predict an early diagnosis of critical life-threatening diseases like Cancer by gene selection and classification.

“Markov Blanket” first introduced by Koller and Sahami for feature selection defined by feature relevance. Using the specific feature to evaluate Markov blankets accurately is very difficult due to the noisy data and limited sample size [29]. To address limited sample size and noise in data issues, Tsamardinos and Aliferis [17] provided theoretical results relating feature selection principles with Markov blankets in Bayesian networks. Their theoretical results have shown that if a probability distribution can be faithfully represented by a Bayesian network, then the class attribute Markov Blanket in the Bayesian network is not only unique but also provide the solution for selecting features [17]. Proposal for the creation of a framework for a culmination of features representing unnecessary data has been put forward by Yu et al. [31]. Tsamardinos and Aliferis [17] developed the IAMB (Incremental Association Markov blanket) algorithm. This method, besides being more efficient than its predecessor, provides a reliable way to find the Markov blanket. IAMB (Incremental Association Markov blanket) method still suffers from reducing the number of conditional independence tests. To do away with the problems resulting from the nature of data, HITON-MB (Markov Blanket of Target) and MMMB (Max-Min Markov Blanket) models have been developed [17]. The variable selection method used for classification, regression, and prediction called HITON.

3.5 Feature relevance and redundancy

In order to arrive at an accurate result, it is imperative that selected data should have attributes or features that are relevant. Irrelevant data leads to a chaotic situation that gives rise to wrong results. Feature selection is a mechanism that tries to search for data that are relevant while discarding redundant ones. Any data has the potential of being attributed to multiple features. Feature selection techniques basically try to compartmentalize these features into subsets. In this exercise, the difficulty lies in defining such compartments that can store information representing the entire class of data present in the parent data. This can be done by searching the required characteristics while eliminating those which are irrelevant [29]. To solve this difficulty, a large number of strategies and algorithms have been developed over the years. While filtering algorithms remove redundant variables, ranking methods tries to bring together relevant features. The following paragraphs discuss the methods in detail.

Feature relevance: Kahavi and John [68] designed an algorithm classifying data as per their relevance into three classes: Strong relevance, Weak relevance and Irrelevant. Features having strong relevance provide particular information regarding the class and cannot be altered by any other characteristics without disturbing the nature of the original class. Weak relevant attributes also provide information, but these can be altered without disturbing the original class. Irrelevant characteristics provide no information and can be discarded without loss of data.

Feature redundancy: Strong features are the backbone of proper results and feature relevance takes care of the same. However, no proper guiding principle has been defined showing the nature of redundant features that should be eliminated. Further studies should be initiated for the same.

Chapter 4

Markov Blanket and Proposed Enhancement

4 Introduction

Knowing the required and correct characteristics is a vital requirement in any decision-making process. This process relies upon designing a support model to help in decision making and other states of the art discoveries. In the area of medical science, reduction in the number of unnecessary pathological tests leads to lower the lifetime risk of patients and also lowers the cost of health care. Feature selection is one such technique employed to reduce such unnecessary features. Using the Markov Blanket algorithm [24], we can recognize the best significant genes associated with target disease and predict an early diagnosis of critical life-threatening diseases like Cancer by gene selection and classification. Various algorithms help in achieving the same among which HITON is considered one of the best. HITON decreases the number of variables considerably compared to the given dataset, thereby increasing the level of accuracy [17]. HITON with Tabu shows better performance as compared to other algorithms in the same dataset. Analysis of gene expression based on microarray technology is currently one of the most recent research subjects in bioinformatics. In this work, we are investigating multiple approaches to machine learning to solve some typical problems of gene expression analysis.

4.1 Markov Blanket Algorithm

Aiming for a better prediction requires the selection of the variables that reflect the properties of the class the most by removing unwanted data. Feature selection is one such process that helps in attaining the same. Research has since been undertaken to develop a better method that can increase the level of prediction in newer levels. Of late, a new approach called causal feature selection [26] (Brown et al 2006) has become extremely popular among the users, because the selected causal features can only include the causal contrivance around any class, and is finding extensive usage in developing

future prediction models. The point of difference between causal and other traditional approaches lies in the fact that future prediction models built using the causal approach have the ability to be understood in relation to the causal significance of the feature with the class characteristics. Further, causal character results in better prediction even in a dynamic situation which is characterized by a large array of testing and training data and only results in predicting the result of actions. Towards achieving this, a wide variety of causal feature selection algorithms have been prepared [69]. For example, forcing people to smoke is unethical and it is currently impossible to manipulate the majority of human genes to figure out which genes cause disease and how they interact with it. Furthermore, the developments expected by the explosive growth of biomedical and other information cannot be produced in a reasonable amount of time using only the classical experimental approach in which a single gene, protein medication is attempted at each time, as the space required for experiments is immense. There is a clear need for computational methods to catalyse the discovery process. Algorithms that infer such a causal relationship have been developed that can significantly reduce the number of experiments needed to discover the causal structure and respective studies verified their applicability [24] [70]. The purpose of such algorithms lies in the identification of the Markov Blanket (MB) of the properties of the class or a part of the MB. The essential parts of any Markov Blanket are the parents (direct causes), children (direct effects), and spouses (direct causes of children). The components of an MB are linked by a relation which is shown by the Bayesian network [69].

The Bayesian network has been defined as a probabilistic graphical model that strictly denotes a joint probability distribution P over a set of random variables U using a directed open, acyclic graph (DAG) G marked with conditional probability tables of the probability distribution of a node for any given instance of its parents. Information-rich in quality regarding the variables (conditional independence properties) can be gathered from the graphs obtained from any Bayesian network. However, the related probability distribution data, which are constant with such properties, provides a measurable explanation regarding the nature of the relationship among the different variables. The main feature

of a Bayesian network is the Markov condition property which connects the probability distribution P with the graph G. The property states that the essential condition of a node is not to be dependent on those data which have not been derived from itself. The explanation for the above statement can be understood from the fact that derived data tend to have the same properties as their source, leading to faithfulness within the node and the data. Any Bayesian network can be studied from three viewpoints – faithfulness viewpoint, probability viewpoint, and graphical viewpoint [71].

1. In any Bayesian network, faithfulness assumes utmost importance. The necessary condition for establishing faithfulness among a Bayesian network G and a joint probability distribution P lies in the presence of every unrestricted freedom demanded by the graph G and the Markov condition in P [72] [71]. If the faithfulness assumption is followed, the Markov blanket of T becomes unique, and retrieval of the same from the related Bayesian network over the domain U becomes irrelevant. This network can now be considered to comprise of T's parents, children, and spouses (Figure 4.1).

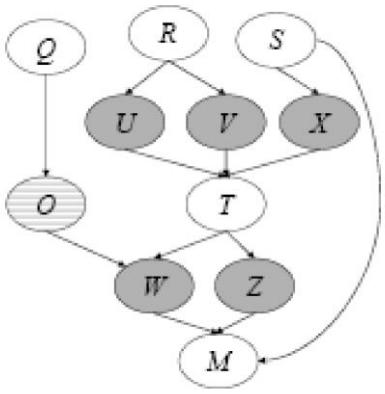

Figure 4.1: An example of a Bayesian network [71]

This process, however, requires the Bayesian network to remain in a state of advanced readiness. Bayesian network is accumulated before advancing towards getting the Markov blanket of a variable. Theoretical knowledge of the Bayesian network, also known as the NP-complete problem, is difficult to accomplish. As such, an ideal solution for this problem lies in prompting the Markov blanket

without having the whole Bayesian network ready. This procedure decreases the run time and enhances our capability of solving a large number of problems with the same set of data.

2. As per the probability viewpoint, if the faithfulness criteria are taken care of, the Markov blanket of T is a minute set of data prepared such that all other nodes are not dependent on T. This ensures the uniqueness of each node and decreases the interference of other nodes in deriving the result.

3. Graphical viewpoint, on the other hand, states that faithfulness conditions being met, the Markov blanket of T is the same to its parents, children, and spouses.

Theorem 1: If a Bayesian network G is faithful to a joint probability distribution P, then,

1. If two pairs of nodes X and Y are dependent on one another under certain sets of conditions, and in the presence of other sets of nodes, there will be an edge between X and Y.
2. In the case of three nodes, X, Y and Z arranged such that Z is equidistant from both X and Y and is placed between the two, X > Z < Y represents a part of the graph of G if X and Y are dependent on every other set of nodes that contains Z under certain conditions [73].

The graphical viewpoint, as conditional independence of two variables X and Y are conditionally independent given Z, denoted as I(X, Y|Z), together with Theorem 1, gives information regarding the topology. This finds reference in finer and more recent algorithms like MMPC/MB, HITON-PC/MB, PCMB, and IPC-MB besides others. All these algorithms have been briefly explained below.

KS (Koller & Sahami algorithm): This concept has been conceived by Pearl [72] and further developed by Koller and Sahami [29]. Their study made Markov blanket an area of interest to data analytics who have since conducted many trials to check the efficiency of the system. Koller and Sahami developed a theoretical idea employing cross-entropy to decrease the amount of information lost during feature selection thereby minimizing the amount of predictive information lost during feature elimination [29]. This concept has been named KS after the developers. An algorithm was

designed using the theory developed, and this was a pioneering effort in the area of Markov blanket. This algorithm did not, however, give accurate results. KS algorithm is bounded by two conditions: (1) the number of variables that the algorithm can accept and (2) the number of variables that could be kept. This boundary decreased the difficulty in searching characteristics, but accuracy was reduced [17] [34].

IAMB (Incremental Association Markov Blanket): This method was designed in 2003 for classifying data that are rich in resources. The algorithm finds its basis among the earlier attempts made in this regard, particularly KS has the same model as its predecessors. The only difference between the older algorithms and IAMB lies in the fact that while the older algorithms consider variables for adding in the beginning, IAMB does so every time a feature is added to the blanket. This results in IAMB performing better than the previous algorithms as this process decreases the possibility of the addition of negatives in the beginning [17] [74]. However, IAMB also suffers from certain drawbacks requiring researchers to propose other varieties like interIAMB, IAMBnPC and their combined version interIAMBnPC [17].

MMPC/MB (Max-Min Parents and Children / Markov Blanket): The previous algorithms relating to Markov blanket relied on the size of the Markov blanket for selection of features which often led to inaccurate results. MMPC/MB was an epoch-making idea as it was dependent upon the connections present in the graph that was truthful to the data. This method employs a two-step approach. Studies conducted by Pena et al. [34] show that the method does not yield accurate results. However, it paved the way for future algorithms like HITON-PC/MB, PCMB, and IPC-MB, which adhere to the same two-phase framework.

HITON-PC/MB: HITON-PC/MB is an effort to render data more competent to deal with the practical obstacles. The way of functioning of HITON-PC/MB is similar to that of MMPC/MB but with minor differences. HITON-PC/MB incorporates the feature of adding and removing nodes which aid in the removal of false-positive data reducing the size of the data set. HITON-PC/MB was also

nor considered a robust mechanism [17]. This is treated as a good learning mechanism that does not require learning the entire Bayesian network.

Conditional Independence: Two random variables X and Y are conditionally independent given Z, denoted as $I(X, Y|Z)$, iff $P(X = x, Y = y|Z = z) = P(X = x|Z = z)P(Y = y|Z = z)$, for all values x,y,z of X, Y, Z respectively, such that $P(Z = z) > 0$ [75].

Where uppercase letters X, Y, Z denotes random variables and lowercase letters x, y, z denotes actual realized values of these random variables.

The algorithm for the HITON is presented below. [17]

Algorithm for HITON

HITON (Data D, Target T: Classifier-inducer A)

"Returns a minimal set of variables required for optimal classification of T using algorithm A."

MB (T) = HITON-MB (D, T) // Identify Markov Blanket
Vars = Wrapper (MB (T), T, A) // Use heuristic search to remove unnecessary variables
Return Vars

Algorithm for HITON-MB

HITON-MB (Data D, Target T)

"Returns the Markov Blanket of T"

PC = parents and children of T returned by HITON-PC (D, T)
PCPC = parents and children of the parents and children or T
CurrentMB = PC ∪ PCPC
// retain only parents of common children and remove parents of parents, children of parents, and children of children.
∀ Potential spouse $X \in$ CurrentMB and $\forall Y \in$ PC:
    if - $\exists S \subseteq \{Y\} \cup V - \{T, X\}$ so that $\perp (T; X|S)$ // V denotes the full set of variables and $\perp(T; X|S)$ the conditional independence of T with variable set X given variable set S.
    then retain X in CurrentMB
    else remove it
Return CurrentMB

Algorithm for HITON-PC

HITON-PC (Data D, Target T)

"Returns parents and children of T."

CurrentPC = {}
Repeat
Find variable Vi ∉ CurrentPC that maximizes
association (Vi, T) and admit Vi into CurrentPC //admits one-by-one the variables in the current estimate of the parents and children set CurrentPC.
If there is a variable X and a subset S of CurrentPC
s.t. ⊥(X: T | S) // ⊥ denotes the conditional independence of T with variable set X given variable set S , and variable subset is discovered that renders it independent of T, then the variable cannot belong in the parents and children set.
Remove X from CurrentPC;
Do not consider X again for admission
Until no more variables are left to consider
Return CurrentPC

Algorithm for WRAPPER

Wrapper (Vars, T, A)

"Returns a minimal set among variables Vars for predicting T using classifier-inducer algorithm A and a wrapping (heuristic search) approach."
Repeat
Select and remove a variable from Vars.
If the internally cross-validated performance of A remains
the same, permanently remove the variable.
Until all variables are considered.
Return Vars

PCMB (Parent- Children Markov Blanket): PCMB is one such algorithm that succeeds MMPC/MB and HITON-PC/MB. PCMB can be regarded as the first theoretically sound model. In tune with the requirements of its predecessors, PCMB also requires two identical assumptions characterized by faithfulness and correct test. PCMB employs the same operational strategy as establishing connections among parents and children like MMPC/MB and HITON-PC/MB. Studies suggest that like IAMB, PCMB can accommodate a large number of attributes [34] and deliver better

and accurate results as compared to IAMB with an identical volume of data [71]. This algorithm, however, suffers from a demerit. Given a large volume of data, PCMB is much slower than IAMB.

IPC-MB (Iterative Parent- Children Markov Blanket): IPC-MB algorithm is the newest addition to the list of such algorithms. It aims at providing a better result than its predecessor, PCMB. The structure of IPC-MB is much similar to MMPC/MB, HITON-PC/MB and PCMB and recognizes the direct connection between the parents and the children. The search procedure is repeated every time a new candidate is found. This helps in the identification of false positives. The later part of Theorem 1 forms the basis of further recognition of true spouses. In contrast to PCMB, IPC-MB decides the connection between pairs of variables in a better way. The experimental method is undertaken as per the following chronology- (1) identifying and eliminating negatives. As the volume of data is large, it is easier to trace our negatives than finding positives, (2) identifying and eradicating all possible negatives in a short span of time to avoid having multiple conditions in any given data set. This saves time as unnecessary tests are avoided, (3) Removal of non-positives is first carried out on a null set. One variable at a time is added, and the test is repeated. This makes the removal of false positives easy and fast. As on date, IPC-MB has been found to the best among this class of algorithms as per the time required, proficiency of data [71].

However, faithfulness assumption criteria form the foundation of all algorithms inclusive of IAMB and its variations. It is a matter of solace that almost all distributions are found to agree with faithfulness assumptions in their own space [76].

4.2 Tabu Search algorithm

Tabu search (TS) is a higher-level algorithm that finds application in arriving at solutions to problems wherein selection and placing in order of data is of importance. One of the recently developed and the best-known feature selection processes, TS provides a viable alternative for an accurate future prediction [77].

The following paragraphs try to describe the design component of any problem. This is followed by steps regarding incorporating the design to a Tabu search. Finally, we would try to describe each component separately.

Any given problem contains a host of data differentiated by their characteristics. Data representing common features provides a solution to the problem. It consists of various components with values 1 if the characteristic is chosen or 0 if not chosen.

Regarding the valuation of the results, a wide array of norms are in use, either involving accuracy or features. Most of the differentiation techniques involve the calculation of accuracy, which is the number of categorized observations in respect of the total observations.

$$Accuracy = \frac{No\ of\ categorized\ observations}{\text{Total number of observations}}$$

In any feature selection problem, characteristics of observations form an important part. As such, designing a robust model involves the minimization of required characters. This method is called features. Features are calculated as:

$$Features = \frac{Selected\ features}{Total\ features}$$

Neighbourhood:

In the context of feature selection problems (also referred to as one flip neighbourhood N_1^0) is defined, for all the elements 's' which are present in the search space, as follows:

$$N_1^0(s) = \{ s' \mid \exists i \in \{1, \ldots, n\}\ s.t.\ a_i' \neq a_i\ and\ \forall j \neq i, a_j' = a_j \}$$

Where s is a subset of features, the solution is represented by a bit string of size n. $s = [a_1, .., a_n]$ represent the total number of features and i^{th} bit a_i indicates if the feature i is chosen ($a_i = 1$). [77]

It is evident that the number of selected features needs to reach a minimum, thus in order to achieve a good solution is said to be one which can be represented with most of its bits equalling 0. Thus, we can conclude that the probability of flipping from 0 to 1 is higher than the probability of transition between 1 and 0, thus creating a bias. In order to remove this bias, the entire neighbourhood is divided into two sub-neighbourhoods, via the add neighbourhood and the drop neighbourhood. The former (add neighbourhood) contains the set of all solutions where the transition of a bit is between 0 and 1, while the latter (drop neighbourhood) contains all transitions between 1 and 0. Thus, it can be concluded that:

$$N_1^0(s) \;=\; N_A(s) \;\cup N_D(s) \;\text{ And } N_A(s) \;\cap N_D(s) \;=\; \emptyset$$

The neighbourhoods are defined as follows:

$$N_A(s) \;= \{\, s' \,\big|\, \exists i \;\in \{1, \ldots, n\}\; s.t.\; a_i' = 1 \; and \;\; a_i = 0 \; and \; \forall j \;\neq\; i, a_j' \;=\; a_j\}$$

$$N_D(s) = \;\{\, s' \,\big|\, \exists i \;\in \{1, \ldots, n\}\; s.t.\; a_i' = 0 \; and \;\; a_i = 1 \; and \; \forall j \;\neq\; i, a_j' \;=\; a_j\}$$

Where, $N_A(s)$ represent add neighbourhood where the transition of a bit is between 0 and 1. $N_D(s)$ represents drop neighbourhood where the transition of a bit is between 1 and 0 [77].

Designing prediction models that have the ability to provide fairly accurate results requires decreasing the number of selected characteristics. So, accurate results can be provided by models fed with data with value = 0.

In algorithms like Tabu Search, exploring the vicinity of a particular solution requires a lot of time and effort. Any Feature Selection (FS) problem evaluates the results by the application of certain classifiers. With respect to the classifier used, the computation time may be high if the dataset includes a large number of genes. Hence repeat of the checking the vicinity of solutions has to be ruled out. To reduce the problems mentioned above, D. Schindl and N. Zufferey designed the Learning Tabu Search (LTS) [78]. The basis of LTS is a function that estimates the quality of individual solutions in its vicinity. This process investigates the vicinity of the solution in two phases – quality appraisal of

the vicinity is conducted, followed by a thorough evaluation of the best ones. The principle of this assessment lies in the statement 'feature or amalgamation of features that have the potential to deliver quality results during the process of search should be preferred while finding solutions'. Hence, the requirement of memory to preserve the quality of each group of characteristics is a prerequisite in LTS. The process of updating has to be carried out regularly, the regularity parameter being defined by respective LTS as it has a bearing on the learning process. This update procedure aims to examine areas that contain better solutions. There is a diversification step that leads the algorithm into new domains.

Studies indicate that LTS gives better results as compared to other models as well as the TS model. Experiments also show that even though both TS and LTS are based on the same principle, the later shows a better promise in the delivery of accurate results.

In order to perform Tabu Search, the algorithm initially generates an initial Markov Blanket (DAG) for the target variable, which may be suboptimal. Thus, a Tabu search is applied to improve the initial Markov blanket. The algorithm is terminating after a fixed number of iterations when no further improvement of the MB search is possible. The algorithm is performed in two different stages, details of which are given below:

Stage 1:

The goal of this stage and corresponding function (*InitialMBSearch)* is to generate an initial Markov Blanket for the target variable *T* using the data. This algorithm has four different phases, via:

1. **Phase 1:** In this phase, we find all the vertices which are adjacent to *T*, and all vertices adjacent to these vertices. Also, all pairs of adjacent vertices are connected by using an undirected graph. This is done by making repeated calls to the function *Checkedges.*
2. **Phase 2:** In this phase, we apply orient to some of the edges by applying edge orientation rules. For each undirected edge X – Y, if it aligns as X → Y in every directed acyclic graph condition, conditional independence to T, then the edge is oriented as X → Y.

3. **Phase 3:** In this phase, the algorithms choose an orientation for some of the remaining undirected edges which are not oriented. This is a simplified heuristic phase of the entire algorithm.
4. **Phase 4:** In the final phase of this stage, we prune the leftover vertices and edges that aren't a part of the Markov Blanket.

Stage 2:

The second stage involves the application of Tabu Search onto the initial Markov Blanket in order to improve this. A total of four different movements are allowed in the TS model, which are: edge addition, edge deletion, edge reversal and edge reversal with node pruning, the operation of which are mostly self-explanatory. For each of the steps, the corresponding Markov Blanket is computed after each move and subsequently the related probabilities, predictions scores using which the best move is selected. The Tabu search algorithm also keeps track of a certain number of previous moves, so that no repetition of moves is observed.

At the end of the Tabu Search, the procedure should return an improved Markov Blanket, which is then used for classification for the target variable.

The algorithm which is performed in this study is described using the following pseudo-code: [35]

Algorithm for Tabu Search Enhanced Markov Blanket

InitialMBSearch (Data D, Target T, Depth d):

Procedure **checkedges** (Vertex V, Graph G, Depth-of-search d, Array of Sets sepSet, set of edges Forbidden):

For each $V_i \in$ vertex (G) \ V;
If $V - v_i \in$ Forbidden
add $V - v_i$ to edges (G);
For each depth = 0,..,d
If (adj(V) has at least depth + 1 vertexes in it)
For each $v_i \in$ adj(V)
If ((v_i is independent of V conditional on S))
Where $S \subset \{adj\ (V) \setminus v_i\}$ & (|S| = depth)
Remove edge $V - v_i$ from edges (G);
Add edge $V - v_i$ to Forbidden;
sepSet(v_i,V) = S;

End procedure
/* The graph G is modified by the procedure **checkedges,** and so it contains different adjacencies at different point in the algorithm. The attributes Forbidden is known as short term memory function that contain moves leading to a new solutions. The Depth-of-search determines the maximum size of the array in conditional independence tests.*/

Initialize vertex (G) to all variables in data set D;
/* if the final sepSet(v_i, v_j) is NULL than v_i and v_j are not separated conditional on any subset of vertices.*/
Initialize edges (G) to ∅;
Initialize sepSet(v_i, v_j) to Null;
Initialize Forbidden to ∅;
/* Finding adjacency; T is the target variable. */
checkedges (T, G, d, sepSet, Forbidden);
For each V in adj(T)
checkedges (V, G, d, sepSet, Forbidden);
For each $V \in adj(adj(T)) \setminus adj(T)$
checkedges (V, G, d, sepSet, Forbidden);
/* Orient G using Orientation rules. */
For each triple of vertices (X, Y, Z)
If $X * - Y - * Z$ and X is not adjacent to Z, and If $Y \notin$(sepSet(X, Z)), then
Orient as $X* \rightarrow Y \leftarrow *Z$;
Repeat
If $X \rightarrow Y - Z$ **and** X, Z are not adjacent **then** orient as $Y \rightarrow Z$;
If $(X \rightarrow Z \rightarrow Y)$, **and** $X * - Y$, **then** orient $X* \rightarrow Y$;
If $W \rightarrow Y \leftarrow Z$, and Z is not adjacent to W, **and** W, Y, Z are all adjacent to V,
$W - V - Z$ do not collide at V, then orient $V * - Y$ as $V * \rightarrow Y$;
Until no more edges can be oriented
/* any undirected and bi-directed edges are deleted after the edge orientation step */

/* Transform into a MBDAG: */

For any v_i in G, s. t. $v_i \leftrightarrow T$ or $v_i - T$;

Orient this edge as $T \rightarrow v_i$; put edge $T \leftarrow v_i$ into the edge list A;

For any w, v_i in G, s. t. $w - v_i \rightarrow T$&w $\notin \{T\} \cup adj(T)$

Remove the edge $w - v_i$; put $w - v_i$ into the edge list A;

For any node v_i , $v_i \notin \{T\} \cup adj(T) \cup adj(adj(T))$

Remove v_i and all the corresponding edges; record the corresponding Edges in A;

Remove any undirected or bi-directed edges from G; put them into the edge list A

Remove any remaining edges among parents and among adj(T) from G; put them into the edge list A

For any node v_i, $v_i \notin \{T\} \cup adj(T) \cup adj(adj(T))$

Remove $v_i$and all the corresponding edges; record the corresponding edges in A;

/* at this point G is an MB DAG: MBDag, A: an edge list not in the output MBDAG, but might be added*/

Return (MBDag, A).

TabuSearch (Data D, Target Y):

Initialize bestSolution := currentSolution := MBDag;

bestScore := currentScore := thescoreofMBDag;

tabuTenure = 7 (in our experiments); tabuList := $\emptyset$.

repeat until (bestScore does not improve for k consecutive iterations)

form candidateMoves for currentSolution

find bestMove among candidateMoves according to function score

update currentSolution by applying bestMove

add bestMove to tabuList

/* Tabu moves will not be re-visited in the next k iterations.*/

if (bestScore < score (bestMove))

update bestSolution and bestScore by applying bestMove

return bestSolution (a MBDag)

InitialMBsearch (D, T, d)

TabuSearch (D, Y)

// * End pseudo-code of TS/MB algorithm.

tabuTenure**:** In Tabu Search implementation, the algorithm keeps all records of m previous moves that are taken for finding the solution. New solutions are marked as tabu by reference to this record if they include attributes of previous solutions found within the horizon of m movements. This m previous moves, if it leads to a tabu solution then the value of m is called the tabuTenure. [79] [35]

(NOTE: sepSet(vi, vj): a mapping of a set of nodes. Graph G Includes different adjacencies in the algorithm at different points. adj(vi): the set of adjacent nodes to node vi in G. Vertex (G) represents the set of vertexes in the graph G. Edges (G) represent the set of edges in the graph G.

Points: non-arrow head, " - ", or arrow head, ">". A "*" is a meta-symbol that represents either arrow head or arrow tail. X ∗ −Y represents either X ← Y or X – Y. In graph G, edge X ∗ −Y is replaced by X∗ → Y the end point with ∗ is left the same. Thus the instruction to replace X ∗ −Y with X∗ → Y says if the edge is X – Y, replace it with X → Y; and if the edge is X ← Y, replace it with X ↔ Y. [35])

Chapter 5
Machine Learning Techniques

5.1 Classification Methods

Many classification techniques are proposed by the research community in machine learning, data mining and pattern recognition (Duda et al., 2000) [18]. Some of the commonly used classification methods are Bayesian classifier, K-nearest neighbour, Artificial Neural Networks and Support Vector Machine (SVM). Classification methods are the heart of machine learning. Data classification involves the following two major steps, learning and classification [18] in learning step or training step, where a classification algorithm builds the classifier by analysing a given training dataset consisting of samples with associated class labels. In the Classification step, built model is used to determine the class label. In this study we deal with a classification problem and that majorly focuses on, once the most predictive and informative genes are selected using our proposed HITON (implementation of Markov Blanket algorithm) with Tabu search (HITON + TABU) algorithm, we input those selected genes into a classifier to get more accurate prediction results.

Supervised methods of analysis are predominantly used to identify the differences at the molecular level between the known classes and to diagnose or predict the class of the new sample [18]. This technique is also known as classification. However, we have used K-nearest neighbour, Neural Networks and Support Vector Machine in our experiments. A brief description of these techniques is given below:

5.1.1 K-Nearest Neighbour (KNN)

K-Nearest Neighbour is amongst the most rudimentary yet important classification algorithms employed in machine learning. Though KNN is used both for classification as well as regression, yet it has wide acclaim as a classifier. This algorithm is widely used in recognition of patterns, mining of data, etc. KNN does not have any preconceived notion regarding how data is scattered. The raw data

given is classified as per their features. This classification of data into groups follows the nearest neighbour logic when the data is assigned a class whose features are very close to the features of the data [20].

The K-Nearest Neighbour is simple and easy to implement. In this method, the case is classified by a majority vote of its neighbours, once the case is assigned to the class and most common amongst its K nearest neighbours measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbour.

There are specifically, four most popular distance functions: Euclidean Distance, Manhattan Distance, Minkowski Distance, and Hamming Distance [80].

$$\text{Euclidean: } \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \quad [81]$$

$$\text{Manhattan: } \sum_{i=1}^{k}|x_i - y_i| \quad [82]$$

$$\text{Minkowski: } \left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{\frac{1}{q}} \quad [83]$$

The above Euclidean, Manhattan and Minkowski distance measures are only valid for continuous variables, for instance of categorical variables the Hamming distance must be used.

$$\text{Hamming: } D = \sum_{i=1}^{k}|x_i - y_i|$$

$$x = y \Rightarrow D = 0$$

$$x \neq y \Rightarrow D = 1 \quad [84]$$

Example: if the value of x = male and y = male then Hamming distance is 0. If value of x = male and y = female then Hamming distance is 1.

Figure 5.1 shows the general concept of the k-nearest neighbour algorithm.

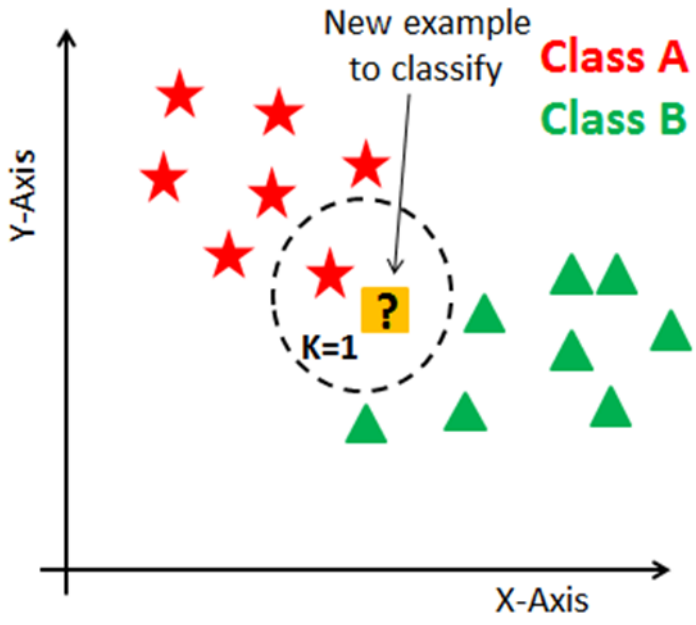

Figure 5.1: Image showing the working of a K-nearest neighbour algorithm [20]

For instance, in Figure 5.1, class A (Red- star) and class B (Green- triangle) are objects at different places in the hyperspace. Classification of an object is done based on voting based on the classes of its neighbours and input is a vector of some k objects where k=1in Figure 5.1, the object itself is classified as its nearest neighbour.

5.1.2 Neural Network (NN)

Neural Network has been built as per the structure of the nerves in the brain. The brain essentially learns from experience. There are many changes in this primary type of neurons in humans. Also, all-natural neurons have the same four necessary components. These components are recognized by their botanical names - dendrites, soma, axon, and synapses. In some cases, neural systems are represented in terms of their understanding including how many layers they are between knowledge and yield, or the supposed shrouded layers of the model. The term neural network is therefore used synonymously with machine learning. A neural network is a set of connected input/output units throughout which every connection includes a weight-related to it. An Artificial Neural Network (ANN) is based on a

set of connected units or nodes known as artificial neurons [18]. Each connection between artificial neurons can transmit a signal from one to another and each connection is linked with a numeric number called weight. Different types of neural networks are Multilayer Perceptron (MLP), Radial Basis Function Networks (RBNF), CNN, etc. Figure 5.2 shows the structure of a simple neural network.

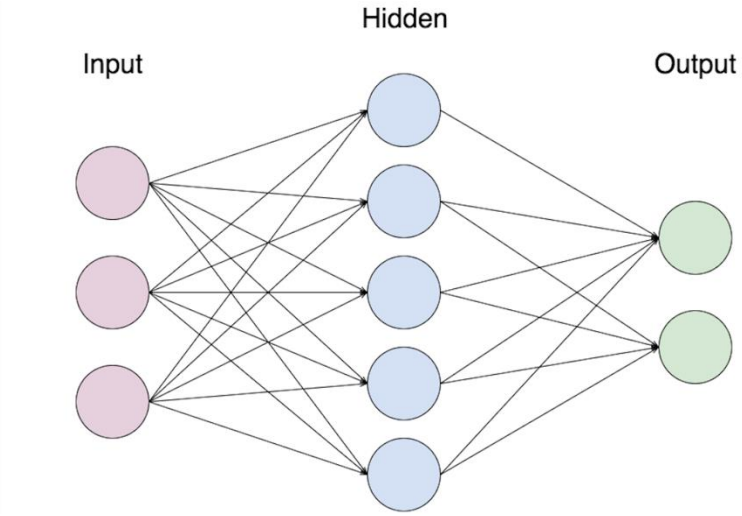

Figure 5.2: Shows the structure of a simple architecture of NN [85]

As per the structure of the simple architecture of neural network in Figure 5.2, the first column is represented as the Input Layer of neuron which is also called “nodes” or “units” linked by lines connected with a numeric number called weight. The output, h_i, of neuron i in the hidden layer is:

$$h_i = \sigma(\sum_{j=1}^{N} V_{ij} x_j + T_i^{hid}), \quad [86]$$

Where $\sigma()$ is called activation function, N the number of input neurons, V_{ij} the weights, x_j inputs to the input neurons, and T_i^{hid} the threshold terms of the hidden neurons.

Multilayer Perceptron (MLP): MLPs consists of multiple layers of computational units, usually interconnected in a feed-forward way. Neurons in one layer have directed connections to the neurons of the subsequent layer. The input layer contains neurons that represent the predictor/independent

variables in the data. The top layer is the output layer which includes the dependent variable. The layers in between are referred to as 'hidden' layers. MLPs learn through a variety of techniques, most popular being back propagation. In back propagation the output variables are compared with the answers to compute the values of error functions. Using various methods these error functions are then fed back to the network. After performing this process over numerous learning cycles the network attains the stage where the error between predicted and actual classification is sufficiently small. In this thesis Multilayer Perceptron was used for the Ranking feature, it allows for the examination of the inputs as they relate to the output values of the multilayer perceptron without examining every subset.

Radial Basis Function Networks (RBF): RBF contains three layers: an input layer, a hidden layer with a non-linear RBF activation function and a linear output layer. The neurons in the hidden layer contain functions whose outputs are inversely related to the distance from the centre of the neuron. The output layer is a linear output unit. RBF networks are fast to learn and reasonably compact.

5.1.3 Support Vector Machine (SVM)

Vladimir Vapnik, Bernhard Boser and Isabell Guyon introduced the concept of support vector machine in their paper [87]. The modern standard soft margin was proposed by Corinna Cortes and Vapnik in 1993 and published in 1995 [88].

A support vector machine (SVM) is a machine learning (ML) algorithm that analyses data for both numeric prediction and classification. The support vector machine is also known to be a supervised learning method. SVM transforms the initial knowledge into a better dimension using a nonlinear mapping [87]. The process of the SVM algorithm is based on finding the hyperplane correctly. Classifies all the data while being farthest away from the data points. The algorithm finds this hyperplane (aka optimal separating hyperplane) maximal marginal hyperplane using support vectors and margins defined by the support vectors. SVM builds a twin classifier and assigns values of +1

for positive class and -1 for negative class. Data is put into respective classes as per their features [89]. Figure 5.3 shows the separating hyperplane with small margin or large margin.

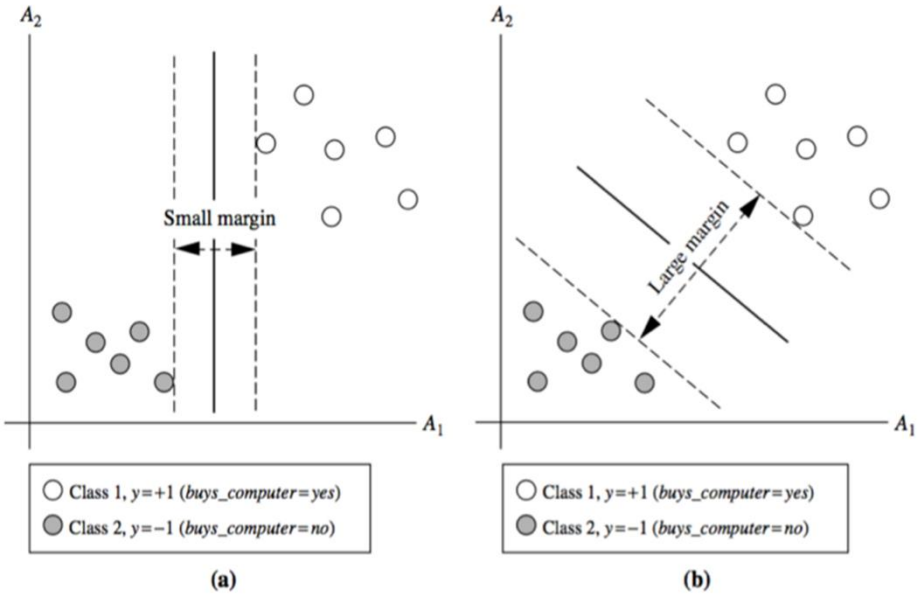

Figure 5.3: (a), (b) Showing the class division using SVM [90]

The dotted lines in Figure 5.2 shows the separation of points from the two classes with a small margin in (a) and a large margin in (b). A line is not good if it passes closer through a certain point because it will contain noise and the solution won't be precise. Therefore, the main aim is to find the line far from all the points on which the SVM method is based on finding the hyperplane correctly or decision boundary between two classes "buys computer=yes" class and "buys computer=no" class. Support Vector is the data point that is nearest to the decision boundary which helps to maximize the margin of the classifier and this is the point that helps us build our SVM.

Formally a separating hyperplane can be defined as:

$W \cdot X + b = 0$ [91]

In the above equation, W is a weight vector, b is a scalar often referred to as a bias.

Applying two attributes Class 1 and Class 2, (e.g., X(x1, x2)), where x1 and x2 are the training tuples. Thus, any points on top of the separating hyperplane belong to Class 1:

$W \cdot X + b > 0$

And any points below the separating hyperplane belong to Class 2:

$W \cdot X + b < 0$ [91]

5.2 Tools

For the data analysis, classification, feature selection, and implementation of algorithms, we have used MATLAB R2018b.

5.2.1 MATLAB

MATLAB stands for matrix laboratory. MATLAB is a high-level programing language for scientific computing. MATLAB integrates computing, visualization, and programming into a user-friendly environment where identifiable mathematical notation represents problems and responses. It includes some built-in functionality like Matrix manipulation and linear algebra, data analysis, scientific graphics and visualisation, etc. This functionality of MATLAB allows for algorithm construction, modelling, simulation, prototyping, application advancement, including Graphical User Interface construction.

MATLAB was written to provide easy entrance to matrix software developed by the LINPACK and EISPACK projects, which collectively describe the state of art in software for matrix computation. It has developed over a span of years with information from various users. In university environments, it is the official instructional mechanism for beginning and superior courses in arithmetic, engineering, and science. MATLAB is the instrument of choice for high-benefit research, progression, and examination.

MATLAB highlights a family of particular application resolutions called toolboxes. Particularly relevant to most users of MATLAB, toolboxes allow image processing, signal processing, optimization and genetic algorithm. MATLAB is an interactive environment in which commands may be scripted to implement your own procedures or functions that perform the specific tasks.

5.3 Cross-validation

Cross-validation is a procedure used to appraise the machine learning model on data. This technique is generally used when the number of samples is less. The cross-validation technique often used in machine learning to compare and select a model for a given modelling problem because it is easy to implement, easy to learn. Cross-validation is a systematic method of performing continuous holdout which improves it efficiently by reducing the variance of estimation. We take a training array and build a classifier. Then we evaluate the efficiency of that classifier, and there's some degree of variance in that evaluation because it's all below the statistics. Cross-validation is a technique that can prevent the risk of overfitting.

Accuracy of a classifier refers to the ability of the classifier to correctly predict the class label of new data. There are different ways of finding the predictive accuracy of the model. The most commonly used technique is the holdout method [18]. In this method, the given dataset is randomly divided into two independent sets: a training set and a test set. The training set is used to build model and accuracy is estimated using a test set.

Another technique used to determine predictive accuracy is k-fold cross validation. This technique is generally used when the number of samples is less. In k-fold cross validation, the initial data are randomly partitioned into k mutually exclusive subsets D1, D2⋯ Dk. each of approximately the same size. In this process, the sample data set is subdivided into k number of smaller sets. The k-fold technique makes use of a part of this sample to construct the model and another part of the sample for testing. This method uses the following steps for each k-fold.

- The given data is subdivided into k number of subsets.
- A model is prepared and tested for one such data subset, say k-10 subset.
- On successful testing of the model, the same is corroborated for the other subsets. In this step, all the features of the model viz., accuracy, run time, etc. are checked.

The process of estimating the performance of cross-validation is done by subjecting the model to all the remaining subsets. In all experiments 10-fold cross-validation has been used. The level of

performance for each of the subsets are recorded individually and then the average of the results is calculated to get an estimate.

5.4 Training and Testing Ratio

In all experiments 10-fold cross-validation has been used. The models were trained using a training: testing ratio of 70:30 as this ratio was found to give the best prediction accuracy of the test data. This gives relatively accurate results when the number of data is large. Various studies conducted in this regard suggest that 70 parts of training data and 30 parts of testing cum validation data provides good result. 70:30 split ensures that trained ANN, SVM and KNN are generalised and has the capacity to provide accurate results rather than any other ration [17].

5.5 Evaluated methods

For classification, especially for two-class problems, a variety of measures (error-rate, accuracy, specificity, sensitivity, and precision) has been derived from the confusion matrix. Let us define an important measure of binary classification by 2×2 tables; it is used for evaluating the classifiers according to their performance. There are four possible cases, as shown in Table 1

Table 1: Confusion Matrix

	True class		
Predicted class	Positive	Negative	Total
Positive	True positive (tp)	False positive(fp)	p
Negative	False negative (fn)	True Negative(tn)	n
Total	p'	n'	N

A confusion matrix presents information on expected classifications as well as actual classifications conducted after a classification system. There are four potential classifier outcomes to predict class label instances:

- If the instance is classified as positive, then it is a true positive.
- If the instance is classified as negative, then it is a true negative.

- If the instance is positive and classified as negative, then it is a false negative.
- If the instance is negative and classified as positive, then it is a false positive.

We may derive some classification performance characteristics from the confusion matrix. Based on Table 2, we can sum up some performance measures as follows.

Table 2: Performance measures based on confusion matrix (Table2)

Name	Formula
Error rate	(fp+fn)/N
Accuracy	(tp+tn)/N ((or) 1- Error-rate)
tp-rate	tp/p'
tn-rate	tn/n'
Precision (or positive predictive value)	tp/ P
Recall =tp-rate	tp/p'
Sensitivity = tp-rate	tp/p'
Specificity	tn/n' (or (1- fp-rate)

Chapter 6
Results and Discussion

In this section, the performance of the Hiton and proposed modification on Hiton with Tabu Search feature selection is empirically evaluated on five cancer microarray datasets: Lymphomas, Prostate, Leukemia, Brain Tumor, and Lung Cancer. The algorithm was independently executed multiple times on the previously described five datasets to find a subset of genes having statistically meaningful conclusions.

MATLAB has been used for training, validation, classification, feature selection, and implementation of algorithms. The proposed hybridized HITON algorithm with Tabu search has been compared with HITON algorithm for better prediction accuracy with three different classifiers. The three different classifiers used in this research are K - nearest neighbour (KNN), Support Vector Machine (SVM), and Neural Network (NN). Experiments were run on a computer with an Intel(R) Core(TM) i7 Dual processor running at 2.50GHz using 16 GB of RAM, running Windows version 10 Pro.

6.1 Datasets Description

In order to test the level of accuracy in the performance of the gene selection methods, the methods are put to a real test. A microarray gene expression dataset is usually represented as an N×M matrix, where N, which is the row, represents experimental samples and M, which is the column, represents a number of genes involved in the experiments. Each cell in the N×M matrix is the level of expression of a specific gene in a specific experiment.

These datasets are used to demonstrate the feature selection methods' optimization power. The datasets consist of small samples with high dimensionalities. The Lung cancer diagnosis data set is downloaded from "http://www.pubmedcentral.nih.gov". The five different microarray datasets used in this research are Lymphomas, Prostate, Leukemia, Brain Tumor and Lung cancer data have been presented in the gene expression model selector [92] and are described in Table 3.

In Table 3, we present a detailed description of these benchmark microarray gene expression datasets with respect to the number of classes, number of samples, and the number of genes.

Table 3: Description of experimental datasets

Data Set	Samples	Genes	Classes
Lymphomas Cancer	77	5470	2
Prostate Cancer	102	1500	2
Leukemia Cancer	47	2000	2
Brain Tumor Cancer	50	10367	4
Lung Cancer	160	12600	2

6.2 Results and Performance Analysis

This study depicts the results in four different parts. Table 4 provides the results of average selected genes, running time in seconds, average test error on five cancer datasets as previously mentioned in Table 3 using Hiton and HITON with Tabu feature selection algorithms. In Table 5, the results show the performance of Classification based on accuracy for HITON algorithm for all microarray cancer datasets. Table 6 provides the output of Classification based on HITON with Tabu search algorithm for all microarray datasets. Table 7 compiles both the results and gives a comparison of the performance.

Table 4: Results for feature selection using Hiton and Hiton with Tabu Search methods.

Feature Selection Methods	Hiton	Hiton + Tabu
Prostate cancer data set		
Average -selected genes	12.05	**10.05**
Running time	**4.539613**	6.859613
Average test error	11 **%**	**6.35%**
Leukemia cancer data set		
Average -selected genes	20.5	**16.8**
Running time	26.33163	**21.18579**
Average test error	6.81%	**3.99%**
Lymphomas cancer data set		
Average -selected genes	7.34	**7.21**
Running time	**7.634572**	11.51453
Average test error	8%	**7.74%**
Brain Tumor cancer data set		
Average -selected genes	10.38	**10.10**
Running time	**8.887203**	13.43395
Average test error	**9.98 %**	10 %
Lung cancer data set		
Average -selected genes	19	**13**
Running time	39.20955	28.64617
Average test error	31.97%	**27.61%**

Table 5: Prediction Accuracy for Hiton Algorithm Datasets / Classifiers.

Classifiers	KNN	NN	SVM
1- Prostate cancer	97.83%	93.10%	**98.04%**
2- Leukemia cancer	**98.81%**	96.38%	98%
3- Lymphomas	93.34%	N/A	**98.50%**
4. Brain Tumor	91.36%	96.03%	**97.80%**
5- Lung Cancer	**90.65%**	88.38%	89%
Average	94.39%	93.47%	**96.26%**

Table 6: Prediction Accuracy for Hiton with Tabu Search Algorithm Datasets / Classifiers.

Classifiers	KNN	NN	SVM
1- Prostate cancer	98%	96.25%	**99.40%**
2- Leukemia cancer	**99.83%**	96.38%	99.80%
3- Lymphomas	95.54%	N/A	**98.72%**
4. Brain Tumor	93.54%	97.29%	**97.95%**
5- Lung Cancer	91.65%	84.65%	**92.36%**
Average	95.71%	93.64%	**97.65%**

Table 7: Summary of the performance for (i) Hiton and (ii) Hiton with Tabu Search Algorithm for all microarray datasets.

Classifier / **Feature selection**	**KNN**		**NN**		**SVM**	
	Hiton	**Hiton +Tabu**	**Hiton**	**Hiton +Tabu**	**Hiton**	**Hiton+Tabu**
1-Prostate Cancer						
# of selected genes	45	42	52	45	46	38
Accuracy	97.83%	**98%**	93.10%	**96.25%**	98.04%	**99.40%**
2-Leukemia Cancer						
# of selected genes	28	26	34	34	28	18
Accuracy	98.81%	**99.83%**	**96.38%**	**96.38%**	98%	**99.80%**
3-Lymphomas						
# of selected genes	77	77	N/A	N/A	63	61
Accuracy	**93.34%**	**95.54%**	N/A	**N/A**	98.50%	**98.72%**
4- Brain Tumor						
# of selected genes	53	52	50	48	46	45
Accuracy	91.36%	**93.54%**	96.03%	**97.29%**	97.80%	**97.95%**
5- Lung Cancer						
# of selected genes	60	47	67	67	49	46
Accuracy	90.65%	**91.65%**	**88.38%**	84.65%	89%	**92.36%**

Table 7 shows a comparison of our work with HITON [13] for the cancer microarray datasets Prostate, Leukemia, Lung, Brain Tumor, and Lymphomas. From this table, it is very clear that obtained accuracy by our proposed algorithm Hiton with Tabu search performed better or at par with almost all datasets and in most cases the results are accurate with least selected genes.

1. Lymphomas Cancer data set: Lymphomas dataset, which consists of 77 samples and 5470 genes is used with our proposed algorithm and classification accuracy is computed. It has been observed that the average running time in respect of the two algorithms was HITON = 7.634572 seconds, HITON with Tabu = 11.5145 seconds. The average selected genes for HITON were 7.34 while for Hiton with Tabu algorithm the average selected genes were 7.21. Average Test Error was calculated at 8% in the case of HITON and 7.74% in the case of HITON with Tabu search. Proposed algorithm HITON + Tabu with SVM achieved 98.72% classification accuracy for Lymphomas dataset, whereas HITON with SVM gives 98.50% classification accuracy, an increase by 0.22% accuracy using HITON + Tabu search. Figure 6.1 shows the accuracy of results of the Lymphomas dataset.

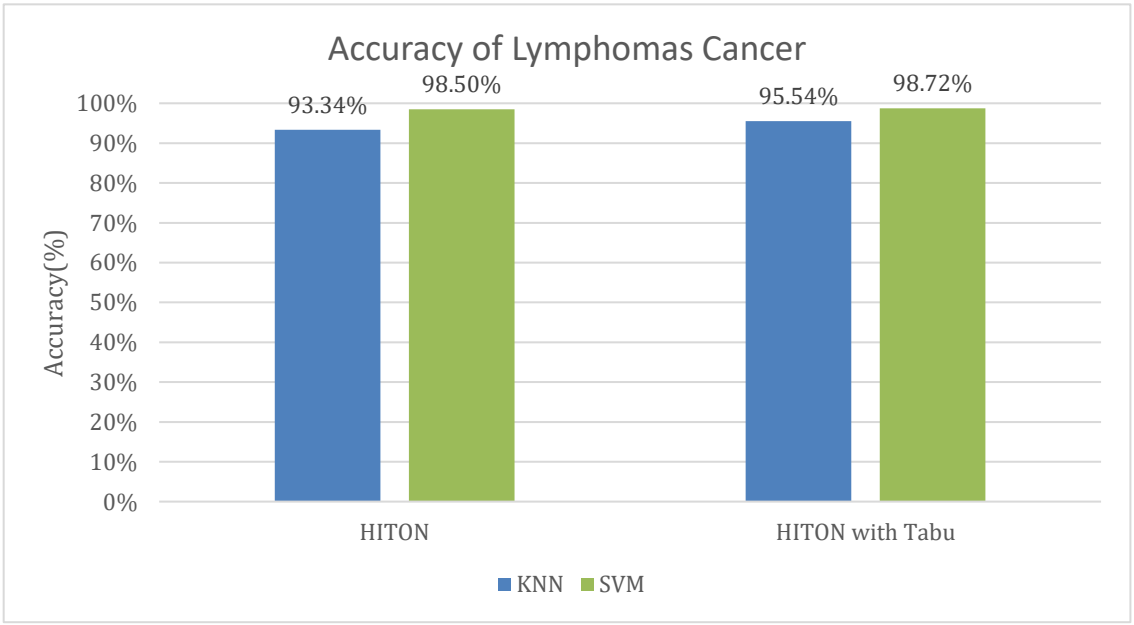

Figure 6.1: Graphical Analysis of Lymphomas cancer dataset

Using HITON algorithm, SVM classifier performed better with an accuracy of 98.50% with 63 selected genes, and KNN classifier, the accuracy was 93.34% with 77 selected genes. No result was

obtained in the above dataset using the NN classifier. Using HITON with Tabu search, the accuracy obtained with KNN was 95.54% with 77 selected genes, and 98.72% of accuracy was achieved by SVM with 61 selected genes, while the NN classifier did not yield any output.

2. Prostate Cancer data set: Prostate Cancer dataset consists of 102 samples and 1500 genes. It has been observed that the average running time for the two algorithms was: HITON = 4.539613 seconds, HITON with Tabu = 6.859613 seconds. The average selected genes for HITON was 12.05 while the average selected genes for Hiton with Tabu search was 10.05. Average Test Error was calculated at 11% in the case of HITON and 6.35% in the case of HITON with Tabu search. Proposed algorithm HITON + Tabu with SVM achieved 99.40% classification accuracy for Prostate cancer dataset, whereas HITON with SVM gave 98.04% classification accuracy, an improvement by 1.36% accuracy using HITON + Tabu search. Figure 6.2 shows the accuracy results of the two algorithms with the classifiers for the prostate cancer dataset.

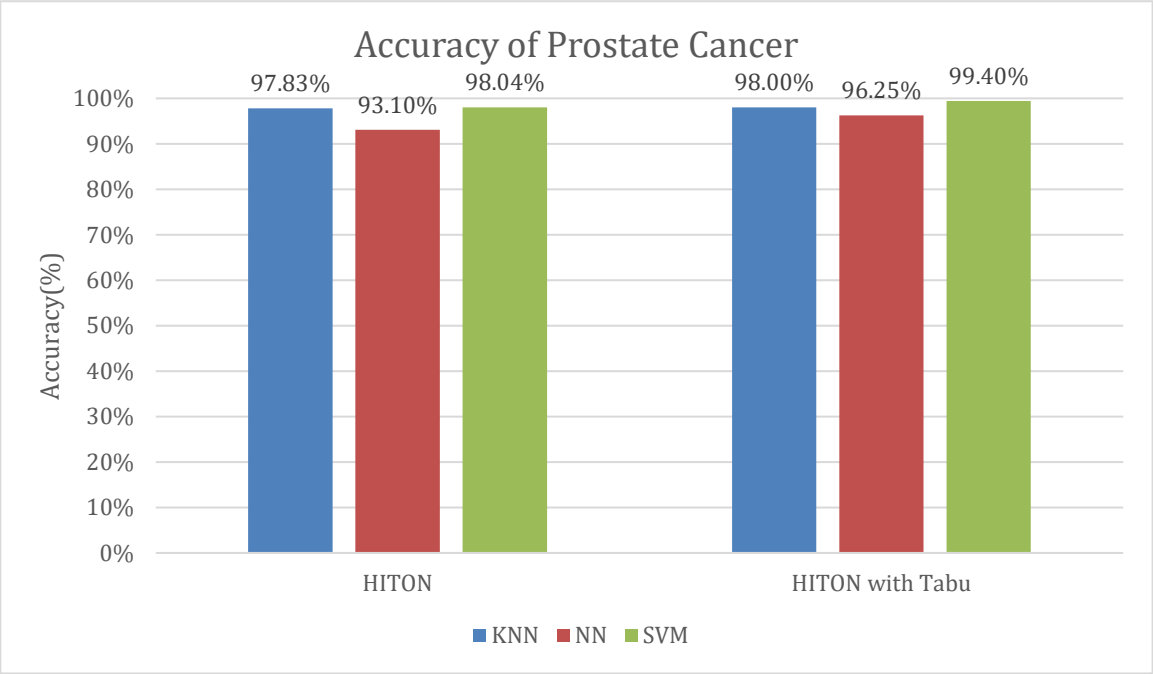

Figure 6.2: Graphical Analysis of Prostate cancer dataset

Using HITON algorithm with the prostate cancer data set, accuracy achieved was 97.83 % with 45 selected genes using the KNN classifier. NN classifier showed an accuracy level of 93.10% with 52 selected genes while the level of accuracy improved to 98.04 % with 46 selected genes using the

SVM classifier. Using HITON with Tabu search, the levels of accuracy recorded using the various classifiers stand at 98% with 42 selected genes for KNN, 96.25% with 45 selected genes for NN, while the SVM classifier showed a high accuracy level of 99.40% with 38 selected genes.

3. Leukemia Cancer: Leukemia cancer dataset consists of 47 samples and 2000 genesIt has been observed that the average running time of the two algorithms was: HITON = 26.33163 seconds, HITON with Tabu = 21.18579 seconds. The average selected genes for HITON was 20.5 while the selected genes for Hiton with Tabu search was 16.8. Proposed algorithm HITON + Tabu with KNN achieved 99.83% classification accuracy for Leukemia dataset, whereas HITON with SVM gives 98.81% classification accuracy, an improvement by 1.02% accuracy using HITON + Tabu search. Figure 6.3 shows the accuracy of the three classifiers for the two algorithms.

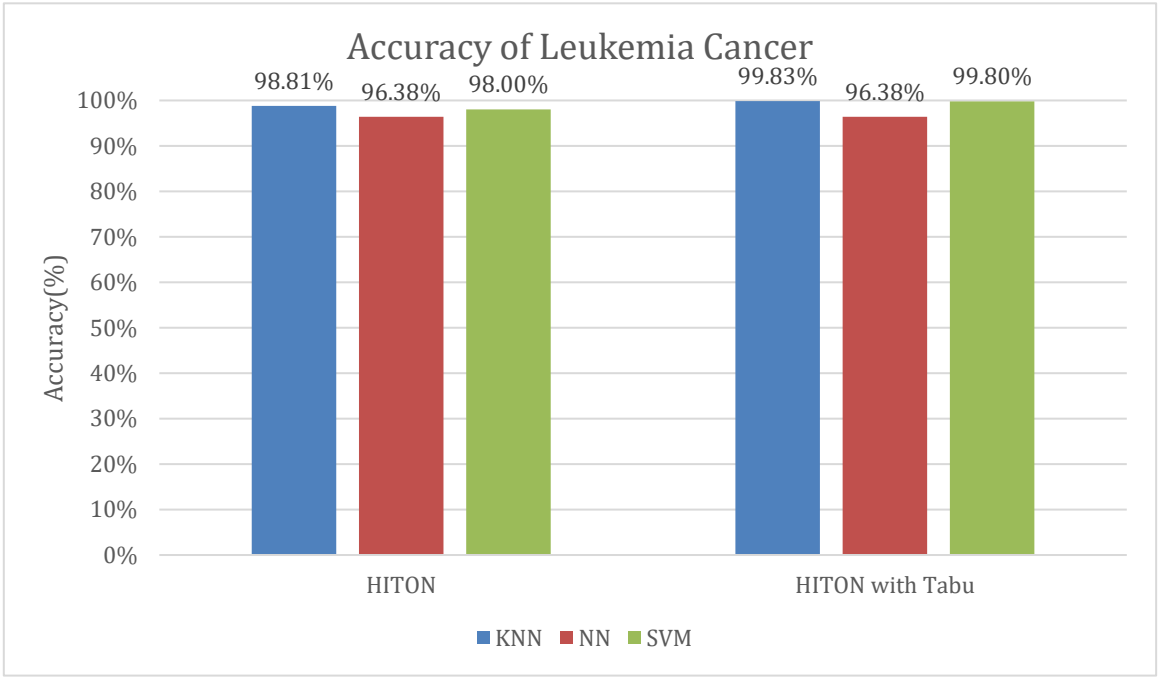

Figure 6.3: Graphical Analysis of Leukemia cancer dataset

Figure 6.3 shows that using HITON algorithm, KNN achieved an accuracy of 98.81% with 28 selected genes, while the SVM classifier achieved an accuracy of 98% with 28 selected genes. NN classifier, however, fared lower at an accuracy of 96.38% with 34 selected genes. Using HITON with Tabu search, the SVM classifier achieved a high accuracy of 99.80% with 18 selected genes.

4. Brain Tumor Cancer data: It has been observed that the average running time in respect of the two algorithms was: HITON = 8.887203 seconds, HITON with Tabu = 13.43395 seconds. The average selected genes with HITON was recorded as 10.38 while the selected genes for Hiton with Tabu search was 10.10. Average Test Error was calculated at 9.98% in the case of HITON and 10% in the case of HITON with Tabu search. Proposed algorithm HITON + Tabu with SVM achieved 97.95% classification accuracy for Brain Tumor dataset, whereas HITON with SVM gives 97.80% classification accuracy, an increase by 0.15% accuracy using HITON + Tabu algorithm. Figure 6.4 shows the accuracy of the three classifiers with the two algorithms for Brain Tumor dataset.

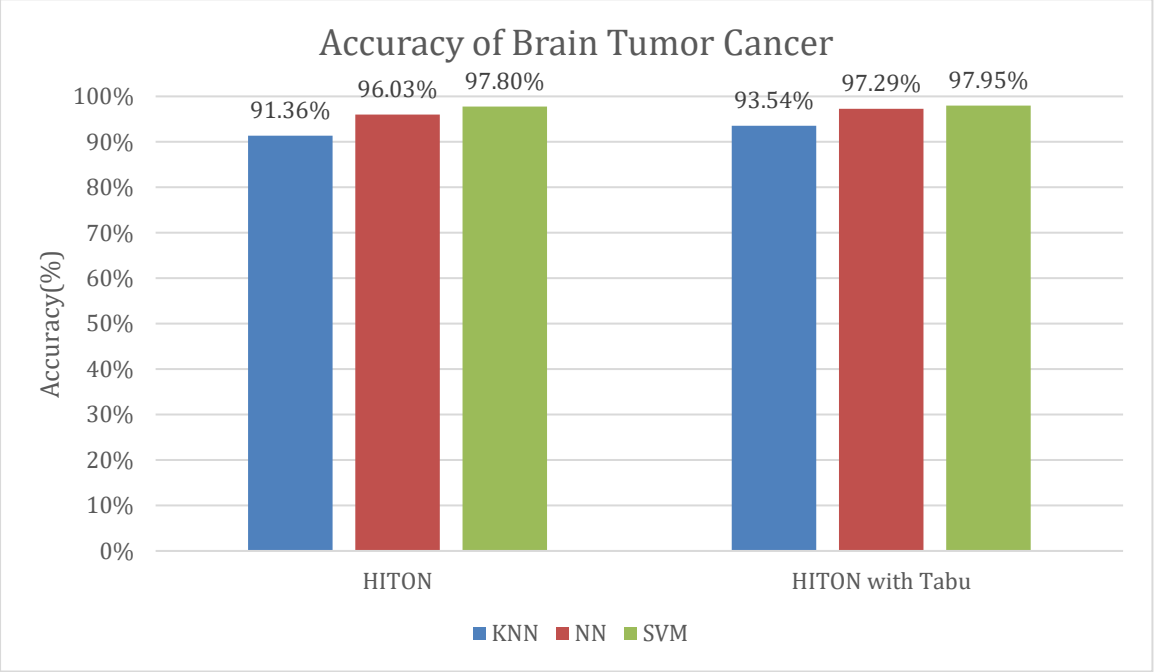

Figure 6.4: Graphical Analysis of Brain Tumor cancer dataset

Figure 6.4 shows that using HITON algorithm with the SVM classifier achieved the highest accuracy at 97.80% with 46 selected gene followed by NN with accuracy of 96.03% with 50 selected genes. The accuracy of the KNN classifier was the lowest at 91.36% with 53 selected genes. Using HITON with Tabu search, SVM classifier achieved the best accuracy for the Brain Tumor dataset followed by the NN classifier. KNN classifier has the worst accuracy in this case. An accuracy of 93.54% with

52 selected genes was achieved for KNN, 97.29 % accuracy with 48 selected genes was achieved for NN and 97.95% accuracy with 45 selected genes was achieved for SVM classifier.

5. Lung Cancer: In the Lung Cancer dataset consists of 160 samples and 12600 genes. Average Test Error was calculated at 31.97% in the case of HITON and 27.61% in the case of HITON with Tabu search. Proposed algorithm HITON + Tabu with SVM achieved 92.36% classification accuracy for Lymphomas dataset, whereas HITON with SVM achieved 89% classification accuracy, an increase by 3.36% accuracy using HITON + Tabu algorithm. Figure 6.5 shows the accuracy of the three classifiers with the two algorithms for Lung cancer dataset.

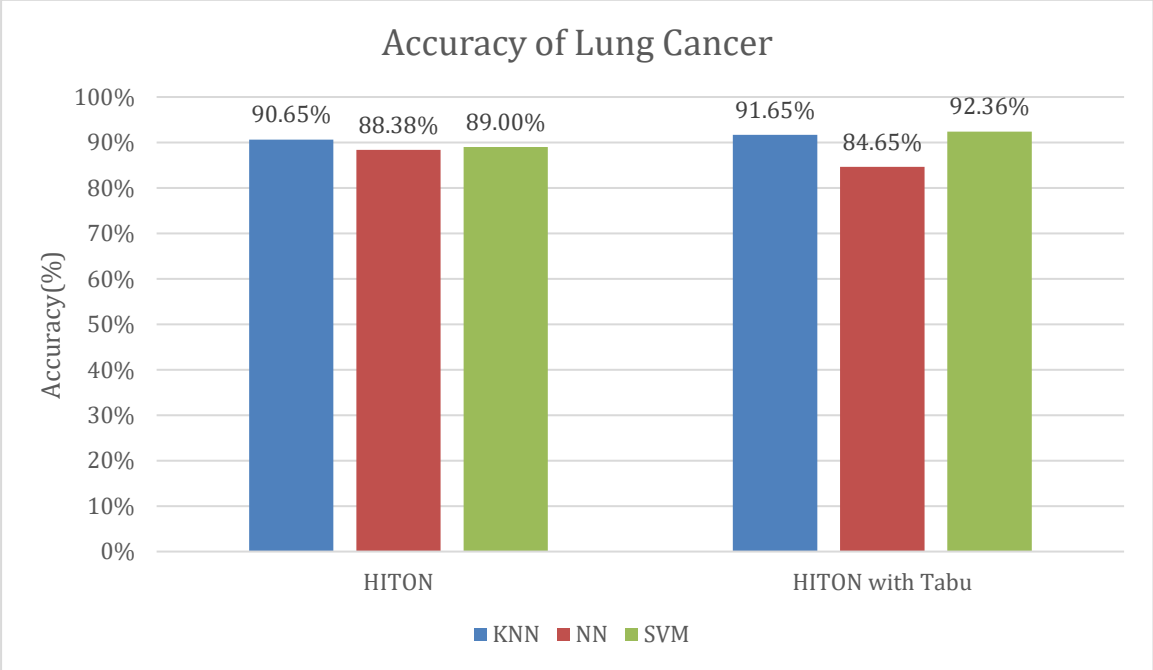

Figure 6.5: Graphical Analysis of Lung cancer dataset

Figure 6.5 shows that using HITON algorithm KNN achieved an accuracy of 90.65% with 60 selected genes, while the SVM classifier showed an accuracy of 89% with 49 selected genes. NN classifier, however, fared lower at an accuracy of 88.38% with 67 selected genes. Using HITON with Tabu search, the SVM classifier showed a high accuracy of 92.36% with 46 selected genes. Figure 6.6 shows the combined results of accuracy for the five datasets using the three classifiers with the HITON and HITON+Tabu algorithms.

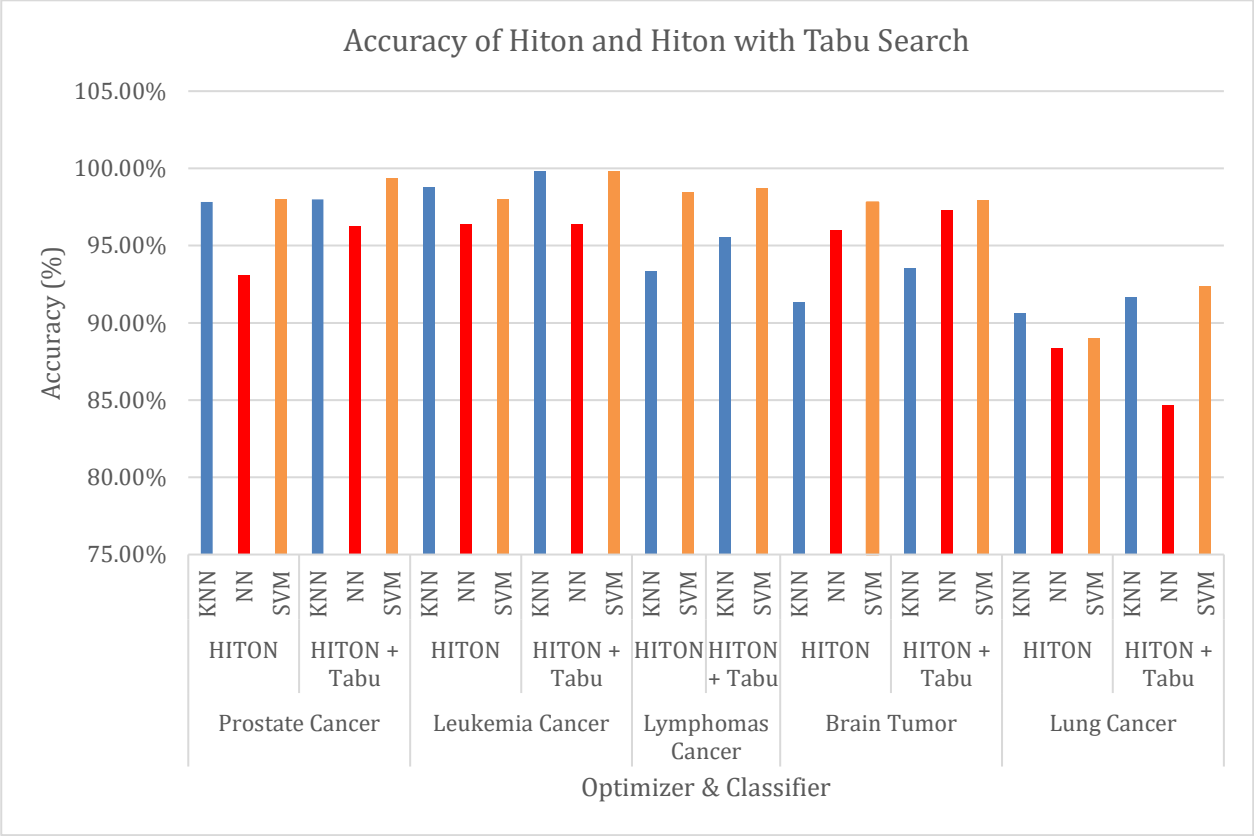

Figure 6.6: Graphical Analysis of Hiton and Hiton with Tabu Search Algorithm for all microarray datasets

Running Time: It can be observed that when the HITON algorithm was executed with 7.34 average selected genes in Lymphomas dataset, the run time was recorded at 7.634572 seconds. In the Prostate cancer dataset with 12.05 average selected genes, HITON showed a much lower run time at 4.539613 seconds. The runtime, however increased when the HITON with Tabu algorithm was run using 10.05 average selected genes. The run time of HITON increased considerably in the Leukemia cancer dataset with 20.5 average selected genes. These results point to the fact that no conclusive decision can be taken as regards the performance of HITON and HITON with Tabu in terms of the run time. There is an additional time to run Tabu search to refine the results. The run time does not show any definite trend in relation to a number of genes selected.

The performance of HITON also shows a similar trend with Lung cancer dataset where 19 genes had been selected, the run time was seen as 39.20955 seconds. However, when 13 selected genes from the Lung dataset were obtained by running the HITON with Tabu search, the run time decreased to 28.64617 seconds. The HITON Shows run time for 10.38 average selected genes pertaining to the Brain Tumor dataset as 8.887203 seconds and 10.10 average selected genes using HITON with Tabu, a running time of 13.43495 seconds.

From the above observations, it can be understood that the running time is not dependant on the number of genes selected. However, running time is considerably higher while using Leukemia and lung cancer datasets for any number of selected genes.

Average test error: With the HITON algorithm, we can observe average test error as 11% on the Prostate dataset. In the Prostate dataset, the average test error is seen as 6.35% using HITON with Tabu search. The average test error for the Leukemia dataset stands at 6.81% using HITON while the HITON with Tabu search, the average test error is seen as lower to 3.99%. HITON and Proposed algorithms show a high average test error on Lung cancer dataset with average test error of 31.97% and 27.61% for the HITON+Tabu and HITON algorithms, respectively. However, for the Lymphomas dataset, the HITON and HITON with Tabu search have an average test error of 8% and 7.74% respectively. The corresponding figures for Brain Tumor dataset, the HITON and HITON with Tabu algorithm have an average test error of 9.98% and 10% respectively.

Average test error using HITON and Proposed algorithm on various datasets like Prostate, Leukemia, Lung, Brain Tumor, Lymphomas was recorded. From Table 4, it is clear that obtained average test error by our proposed algorithm HITON with Tabu search performed better or at par with HITON for almost all datasets and in most cases, the results are accurate with least test error.

Table 5 and Table 6 is the comparison of the accuracy obtained using HITON alone and a mixture of HITON and Tabu. The performance of the classifiers can be explained as under.

KNN Classifier: Using 45 features of prostate cancer dataset and HITON alone, accuracy was marginally lower using KNN than HITON with Tabu, though the later had 42 features. With 26 features of the Leukemia data set and using the KNN classifier on HITON with Tabu search, showed better accuracy than HITON. A similar observation is seen on cell lymphomas dataset, which has 77 features. With 52 features in Brain Tumor dataset HITON with Tabu search shows better accuracy. On the lung cancer dataset using the HITON with Tabu fares better than the HITON.

NN Classifier: Even with an increase (52 for HITON alone and 45 for HITON with Tabu) in the number of features on the prostate cancer dataset, HITON with Tabu performed better. With 34 features of the Leukemia data set and using the NN classifier on both algorithms, the same accuracy level is observed. Lymphomas do not give any output and hence no comment can be made. With an equal number of features (67 in both classes of algorithms), in the Lung cancer dataset, HITON showed a higher accuracy. With a higher number of features (50) of lung cancer dataset, HITON with Tabu shows higher accuracy than HITON (48 feature).

SVM Classifier: With 38 features in prostate cancer, HITON with Tabu shows better accuracy than HITON with 46 features. With 18 features, Leukemia dataset shows better accuracy using proposed HITON with Tabu search. In the Brain Tumor dataset, with 46 features for HITON and 45 features for HITON with Tabu displayed better accuracy. However, in the case of lung cancer dataset, accuracy using HITON with Tabu was better for 46 features.

TOP SCORING GENES:

1. **Prostate Cancer Datasets:** Using HITON + Tabu algorithm with the prostate cancer data set, accuracy achieved was 99.40 % with 38 selected genes using the SVM classifier. Selected 38 genes are:

    - leukocyte immunoglobulin-like receptor, subfamily A (with TM domain), member 2
    - Cluster Incl AF009787: Homo sapiens T cell receptor beta chain (TCRB) mRNA
    - Ric (Drosophila)-like, expressed in neurons
    - Rho GTPase activating protein 1
    - survival motor neuron pseudogene
    - leukocyte immunoglobulin-like receptor, subfamily A (with TM domain), member 1
    - Cluster Incl AL008627
    - forkhead box D2

- cytokeratin 2
- putative peroxisome microbody protein 175.1
- synovial sarcoma, X breakpoint 5
- Cluster Incl AL022097
- Cluster Incl U33547
- class II DN alpha
- Cluster Incl W26104
- Cluster Incl W27234
- Cluster Incl W27604
- Cluster Incl AL031393
- Cluster Incl AF098641
- trefoil factor 3 (intestinal)
- pancreatic elastase IIB
- dihydrofolate reductase pseudogene 1
- melanoma antigen, family A, 12
- thymosin, beta 10
- Cluster Incl X72882
- Cluster Incl M57423
- Cluster Incl Z70218
- Cluster Incl L17328
- Cluster Incl S81916
- QT syndrome 3
- caspase 8, apoptosis-related cysteine protease
- muscle specific gene
- chorionic somatomammotropin hormone 1 (placental lactogen)
- Cluster Incl D25272
- G antigen 1
- G antigen 6
- Fc fragment of IgG, low affinity IIIb, receptor for (CD16)
- Cluster Incl W26594

2. **Leukemia Cancer Datasets:** Using HITON + Tabu algorithm with the Leukemia cancer data set, accuracy achieved was 99.80 % with 18 selected genes using the SVM classifier. Selected 18 genes are:

    - interleukin 10
    - signal transducer and activator of transcription 1, 91kD
    - M10098 Human 18S rRNA gene
    - glyceraldehyde-3-phosphate dehydrogenase
    - Musashi (Drosophila) homolog
    - olfactory receptor, family 2, subfamily F, member 1
    - T cell receptor alpha variable 13-1
    - Fc fragment of IgG
    - Cluster Incl AF061055
    - dopachrome tautomerase
    - tankyrase, TRF1-interacting ankyrin-related ADP-ribose polymerase

- carboxyl ester lipase-like
- small inducible cytokine subfamily C, member 2
- basic transcription factor 3, like 3
- Cluster Incl M14087
- gamma-aminobutyric acid (GABA) B receptor
- voltage-dependent anion channel 1 pseudogene
- Cluster Incl U94902

3. **Lymphomas Cancer Datasets:** Using HITON + Tabu algorithm with the Lymphomas cancer data set, accuracy achieved was 98.72 % with 61 selected genes using the SVM classifier. Selected 61 genes are:

    - voltage-dependent anion channel 1 pseudogene
    - Cluster Incl U94902
    - zinc finger protein 259, pseudogene
    - proline-rich protein HaeIII subfamily 1
    - SRY (sex determining region Y)-box 1
    - HGC6.1.1 protein
    - interleukin 4
    - tumor necrosis factor receptor superfamily, member 6
    - transferrin receptor (p90, CD71)
    - pre-T/NK cell associated protein
    - Cluster Incl U50277
    - glycine receptor, alpha 1 (startle disease/hyperekplexia, stiff man syndrome)
    - Cluster Incl X61070
    - potassium voltage-gated channel, Shab-related subfamily, member 2
    - mannosyl (alpha-1,6-)-glycoprotein beta-1,6-N-acetyl-glucosaminyltransferase
    - bone morphogenetic protein 3
    - Cluster Incl D84143
    - ATPase, H+ transporting
    - Cluster Incl M21388
    - Cluster Incl S80491
    - immunoglobulin kappa variable 2/OR22-4
    - G protein-coupled receptor 12
    - pancreatic beta cell growth factor
    - Cluster Incl X61079
    - Cluster Incl S83374
    - Cluster Incl U82303
    - Cluster Incl U82306
    - Cluster Incl AF005081
    - Cluster Incl AF015124
    - solute carrier family 34 (sodium phosphate), member 1
    - putative opioid receptor, neuromedin K (neurokinin B) receptor-like
    - ribosomal protein S19
    - surfactant protein A binding protein
    - RIG-like 14-1

- tolloid-like 1
- G protein-coupled receptor 45
- Cluster Incl AC004076
- gonadotropin-releasing hormone 2
- putative mitochondrial space protein 32.1
- R3H domain (binds single-stranded nucleic acids) containing
- matrix metalloproteinase 20 (enamelysin)
- potassium voltage-gated channel, Shaw-related subfamily, member 3
- UDP-N-acetyl-alpha-D-galactosamine
- interleukin 1 receptor antagonist
- Cluster Incl AF063725
- airway trypsin-like protease
- double homeobox, 1
- Cluster Incl AF058075
- MHC binding factor, beta
- DNA-binding protein amplifying expression of surfactant protein B
- olfactory receptor, family 10, subfamily H, member 3
- Cluster Incl X63966
- Cluster Incl S79281
- forkhead box E2
- Cluster Incl AF072164
- ER to nucleus signalling 1
- caspase 13, apoptosis-related cysteine protease
- peptidoglycan recognition protein
- UDP glycosyltransferase 2 family, polypeptide B11
- Cluster Incl W28702
- ribosomal protein L28

4. **Lung Cancer Datasets:** Using HITON + Tabu algorithm with the Lung cancer data set, accuracy achieved was 92.36 % with 46 selected genes using the SVM classifier. Selected 46 genes are:

- Cluster Incl AF061055
- RIG-like 5-6
- Cluster Incl AF043586
- dopachrome tautomerase
- tankyrase, TRF1-interacting ankyrin-related ADP-ribose polymerase
- carboxyl ester lipase-like
- small inducible cytokine subfamily C, member 2
- basic transcription factor 3, like 3
- acetyl-Coenzyme A carboxylase beta
- Cluster Incl M14087
- gamma-aminobutyric acid (GABA) B receptor, 1
- voltage-dependent anion channel 1 pseudogene
- Cluster Incl U94902
- zinc finger protein 259, pseudogene
- proline-rich protein HaeIII subfamily 1

- SRY (sex determining region Y)-box 1
- HGC6.1.1 protein
- interleukin 4
- tumor necrosis factor receptor superfamily, member 6
- transferrin receptor (p90, CD71)
- pre-T/NK cell associated protein
- Cluster Incl U50277
- glycine receptor, alpha 1 (startle disease/hyperekplexia, stiff man syndrome)
- Cluster Incl X61070
- potassium voltage-gated channel, Shab-related subfamily, member 2
- mannosyl (alpha-1,6-)-glycoprotein beta-1,6-N-acetyl-glucosaminyltransferase
- bone morphogenetic protein 3
- Cluster Incl D84143
- ATPase, H+ transporting, lysosomal, alpha polypeptide, 70kD, isoform 2
- Cluster Incl M21388
- Cluster Incl S80491
- G protein-coupled receptor 12
- immunoglobulin kappa variable 2/OR22-4
- pancreatic beta cell growth factor
- Cluster Incl U82303
- Cluster Incl U82306
- UDP glycosyltransferase 2 family, polypeptide B11
- caspase 13, apoptosis-related cysteine protease
- Cluster Incl W28702
- ribosomal protein L28
- immunoglobulin kappa variable 1/OR15-118
- double homeobox, 2
- early lymphoid activation protein
- zinc finger protein 154 (pHZ-92)
- huntingtin-associated protein 1 (neuroan 1)
- chromosome 1 open reading frame 1

5. **Brain Tumor Datasets:** Using HITON + Tabu algorithm with the Brain Tumor data set, accuracy achieved was 97.95 % with 45 selected genes using the SVM classifier. Selected 45 genes are:

- tumor protein, translationally-controlled 1
- Kruppel-associated box protein
- Cluster Incl M96936
- melanoma antigen, family B, 3
- Cluster Incl AF005082
- Cluster Incl U96291
- ribosomal protein S8
- glutamate receptor, metabotropic 7
- Cluster Incl X72475
- solute carrier family 14 (urea transporter), member 2
- Cluster Incl AL034450

- complement factor H-related 4
- p65 protein
- U5 snRNP-specific protein, 200 kDa (DEXH RNA helicase family)
- keratin, hair, acidic,3A
- sarcosine dehydrogenase
- immunoglobulin lambda-like polypeptide 2
- eukaryotic translation initiation factor 4E binding protein 1
- galactose-4-epimerase, UDP
- melanoma antigen, family A, 6
- postmeiotic segregation increased 2-like 1
- RNA binding motif protein, Y chromosome, family 1, member C
- T-box 6
- Cluster Incl W27974
- neuroendocrine secretory protein 55
- olfactory receptor, family 10, subfamily H, member 3
- luster Incl X63966
- Cluster Incl S79281
- forkhead box E2
- Cluster Incl AF072164
- ER to nucleus signalling 1
- peptidoglycan recognition protein
- UDP glycosyltransferase 2 family, polypeptide B11
- caspase 13, apoptosis-related cysteine protease
- Cluster Incl W28702
- ribosomal protein L28
- undifferentiated embryonic cell transcription factor 1
- immunoglobulin kappa variable 1/OR15-118
- early lymphoid activation protein
- oxidative stress induced like
- zinc finger protein 154 (pHZ-92)
- huntingtin-associated protein 1 (neuroan 1)
- bone morphogenetic protein 10
- chromosome 1 open reading frame 1
- ATP-binding cassette, sub-family B (MDR/TAP), member 11

From the above discussion, it can be understood as below:

1. The results in Table 7 presents the performance accuracy of different learning classification algorithms and it is very clear that obtained accuracy by our proposed algorithm HITON with Tabu performed better or at par with almost all datasets and in most cases, results are accurate with least selected genes.
2. While one classifier performs well in one dataset, it is outperformed by another classifier in a different dataset. Therefore, there is SVM particular classifier that can maintain a high level of accuracy on all datasets.
3. Classifiers show varying levels of accuracy for individual algorithms and when one algorithm is used along with another.
4. In some cases, as we can see, with Leukemia dataset achieves 99.80% using the proposed algorithm with 18 selected genes in compared to Prostate cancer dataset which achieves 99.40% and 38 selected genes.
5. Lastly, our algorithm performs consistently well across all types of datasets including Prostate, Leukemia, Lymphomas and Brain Tumor. As visualized in Table 7, HITON with Tabu search, accuracy is better or at par with HITON algorithm, respectively.

Chapter 7
Conclusions

7.1 Conclusion

System of data wherein one set of data represents the entire class also referred to as microarray data has several characteristics and is very complex. To increase the performance of such data it is necessary that correct and proper features are selected and the data is classified according to the features. In this study, Markov Blanket and Tabu search were combined for better selection of genes and thereby increasing the prediction accuracy of cancer datasets. Such techniques are integral for feature selection as these reduce the size of the data by keeping the meaningful ones and discarding the unnecessary ones. Studies show that these feature selection methods retain the reliability of the data and give the correct result. This is highly useful in predicting cancer from microarray datasets accurately.

In this study the HITON algorithm was modified to overcome the limitations of the HITON algorithm and combined with Tabu search to obtain a robust feature selection process which resulted in better feature selection of genes to be classified by various machine learning algorithm. The results show reduction in number of selected genes and higher prediction accuracy of microarray datasets by KNN, SVM, and Neural Network classifiers. We conclude that Markov Blanket together with Tabu search provides better gene selection for accurately predicting cancer on microarray datasets.

References

[1] “Fernandez, N., Gundersen, G., Rahman, A., Grimes, M., Rikova, K., Hornbeck, P., & Ma’ayan, A. (2017). Clustergrammer, a web-based heatmap visualization and analysis tool for high-dimensional biological data. Scientific Data, 4(1). DOI: 10.1038/sdata.2017.”.

[2] Z. W. e. al, “Microarray Analysis of Gene Expression Involved in Another Development in rice (Oryza sativa L.), Plant Molecular Biology, vol. 58, no. 5, pp. 721-737, 2006. Available: 10.1007/s11103-005-8267-4.,” 2006.

[3] “http://erohde.blogspot.com/2010/09/google-chrome-advantages-and.html”.

[4] “Babu M. M., “An Introduction to Microarray Data Analysis”, Chapter – 11, P-225-227.”.

[5] “https://www.researchgate.net/publication/313277390”.

[6] “Introduction to microarray technology and pixel intensity correlation, Chapter-1”.

[7] “Hogeweg P., 2011, “The Roots of Bioinformatics in Theoretical Biology”, PLoS”.

[8] R. Quinlan, “C4.5: Programs for Machine Learning (Morgan Kaufmann Series in Machine Learning): Morgan Kaufmann, San Mateo,” 1992.

[9] “Surti A. Z., Sharma P., 2018, “Intelligent Techniques for Gene Expression Datasets”, International Journal of Emerging Technology and Advanced,” Engineering (IJETAE-2018), ISSN: 2250-2459..

[10] “Dayhoff M, Eck R., 1967–1968, “Atlas of Protein Sequence and Structure Maryland (Silver Spring)”, National Biomedical Research Foundation”, Page No. 356.”.

[11] Mount WD, 2004, “Bioinformatics: Sequence and Genome Analysis”, 2nd New York: Cold Spring Harbor Laboratory Press, Page No .692..

[12] “Protein Information Resources (PIR) [Internet], 2016, “http://pir.georgetown.edu.”.

[13] Computer Science, PediaPress GmbH Boppstrasse 64,Mainz, Germany, 2011.

[14] “Griffiths AJF, Miller JH, Suzuki DT, et al. An Introduction to Genetic Analysis. 7th edition,New York: W. H. Freeman.,” 2000.

[15] K. Raza, “Application of Data Mining in Bioinformatics, Indian Journal of Computer Science and Engineering,” 2010.

[16] “Okim Kang, Automated English Proficiency Scoring of Unconstrained Speech Using Prosodic Features”.

[17] “Aliferis, C. F., Tsamardinos, I., & Statnikov, A. HITON: a novel Markov Blanket algorithm for optimal variable selection. AMIA ... Annual Symposium proceedings. AMIA Symposium,” pp. 21-25, 2003.

[18] “Duda RO, Hart PE, and Stork DG, 2000. Pattern classification, 2nd ed. New York: John Wiley”.

[19] “Guyon, J. Weston, S. Barnhill, and V. Vapnik,Gene selection for cancer classification using support vector machines. Machine Learning, 46(1):389–422,” 2002.

[20] “Zhang, S., Li, X., Zong, M., Zhu, X., & Wang, REfficient kNN Classification With Different Numbers of Nearest Neighbors. IEEE Transactions On Neural Networks And Learning Systems,” 29(5), 1774-1785. DOI: 10.1109/tnnls.2017.2673241.

[21] “Bishop CM., 1995, “Neural Networks for Pattern Recognition”, USA: Oxford university press.”.

[22] “Andrzej Polanski, Marek Kimmel, 2007, “Bioinformatics”, Book, ISBN : 978-3-540-69022-1”.

[23] “Guyon, C. Aliferis, and A. Elisseeff,"Causal feature selection," Computational methods of feature selection,” pp. pp. 63-82, 2007.

[24] “Tsamardinos, C. F. Aliferis, and A. Statnikov,Time and sample the efficient discovery of Markov blankets and direct causal relations. Proceedings of the Ninth International Conference on Knowledge Discovery and Data Mining (KDD), 2003,” p. pages 673–678, 2003.

[25] J. Pearl, “Probabilistic reasoning in intelligent systems: networks of plausible inference. Elsevier,” 2014.

[26] “I. Tsamardinos, L. E. Brown, and C. F. Aliferis,"The max-min hill-climbing bayesian network structure learning algorithm". Machine Learning, 65(1):31–78,” 2006.

[27] “Azadeh Mohammadi, Mohammad Saraee, Mansoor Salehi, 2011, “Identification of Disease-Causing Genes using Microarray Data Mining and Gene Ontology”, BMC Medical Genomics.”.

[28] “Daxin Jiang Chun Tang Aidong Zhang, “Cluster Analysis for Gene Expression Data: A Survey”, IEEE Transactions on Knowledge and Data Engineering.”.

[29] D. K. a. M. Sahami, “D. Koller and M. Sahami, "Toward optimal feature selection," in "toward optimal feature selection," Stanford InfoLab 1996,” *Stanford InfoLab,* 1996.

[30] “L. Yu, C. Ding, and S. Loscalzo. (2008) Stable Feature Selection via Dense Feature Groups. KDD’08, 803-811.”.

[31] “K. Yu, L. Liu, and J. Li. "Discovering markov blanket from multiple interventional datasets". arXiv preprint arXiv:1801.08295,” 2018.

[32] “I. Tsamardinos, L. E. Brown, and C. F. Aliferis,"The max-min hill-climbing bayesian network structure learning algorithm". Machine Learning, 65(1):31–78”.

[33] “Thrun, D. Margaritis and S."Bayesian network induction via local neighborhoods". Advances in Neural Information Processing Systems,” p. 12:505–511, 1999.

[34] “Pena, J.M., et alTowards scalable and data-efficient learning of Markov boundaries. International Journal of Approximate Reasoning,” 2007.

[35] “Xue Bai Tabu Search Enhanced Markov Blanket Classifier for High Dimensional Data Sets, January 2005, CMU-CALD-05-101”.

[36] “Hongbin Zhang,Feature selection using tabu search method,9 Beijing 100044, People's Republic of China,DEC 2000.”.

[37] “Idown Olayinka Oduntan, A Multilevel Search Algorithm For Feature Selection in Biomedical Data,Canada 2005”.

[38] F. Glover, “Future paths for integer programming and links to artificial intelligence,” Computers&Operations Research, vol. 13, no. 5, pp. 533–549., 1986, pp. vol. 13, no. 5, pp. 533–549.

[39] “Zuerey, David Schindl and Nicolas,Solution methods for fuel supply of trains. INFOR, 51(1):23-30.,” 2013.

[40] “Birattari, Marco Dorigo and Mauro,"Ant colony optimization. In Encyclopedia of machine learning", pages 36-39. Springer.,” 2010.

[41] Guyon and A. Elisseeff, "An introduction to variable and feature selection," Journal of machine learning research, vol. 3, no. Mar, pp. 1157-1182, 2003.

[42] J. Novakovic, “The Impact of Feature Selection on the Accuracy of Naive Bayes Classifier, 18th Telecommunications forum TELFOR,” 2010.

[43] “K. Das et al, 2016, “Informative Gene Selection for Effective Classification of Gene Expression Datasets”, PhD Thesis, Shiksha o Anusandhan University.”.

[44] “Hall, H. Witten and M. A."Data mining: practical machine learning tools and techniques", Amsterdam; Boston, Morgan Kaufmann,” 2011.

[45] “H. Liu and R. Setiono, Chi2.Feature Selection and Discretization of Numeric Attributes, Proceedings of the Seventh International Conference on Tools with Artificial Intelligence,” 1995.

[46] R. Holte, “Very Simple Classification Rules Perform Well on Most Commonly Used Datasets, Machine Learning,” pp. 63-90, (1993) .

[47] “Rendell, K. Kira and L. A."A practical approach to feature selection, Proceedings of the ninth international workshop on Machine learning", Aberdeen, Scotland, United Kingdom,” 1992.

[48] T. Mitchell, “Machine learning,New York: McGraw-Hill Co., Inc.,” 1997.

[49] “Thomas, T. M. Cover and J. A.Elements of information theory, 2nd ed. Hoboken, N.J.: Wiley-Interscience.,” 2006.

[50] “Langley, G. John and P. Estimating Continuous Distributions in Bayesian Classifiers, Eleventh Conference on Uncertainty in Artificial Intelligence,” 1995, pp. 338-345.

[51] F. Rosenblatt, “Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms, Spartan,” 1962.

[52] “D. Nam, S. Y. Kim,"Gene-Set Approach for Expression Pattern Analysis",” *Briefings in Bioinformatics, 9 (3),* p. Page No. 189–197, 2008.

[53] “J. Ferreiraa, M. A. T. Figueiredo, 2014, “Incremental Filter and Wrapper Approaches for Feature Discretization”, Neurocomputing, 123, Page No. 60–74.”.

[54] “C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, "A Survey on Filter Techniques for Feature Selection in Gene Expression Microarray Analysis”,” *IEEE/ACM TransactionTransactions on Computational Biology and Bioinformatics,* pp. Page No. 1106-1119, 2012.

[55] “R. Ruiz, J. C. Riquelme, J. S. Aguilar-Ruize, M. Garcia-Torres,"Fast Feature Selection Aimed at High-Dimensional Data via Hybrid-Sequential-Ranked".,” pp. Page No. 11094-11102., 2012.

[56] “D. Du, K. Li, X. Li, M. Fei, 2014, “A Novel Forward Gene Selection Algorithm for Microarray Data”, Neurocomputing, 133, Page No. 446–458.”.

[57] “M. Zhu, J. Song,"An Embedded Backward Feature Selection Method for MCLP Classification Algorithm", Procedia Computer Science, 17,” 2013, p. Page No. 1047 – 1054.

[58] “Kristian Larsen,Variable Selection Methods: Lasso and Ridge Regression in Python 2019”.

[59] A. Rokotomamonjy, “Perception, Systemes et Information, FRE CNRS 2645, INSA deRouen 76801 Saint Etienne du Rourav, France,” 2003.

[60] “Guyon, J. Weston, S. Barnhill, and V. Vapnik,Gene selection for cancer classification using support vector machines. Machine Learning, 46(1):389–422,” 2003.

[61] “Peralta, A. Soto,Embedded Local Feature Selection within Mixture of Experts, Information Sciences, 269,” 2014, p. Page No. 176–187.

[62] “P. K. A. Mundra, J. C. Rajapakse. "SVM-RFE with m-RMR Filter for Gene Selection”, IEEE Transactions on Nanobioscience, 9(1).,” p. Page No. 31–37., 2010.

[63] “P. Somol, J. Novovicova,Evaluating Stability and Comparing Output of Feature Selectors that Optimize Feature Subset Cardinality,” pp. Page No. 1921-1939, 2010.

[64] “Yuchun Tang,Granular Support Vector Machines Based on Granular Computing, Soft Computing and Statistical Learning,” 2006.

[65] “Chenguang Zhao & Zheng Wang,GOGO: An improved algorithm to measure the semantic similarity between gene ontology terms”.

[66] “C. E. Gillies, M. R. Siadat, N. V. Patel , G. D. Wilson,A Simulation to Snalyze Feature Selection Methods Utilizing Gene Ontology for Gene Expression Classification,” *Journal of Biomedical Informatics, 46,* p. Page No. 1044–1059, 2013.

[67] “L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," Journal of machine learning research, vol. 5, no. Oct, pp. 1205-1224, 2004.”.

[68] “John, R. Kohavi and G. H. Wrappers for feature subset selection,Artificial intelligence,” pp. vol. 97, no. 1-2, pp. 273-324, 1997.

[69] “Peng, Ding and H.,Minimum redundancy feature selection from microarray gene expression data, Journal of bioinformatics and computational biology,” 2005, pp. vol. 3, no. 02, pp. 185-205.

[70] “J Am Med,The center for causal discovery of biomedical knowledge from big data.,” 2015.

[71] “Fu, S.-K. and M.C. Desmarais,Fast Markov Blanket Discovery Algorithm Via Local Learning within Single Pass. in Canadian Conference on AI. Windsor, Canada: Springer.,” 2008.

[72] J. Pearl, “Probabilistic reasoning in expert systems, San Matego: Morgan Kaufmann.,” 1988.

[73] “Spirtes, P., C. Glymour, and R. Scheines,Causation, Prediction and Search (2nd Edition): The MIT Press.,” 2001.

[74] “Yaramakala, S. and D. Margaritis,"Speculative Markov Blanket Discovery for Optimal Feature Selection". in ICDM.,” 2003.

[75] “Local Causal and Markov Blanket Induction for Causal Discovery and Feature Selection for Classification Part I: Algorithms and Empirical Evaluation,” *Journal of Machine Learning Research 11,* 2010.

[76] J. Pearl, “Causality: Models, Reasoning, and Inference: Cambridge University Press.,” 2000.

[77] “Lucien Mousin, Laetitia Jourdan, Marie-Eléonore Marmion, Clarisse Dhaenens, Feature Selection using Tabu Search with Learning Memory: Learning Tabu Search.,” 2016.

[78] “Zufferey, David Schindl and Nicolas,Solution methods for fuel supply of trains. INFOR, 51(1):23{30,” 2013.

[79] “Glover, F. Tabu Search. Kluwer Academic Publishers.,” 1997.

[80] “B. R. Kowalski and C. Bender, "k-Nearest Neighbor Classification Rule (pattern recognition) applied to nuclear magnetic resonance spectral interpretation," Analytical Chemistry, vol. 44, pp. 1405-1411, 1972”.

[81] “Aghdam, H. H., & Heravi, E. J. (2017). Guide to convolutional neural networks: a practical application to traffic-sign detection and classification. Springer.”.

[82] “N. S. Altman, ‘‘An introduction to kernel and nearest-neighbor nonparametric regression,’’ Amer. Statist., vol. 46, no. 3, pp. 175–185, 1992.”.

[83] “Bramer, M. (2013). Principles of data mining, second edition. London: Springer.”.

[84] “Jaime Salvador–Meneses,Compressed kNN: K-Nearest Neighbors with Data Compression.”.

[85] “blog.webkid.io”.

[86] “J. Kennedy and R. Eberhart, ‘‘Particle swarm optimization,’’ in Proc. ICNN Int. Conf. Neural Netw., Dec. 1995, pp. 1942–1948”.

[87] “Boser, Bernard E., Isabelle M. Guyon, and Vladimir N. Vapnik. “A Training Algorithm for Optimal Margin Classifiers.” In COLT ’92: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, edited by David Haussler, 144–152. New York.”.

[88] “Empirical inference. Festschrift in honor of Vladimir N. Vapnik. Schölkopf, Bernhard, Zhiyuan Luo, and Vladimir Vovk, Springer Science & Business Media, 2013”.

[89] “J. Han, J. Pei, and M. Kamber, Data mining: concepts and techniques. Elsevier,” 2011.

[90] “Source: https://diagram.ca/”.

[91] “Sonu Kumar, Sneha Roy,Score Prediction and Player Classification Model in the Game of Cricket Using Machine Learning”.

[92] “A. Statnikov, I. Tsamardinos, Y. Dosbayev, and C. F. Aliferis, "dataset ref:34 GEMS: a system for automated cancer diagnosis and biomarker discovery from microarray gene expression data," International journal of medical informatics, vol. 74, no. 7,” pp. 491-503, 2005..