

# Chapter 1

## 1.1 Introduction

Feature selection has become the focus of many research areas in recent years. With the rapid advance of computer and database technologies, datasets with large number of variables or features may lead to reduce the performance of data mining, and machine learning. Machine learning provides tools by which large quantities of data can be automatically analyzed is feature selection. Feature selection process can be beneficial to a variety of common machine learning algorithms. The feature selector is simple and fast to execute. It eliminates irrelevant and redundant data and, in many cases, improves the performance of learning algorithms, reduces the computational cost and provides better understandings of the dataset. Feature selection or feature selection identifies the relevant feature which are useful to the data mining task. In this thesis we considered breast cancer dataset for experimental purpose to compare three feature selection methods, Correlation Based feature Selection (CFS), RELIEF, and Wrapper Methods. To find out feature selection method may lead to improve the performance of learning algorithms in other word identified classification technique affected by applying feature selection that enhanced its accuracy and therefore reduce time to build learning model or degrades its accuracy on breast cancer dataset.

## 1.2 Motivation

Some dataset consist of large number of features or redundant and irrelevant features can lead to reduce machine learning algorithms performance. If, the data is suitable for machine learning, then the task of learning can be made easier and less time consuming by removing features of the data that are irrelevant or redundant with the task to be learned. The benefits of feature selection for learning can include a reduction in the amount of data needed to achieve learning, improve predictive accuracy, learned

knowledge that is easily understood, reduce cost of disk storage, the size of databases have grown.

### **1.3 Problem statement**

There are many feature selection method no one of these method is optimal. Moreover, there is no well known agreed upon way to categorize them there for, there is a need for more comparison studies.

### **1.4 Research Objectives**

This research provides comparison of three supervised machine learning algorithms before and after applying feature selection techniques on data.

The thesis aimed to:

1. Identify feature selection methods that enhance classification algorithms accuracy and thus improve their performance and also Identify which one of selected learning classifier is achieved highest and significant accuracy.

### **1.5 Outline**

The thesis has been organized in four chapters. Chapter 2 provides an overview of concepts from supervised machine learning and also provides overview of feature selection techniques for machine learning. Chapter 3 presents the experimental methodology and results followed by conclusion and recommendation in chapter 4.

# Chapter 2

## 2.1 Introduction

This chapter provides an overview of concepts of Knowledge Data Discovery (KDD) in section 2.2. Section 2.3 reviews data preprocessing. While section 2.4 provides overview of Supervised Machine Learning, section 2.5 provides overview of classification. Classification Techniques we reviewed in section 2.6.1 in depth description has been given to MLP (Artificial Neural Networks), J48 (Decision Tree) and Naïve Base (Bayesian Network) techniques as they are used in thesis experiments. Provides an overview of concepts of Dimensionality Reduction in section 2.7. Section 3.3 provides overview of Feature Subset Selection, while section 2.8 methods for feature subset selection, section 2.9, Characteristics of Feature Selection Algorithms in section 2.10, section 2.11 sections 2.12 overview of Feature Subset.

## 2.2 Knowledge Data Discovery

Knowledge Data Discovery (KDD) is a process of deriving hidden knowledge from databases. KDD consists of several phases like Data selection, Data Pre processing, Data integration, Data transformation, Data mining, Pattern evaluation, Knowledge representation. Data mining is one of the important phases of knowledge data discovery. Data mining is a technique which is used to find new hidden and useful patterns of knowledge from large databases. There are several data mining functions such as [1]:

1. Association Rules
2. Classification
3. Prediction
4. Clustering

## 5. Sequence discovery.

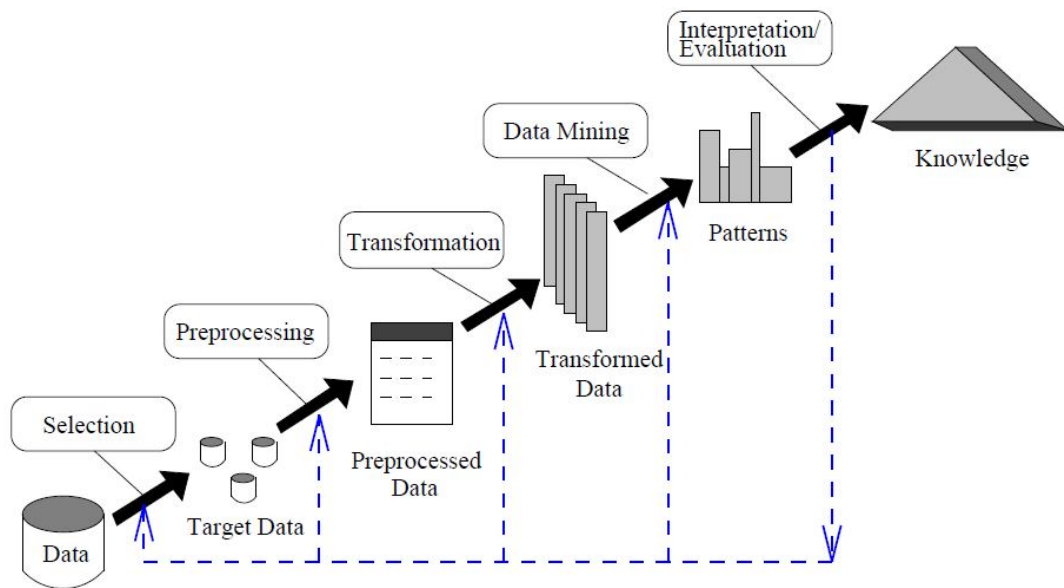


Figure 2.1 knowledge discovery process [1].

## 1. Data Preprocessing

Data preprocessing is applied before data mining to improve the quality of the data. Data preprocessing includes data cleansing, data integration, data transformation and data reduction techniques. **Cleansing**: is used to remove noisy data and missing values, **Integration** is used to extract data from multiple sources and storing as a single repository. **Transformation** transforms and normalizes the data in a consolidated form suitable for mining. **Reduction** reduces the data by adopting various techniques such as: Aggregating the data, Dimensionality reduction, Numerosity reduction and Generation of concept hierarchies and Feature Subset Selection [1].

### 2.3.1 Feature Selection

Data sets for analysis may contain hundreds of feature, many of which may be irrelevant to the mining task or redundant. To pick out some of the useful feature, this can be a

difficult and time consuming task. Keeping irrelevant feature may be detrimental, causing confusion for the mining algorithm employed. This can result in discovered patterns of poor quality. In addition, the added volume of irrelevant or redundant feature can slow down the mining process [1].

## **2.4 Supervised Machine Learning**

Data mining algorithms can follow three different learning approaches:

1. Supervised learning.
2. Unsupervised learning.
3. Semi supervised learning.

Each instance is described by a fixed number of measurements, or features, along with a label that denotes its class. In supervised learning, the algorithm work with a set of examples whose labels are known. The features are either continuous, when the feature values are ordered, or categorical when the feature values are unordered. The labels can be nominal values in the case of the classification task, or numerical values in the case of the regression task [1].

### **1. Classification**

The classification task can be seen as a supervised technique where each instance belongs to a class, which is indicated by the value of a special goal feature or simply the class feature. The goal feature can take on categorical values, each of them corresponding to a class. Each example consists of two parts, namely a set of predictor feature values and a goal feature value. The former are used to predict the value of the latter. The predictor feature should be relevant for predicting the class of an instance. In the training phase the algorithm has access to the values of both predictor feature and the goal feature for all examples of the training set, and it uses that information to build a classification model.

This model represents classification knowledge – essentially, a relationship between predictor feature values and classes – that allows the prediction of the class of an example given its predictor feature values. One of the major goals of a classification algorithm is to maximize the predictive accuracy obtained by the classification model [1]

## **2.5.1 Classification Techniques**

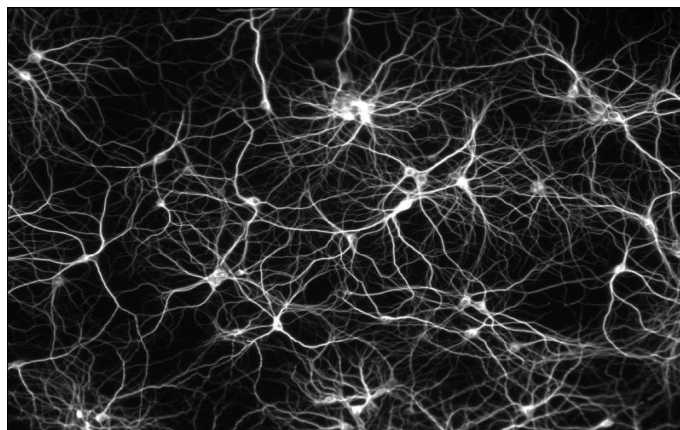
There are several classification techniques such as, Artificial Neural Networks, Decision Tree, Bayesian Network, Support Vector Machines, Association Rule and Distance Based Methods.

More details about Artificial Neural Networks, Decision Tree and Bayesian Network classification Techniques.

## **2.6 Artificial Neural Network**

### **2.6.1 Introduction**

The concept of Artificial Neural Networks (ANN) is basically introduced from the subject of biology where neural network plays an important and key role in human body. In human body work is done with the help of neural network. Neural Network is just a web of inter connected neurons which are millions and millions in number. With the help of these interconnected neurons all the parallel processing is done in human body and the human body is the best example of Parallel Processing [2].

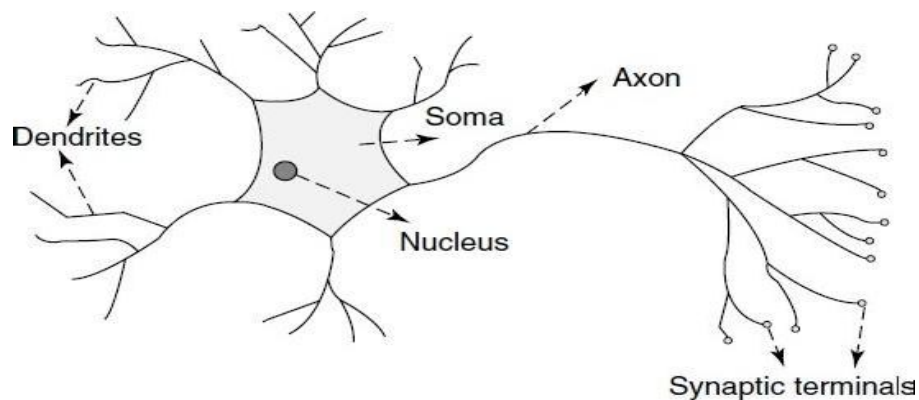


**Figure 2.2** Neural Network in Human Body [2]

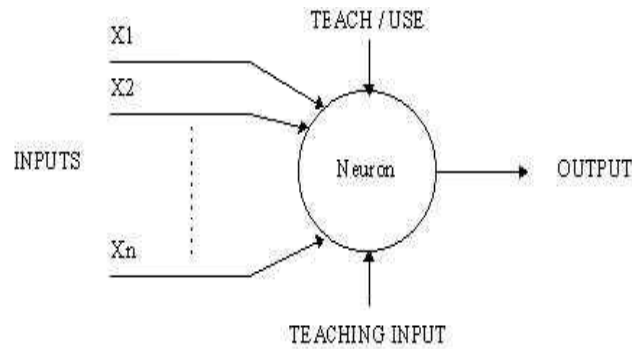
## 2.6.2 Artificial Neural Network Definition

An artificial neural network, often just called a neural network is a mathematical model or computational model based on biological neural networks, in other words, is an emulation of biological neural system. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase [3].

A Neural Network Classifier is based on neural networks consisting of interconnected neurons. From a simplified perspective, a neuron takes positive and negative stimuli (numerical values) from other neurons and when the weighted sum of the stimuli is greater than a given threshold value, it activates itself. The output value of the neuron is usually a non-linear transformation of the sum of stimuli. In more advanced models, the non-linear transformation is adapted by some continuous functions [2].



**Figure 2.3** Human Neurons [2]



**Figure 2.4** Artificial Neuron [2]

### 2.6.3 Training of Artificial Neural Networks

A neural network has to be configured such that the application of a set of inputs produces (either 'direct' or via a relaxation process) the desired set of outputs. Various methods to set the strengths of the connections exist. One way is to set the weights explicitly, using a priori knowledge. Another way is to **'train' the neural network** by feeding it teaching patterns and letting it change its weights according to some learning rule. We can categorize the learning situations as follows [3]:

1. **Supervised learning** or Associative learning in which the network is trained by providing it with input and matching output patterns. These input-output pairs can be provided by an external teacher, or by the system which contains the neural network (self-supervised).
  2. **Unsupervised learning** or Self-organization in which an (output) unit is trained to respond to clusters of pattern within the input. In this paradigm the system is supposed to discover statistically salient features of the input population. Unlike the supervised learning paradigm, there is no a priori set of categories into which the patterns are to be classified; rather the system must develop its own representation of the input stimuli.
1. **Reinforcement Learning** This type of learning may be considered as an intermediate form of the above two types of learning. Here the learning machine does

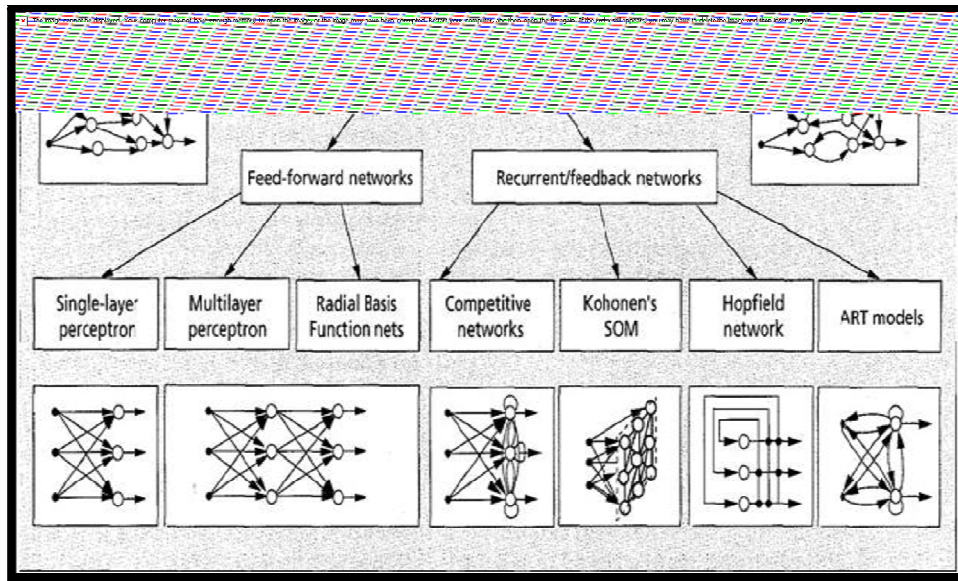


some action on the environment and gets a feedback response from the environment. The learning system grades its action good (rewarding) or bad (punishable) based on the environmental response and accordingly adjusts its parameters.

## 2.6.4 Network Architectures

1. **Feedforward neural network:** The feedforward neural network was the first and arguably simplest type of artificial neural network devised. In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. The data processing can extend over multiple (layers of) units, but no feedback connections are present, that is, connections extending from outputs of units to inputs of units in the same layer or previous layers [2].
2. **Recurrent network:** Recurrent neural networks that do contain feedback connections. Contrary to feedforward networks, recurrent neural networks (RNs) are models with bi-directional data flow. While a feedforward network propagates data linearly from input to output, RNs also propagate data from later processing stages to earlier stages [2].

There are further divisions of Feedback and Feed Forward Network architecture which are shown in below Figure: -



**Figure 2.5** Network Architecture [2].

## 2.7 Types of Neural Networks

There are wide variety of neural networks and there architecture. Types of neural networks range from simple Boolean networks (perceptions) to example self-organizing networks (kohonen networks). There are also other types of networks like Hopfield Networks, Pluse Networks, Radial-Basis Function Networks, Boltzmann Machine. The most important class of neural networks for real world problems solving include [4].

1. Multilayer Perceptron
2. Radial-Basis Function Networks
3. Kohonen Self-Organizing Feature Maps

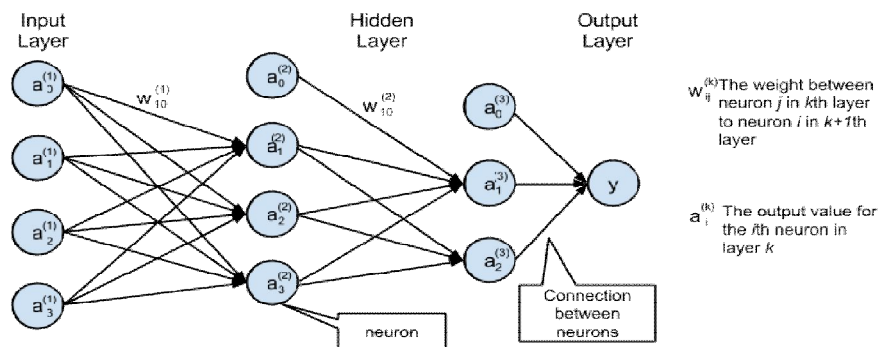
## 2.8 Multilayer perceptron

A multilayer perceptron (MLP) is a kind of Too feedforward artificial neural network, which is a mathematical model inspired by the biological neural network. The multilayer perceptron can be used for various machine learning tasks such as classification and

regression. MLP utilizes a supervised learning technique called backpropagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable

The basic component of a multilayer perceptron is the neuron. In a multilayer perceptron, the neurons are aligned in layers and in any two adjacent layers the neurons are connected in pairs with weighted edges. A practical multilayer perceptron consists of at least three layers of neurons, including one input layer, one or more hidden layers, and one output layer.

The size of input layer and output layer determines what kind of data a MLP can accept. Specifically, the number of neurons in the input layer determines the dimensions of the input feature; the number of neurons in the output layer determines the dimension of the output labels. Typically, the two-class classification and regression problem requires the size of output layer to be one, while the multi-class problem requires the size of output layer equals to the number of classes. As for hidden layer, the number of neurons is a design issue. If the neurons are too few, the model will not be able to learn complex decision boundaries. On the contrary, too many neurons will decrease the generalization of the model. Figure 2.9 an example of MLP [5].



**Figure 2.6** Multilayer Perceptron Network (MLP) [5]

## 2.8.1 The Backpropagation Algorithm

**Backpropagation**, or **propagation of error**, is a neural network learning algorithm is a common method of teaching artificial neural networks how to perform a given task. The back propagation algorithm is used in layered feedforward ANNs. This means that the artificial neurons are organized in layers, and send their signals “forward”, and then the errors are propagated backwards. The back propagation algorithm uses supervised learning, which means that we provide the algorithm with examples of the inputs and outputs we want the network to compute, and then the error (difference between actual and expected results) is calculated. The idea of the back propagation algorithm is to reduce this error, until the ANN learns the training data [5].

## 2.9 Decision Tree

A Decision Tree Classifier consists of a decision tree generated on the basis of instances. The decision tree has two types of nodes:

- a) The root and the internal nodes.
- b) The leaf nodes.

The root and the internal nodes are associated with feature; leaf nodes are associated with classes. Basically, each non-leaf node has an outgoing branch for each possible value of the feature associated with the node. To determine the class for a new instance using a decision tree, beginning with the root, successive internal nodes are visited until a leaf node is reached. At the root node and at each internal node, a test is applied. The outcome of the test determines the branch traversed, and the next node visited. The class for the instance is the class of the final leaf node [6].

### 2.9.1 J48 algorithm

J48 algorithm is based on decision tree it is an open source Java implementation of the C4.5 algorithm in the Weka data mining tool. C4.5 is a program that creates a decision

tree based on a set of labeled input data. This algorithm was developed by Ross Quinlan. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier [6].

## **2.10 Bayesian Networks**

A Bayesian network (BN) consists of a directed, acyclic graph and a probability distribution for each node in that graph given its immediate predecessors [14]. A Bayes Network Classifier is based on a Bayesian network which represents a joint probability distribution over a set of categorical feature. It consists of two parts, the directed acyclic graph  $G$  consisting of nodes and arcs and the conditional probability tables. The nodes represent feature whereas the arcs indicate direct dependencies. The density of the arcs in a BN is one measure of its complexity. Sparse BNs can represent simple probabilistic models (e.g., naïve Bayes models and hidden Markov models), whereas dense BNs can capture highly complex models. Thus, BNs provide a flexible method for probabilistic modeling [7].

### **2.10.1 Naive Bayes**

The naive Bayes algorithm employs a simplified version of Bayes formula to decide which class a novel instance belongs to. The posterior probability of each class is calculated, given the feature values present in the instance; the instance is assigned the class with the highest probability.

In simple terms, a naive Bayes classifier assumes that the presence or absence of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 3" in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of the presence or absence of the other features. For some types of

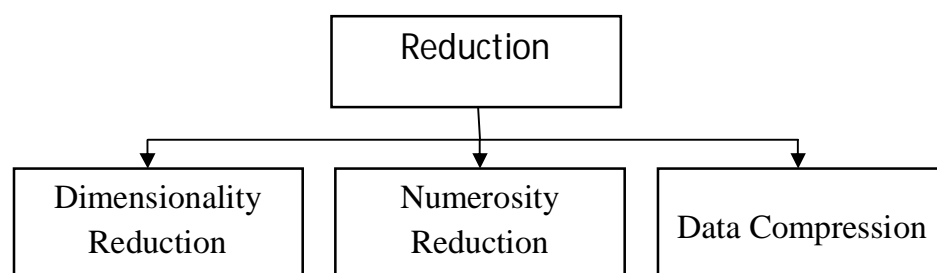
probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods. Despite their naive design and apparently oversimplified assumptions, naive Bayes classifiers have worked quite well in many complex real-world situations [1]. In 2004, an analysis of the Bayesian classification problem showed that there are sound theoretical reasons for the apparently implausible efficacy of naive Bayes classifiers[1] Still, a comprehensive comparison with other classification algorithms in 2006 showed that Bayes classification is outperformed by other approaches, such as boosted trees or random forests[8].

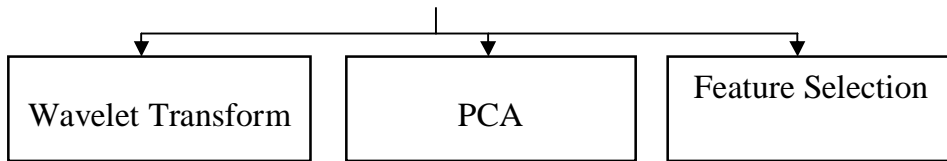
An advantage of Naive Bayes is that it only requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix [9].

## 2.11 Data reduction

**Data reduction** techniques can be applied to obtain a reduced representation of the Data set that is much smaller in volume, yet closely maintains the integrity of the original data. That is, mining on the reduced data set should be more efficient yet produce the same or better analytical results [1].

Data reduction strategies include dimensionality reduction, numerosity reduction, and data compression.





**Figure 2.7** Data Reduction techniques

## 2.12 Dimensionality Reduction

Dimensionality reduction is one of data reduction techniques it is the process of reducing the number of random variables or feature under consideration. Dimensionality reduction methods include [1]:

1. Wavelet transforms and principal components analysis (PCA), which transform or project the original data onto a smaller space.

Feature subset selection is a method of dimensionality reduction in which irrelevant, weakly relevant or redundant feature or dimensions are detected and removed.

### 2.12.1 Feature Selection

Feature subset selection (in machine learning is known as feature subset selection) reduces the data set size by removing irrelevant or redundant feature (or dimensions) [1].

**The goal and benefit of feature subset selection is [1]:**

1. Find a minimum set of feature such that the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all feature.
2. It reduces the number of feature appearing in the discovered patterns, helping to make the patterns easier to understand.

To find a ‘good’ subset of the original feature for n feature, there are  $2^n$  possible subsets.

1. Heuristic methods that explore a reduced search space are commonly used for feature subset selection.
2. These methods are typically **greedy** in that, while searching through feature space, they always make what looks to be the best choice at the time.
3. Their strategy is to make a locally optimal choice in the hope that this will lead to a globally optimal solution. Such greedy methods are effective in practice and may come close to estimating an optimal solution.

**The “best” (and “worst”) feature is typically determined using [1]:**

1. Tests of statistical significance, which assume that the feature are independent of one another.
2. Many other feature evaluation measures can be used such as:
  1. The information gain measure used in building decision trees for classification.
  2. Basic heuristic methods of feature subset selection include the techniques that follow, some of which are illustrated in Figure 2.5.

## **2.13 Feature Selection Heuristic Methods**

Feature selection can be applied by using many methods that include Forward Selection, Backward Elimination, Combination of Forward and Backward and Decision Tree Induction [1].

### **2.13.1 Forward Selection**

The procedure starts with an empty set of feature as the reduced set. The best of the original feature is determined and added to the reduced set. At each subsequent iteration or step, the best of the remaining original feature is added to the set [1].

### **2.13.2 Backward Elimination**



The procedure starts with the full set of feature. At each step, it removes the worst feature remaining in the set [1]

### **2.13.3 Combination of Forward and Backward**

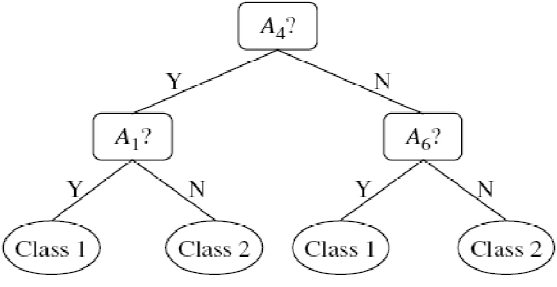
The stepwise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best feature and removes the worst from among the remaining feature [1].

### **2.13.4 Decision Tree Induction**

Decision tree algorithms (e.g., ID3, C4.5, and CART) were originally intended for classification. Decision tree induction constructs a flowchart like structure where each internal (nonleaf) node denotes a test on an feature, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the “best” feature to partition the data into individual classes. When decision tree induction is used for feature subset selection, a tree is constructed from the given data.

All features that do not appear in the tree are assumed to be irrelevant. The set of feature appearing in the tree form the reduced subset of feature.

The stopping criteria for the methods may vary. The procedure may employ a threshold on the measure used to determine when to stop the feature selection process. [1]

Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  Initial reduced set: $\{\}$ $\Rightarrow \{A_1\}$ $\Rightarrow \{A_1, A_4\}$ $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  $\Rightarrow \{A_1, A_3, A_4, A_5, A_6\}$ $\Rightarrow \{A_1, A_4, A_5, A_6\}$ $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$    $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$

**Figure 2.8** Feature Selection Methods [1].

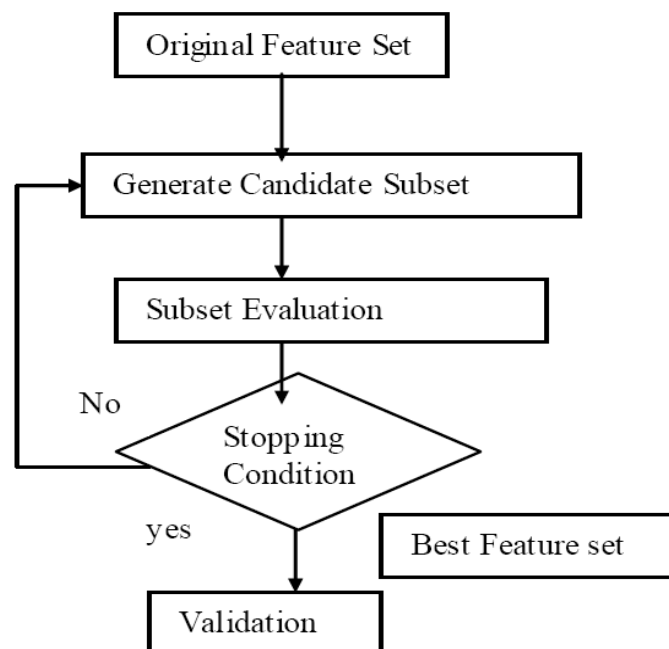
## 2.14 Characteristics of Feature Selection Algorithms

Feature selection algorithms (with a few notable exceptions) perform a search through the space of feature subsets, and, as a consequence, must address four basic issues affecting the nature of the search [10]:

1. Starting point. Selecting a point in the feature subset space from which to begin the search can affect the direction of the search. One option is to begin with no features and successively add feature. In this case, the search is said to proceed forward through the search space. Conversely, the search can begin with all features and successively remove them. In this case, the search proceeds backward through the search space. Another alternative is to begin somewhere in the middle and move outwards from this point.
2. Search organization. An exhaustive search of the feature subspace is prohibitive for all but a small initial number of features. With  $N$  initial features there exist possible subsets. Heuristic search strategies are more feasible than exhaustive ones

and can give good results, although they do not guarantee finding the optimal subset.

3. Evaluation strategy. How feature subsets are evaluated is the single biggest differentiating factor among feature selection algorithms for machine learning. operates independent of any learning algorithm—undesirable features are filtered out of the data before learning begins. These algorithms use heuristics based on general characteristics of the data to evaluate the merit of feature subsets. Another school of thought argues that the bias of a particular induction algorithm should be taken into account when selecting features. Induction algorithm along with a statistical re-sampling technique such as cross-validation to estimate the final accuracy of feature subsets.
4. Stopping criterion. A feature selector must decide when to stop searching through the space of feature subsets. Depending on the evaluation strategy, a feature selector might stop adding or removing features when none of the alternatives improves.



**Figure 2.9** Steps for feature selection [10].

## **2.15 Feature Selection Evaluator Methods**

Feature selection has many variety methods include: Correlation-based Feature Selection (CFS), Classifier, Consistency, Filter, Wrapper, Gain Ratio, Principal Components Analysis (PCA), Info Gain and Relief Evaluators.

### **2.15.1. Correlation-based Feature Selection (CFS)**

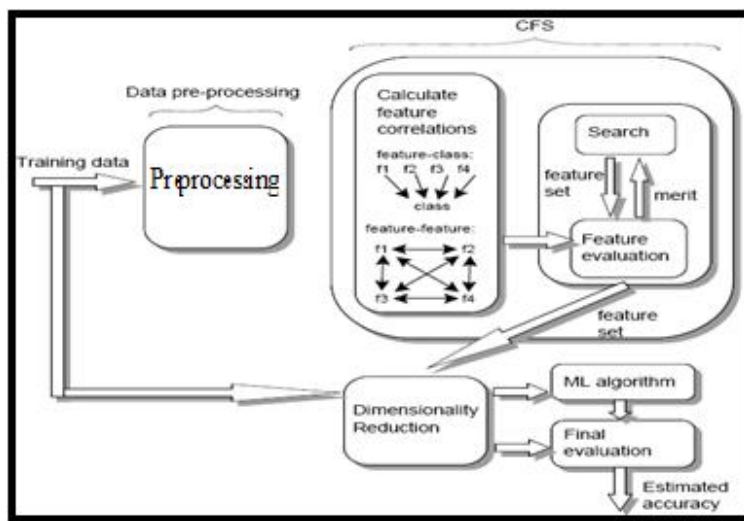
Correlation-based Feature Selection (CFS) assesses the predictive ability of each feature individually and the degree of redundancy among them, preferring sets of feature that are highly correlated with the class but have low intercorrelation. An option iteratively adds feature that has the highest correlation with the class, provided that the set does not already contain a feature whose correlation with the feature in question is even higher. Missing can be treated as a separate value, or its counts can be distributed among other values in proportion to their frequency [8].

The subset evaluators use a numeric measure, such as conditional entropy, to guide the search iteratively and add features that have the highest correlation with the class. CFS evaluates the worth of a subset of feature by considering the individual predictive ability of each feature along with the degree of redundancy between them. Correlation coefficient is used to estimate correlation between subset of feature and class, as well as inter-correlations between the features.

Relevance of a group of features grows with the correlation between features and classes, and decreases with growing inter-correlation. CFS is used to determine the best feature subset and is usually combined with search strategies such as forward selection, backward elimination, bi-directional search, best-first search and genetic search. Equation for CFS is given [10].

$$r_{zc} = \frac{\overline{kr_{zi}}}{\sqrt{k + k(k-1)r_{ij}}}$$

Where  $r_{zc}$  is the correlation between the summed feature subsets and the class variable,  $k$  is the number of subset features,  $r_{zi}$  is the average of the correlations between the subset features and the class variable, and  $r_{ij}$  is the average inter-correlation between subset features [10].



**Figure 3.5** Correlation-based Feature Selection Method (CFS) [10].

### 2.15.2 Wrapper Evaluator Method

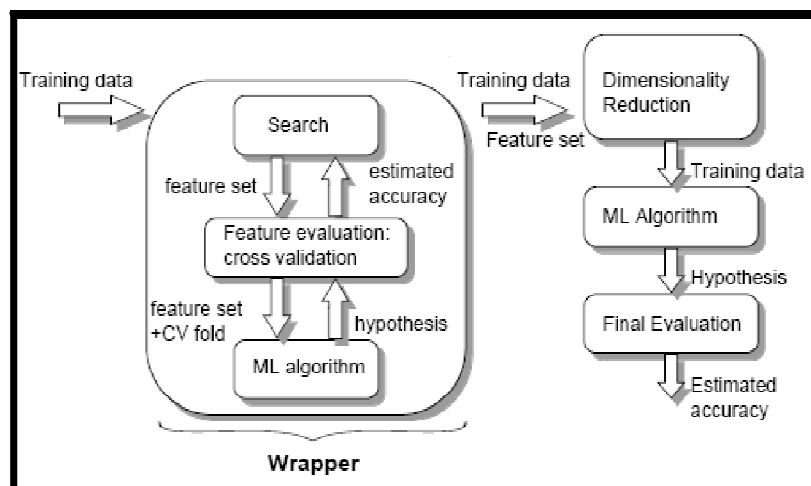
Wrapper evaluator method is to evaluate the feature using the machine learning algorithm to measure the importance of features set that will ultimately be employed for learning, but it employs cross-validation to estimate the accuracy of the learning scheme for each set [8].

This method called wrapper because the learning algorithm is wrapped into the selection procedure. Making an independent assessment of a feature subset would be easy if there were a good way of determining when a feature was relevant to choosing the class.

However, there is no universally accepted measure of “relevance,” although several different ones have been proposed [8].

Wrapper methods generally result in better performance than other methods because the feature selection process is optimized for the classification algorithm to be used [10].

Figure 3.6 shows steps execution of the wrapper method.



**Figure 3.6** wrapper feature selectors [10].

### 2.15.3 RELIEF Evaluator Method

Is instance-based it samples instances randomly and checks neighboring instances of the same and different classes. Parameters specify the number of instances to sample, the number of neighbors to check, whether to weight neighbors by distance, and an exponential function that governs how rapidly weights decay with distance [8].

## **Chapter 3**

### **Experiments and Results**

#### **3.1 Introduction**

This chapter provides an overview of Dataset description used in all experiments in section 3.2. Section 3.3 provides overview of weka tool, Experiments Methodology reported in section 3.4. While section 3.5 shows results of Experiments followed by discussion in section 3.6.

#### **3.2 Dataset Description**

The dataset used in these experiments is breast cancer dataset, was chosen from Seer research data for public use. It downloaded from The Seer library Website [8]. It is represented in "csv" file format.

Pre-processing is an important step that is used to transform the data into a format that suitable to apply data mining techniques and maximizes its output. This dataset contains 20300 instances and 134 features reduced to 24 features. There are 22 nominal features and tow numeric features. The nominal features range from 2 to 23 values.

As a first step, non breast cancer related feature identified and removed. For example, EDO Prostate Path, EOD 10—size, EDO 10—nodes, EOD 10—extent and CS\_SSFS (see appendix A) and type of reporting etc... was discarded. The number of features removed in this process was 85 and the total number of features was reduced from 134 to 49 (see appendix A). Moreover records with missing values were discarded. There are 25

features with zero values, this set has been removed, and the number of features becomes 24. some features have few missing values those features were set to unknown, Features contain values or multiple values coded in numeric values for example Marital Status code is from 1 to 7, Marital Status values transformed to corresponding values in real world, code 1 transformed to single value, code 2 transformed to married, 3 to separated, 4 to Divorced, 5 to Widowed, 6 to Unmarried and 7 to Unknown ...etc.

Table 3.1 shows the description of breast cancer dataset used in experiment.

**Table 3.1** Description of seer breast cancer dataset

<b>Dataset Name</b>	<b>Numbers of Features</b>	<b>Numbers of Instance</b>	<b>Numbers of classes</b>	<b>Missing value</b>
Seer Breast Cancer	24	20300	2	No

### 3.3 Weka Tool

Weka is collection of open source of many data mining and machine learning algorithms, it java based, including:

1. Classification.
2. Pre-processing.
3. Clustering.
4. Association rule extraction.

Weka were developed by researchers at the University of Waikato in New Zealand, it is Java based, Weka support ARFF data format as well as CSV. Moreover, weka can also read SQL database. Weka implements all the popular feature selection methods. It contains: tools for preprocessing, classification/regression, clustering, feature/subset evaluators an addition to search algorithms for feature selection and association rules Also it contains 3 graphical user interfaces are:



1. “The Explorer” (exploratory data analysis)
2. “The Experimenter” (experimental environment)
3. “The Knowledge Flow” (new process model inspired interface).

## **3.4 Experiments Methodology**

The experiments were carried out in following stages:

1. The first stage, feature selection was applied on dataset used in the experiments.
2. The second stage is measure the performance of the selected classifiers before applying feature selection methods on the data.
3. The third step the new reduced data passes to selected classifiers again after applying feature selection methods.
4. Finally, the results from the third stage were then compared with corresponding results in second stage in order to evaluate the effectiveness of CFS, RELIEF and Wrapper methods on the selected classifiers.
5. Difference in accuracy is considered significant when the accuracy become greater than results produced by selected classifiers before applying feature selection, tables of accuracies is given summarizing the results of all experiments in section 3.3.2.

## **3.5 Results**

The experiment results are reported in two parts, part one before applying feature selection on dataset and part two after applying feature selection methods.

### **3.5.1 Feature Selection Results**

Table 3.2 shows the sets of features selected by the chosen feature selection methods i.e. CFS method selected five best features; Wrapper returned eight best features and RELIEF returned list of ranked features shows in table 3.3.

**Table 3.2** shows the sets of selected features by CFS, Wrapper and Relief Methods

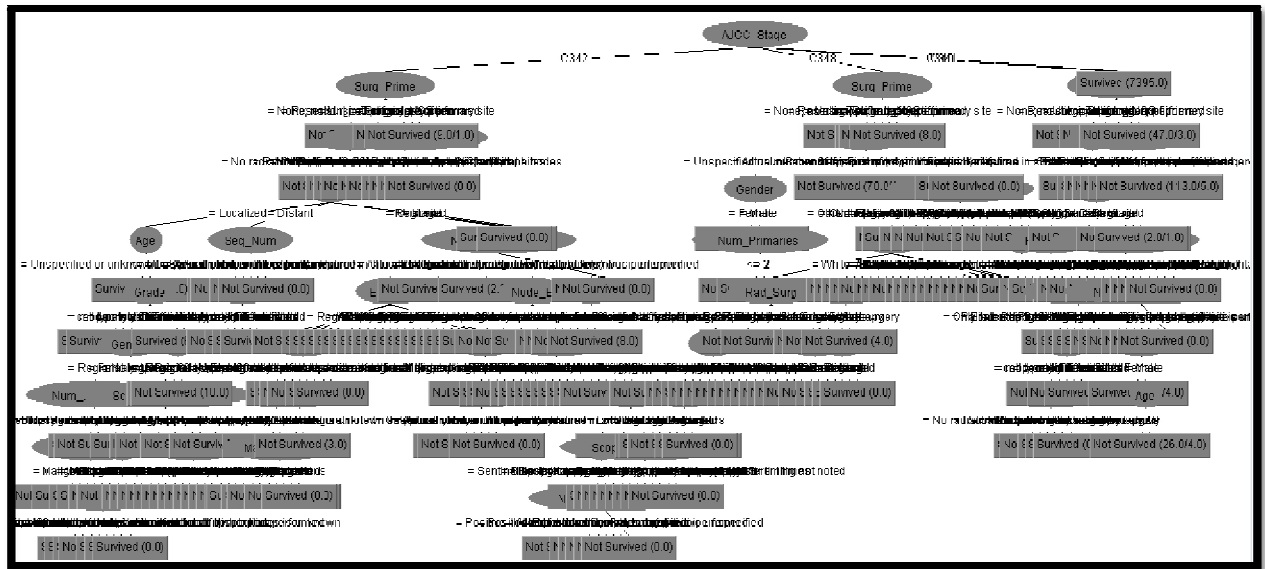
No	All Features	CFS	Wrapper	RELIEF/ J48	RELIEF/ Naïve Bayes	RELIEF/MLP
1	Age	1.		2.	1.	2.
2	Gender	3.		4.	5.	6.
3	Marital Status			7.	8.	9.
4	Ethnicity					
5	Histology		10.	11.	12.	13.
6	Grade		14.	15.	16.	17.
7	Stage		18.	19.	20.	21.
8	AJCC	22.	23.	24.	25.	26.
9	Radiation					27.
10	Behavior					
11	Extension_of_disease					
12	Tumor_size			28.	29.	30.
13	Lympha_Node_Exam					

14	Num_Pos_Nodes		31.			
15	Node_E					32.
16	Num_Primitives					
17	Num_Nodes			33.	34.	35.
18	Scope_LN_Sur			36.	37.	38.
19	Surg_Prime	39.	40.	41.		42.
20	Primary_Site_Code					
21	Reson_Nosurge		43.			44.
22	Rad_Surg		45.			
23	Seq_num	1.		2.	3.	4.
24	<b>Class</b>	<b>5.</b>	<b>6.</b>	<b>7.</b>	<b>8.</b>	<b>9.</b>

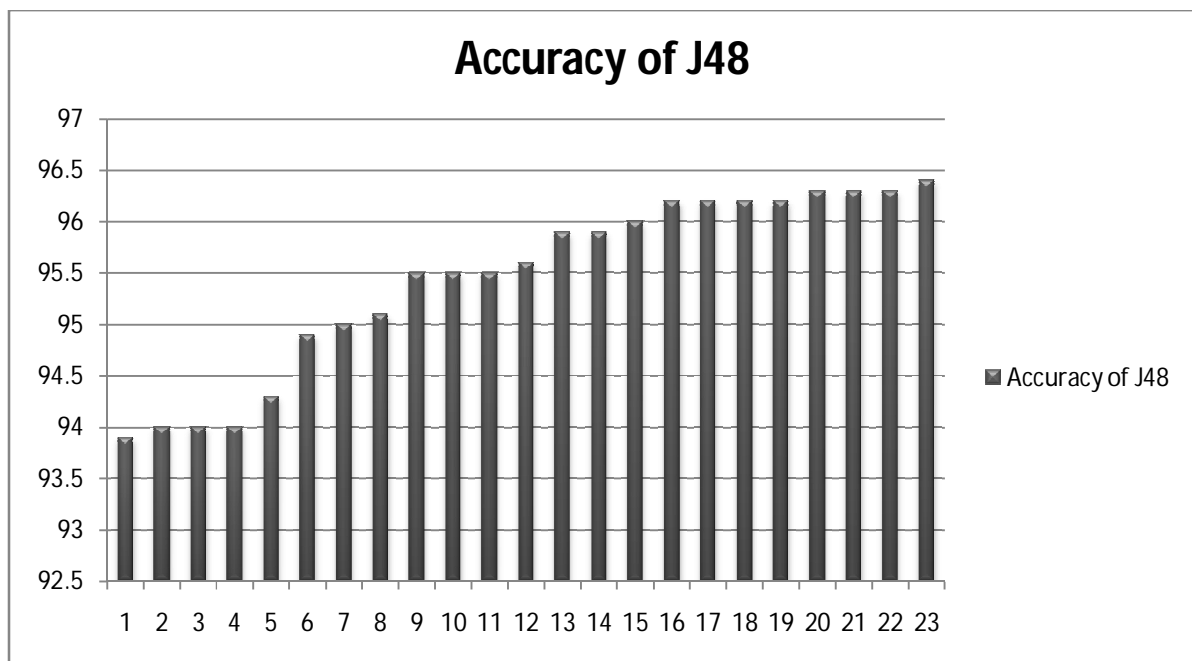
**Table 3.3** shows the sets of selected ranked features by RELIEF Method

<b>NO</b>	<b>Ranked Features</b>	<b>RELIEF/J 48</b>	<b>RELIEF/ Naïve Bayes</b>	<b>RELIEF/ MLP</b>
1	AJCC	1.	2.	1.
2	Num_Nods	2.	3.	4.
3	Seq_Num	5.	6.	7.
4	Marital_Status	8.	9.	10.
5	Radiation	11.	12.	13.

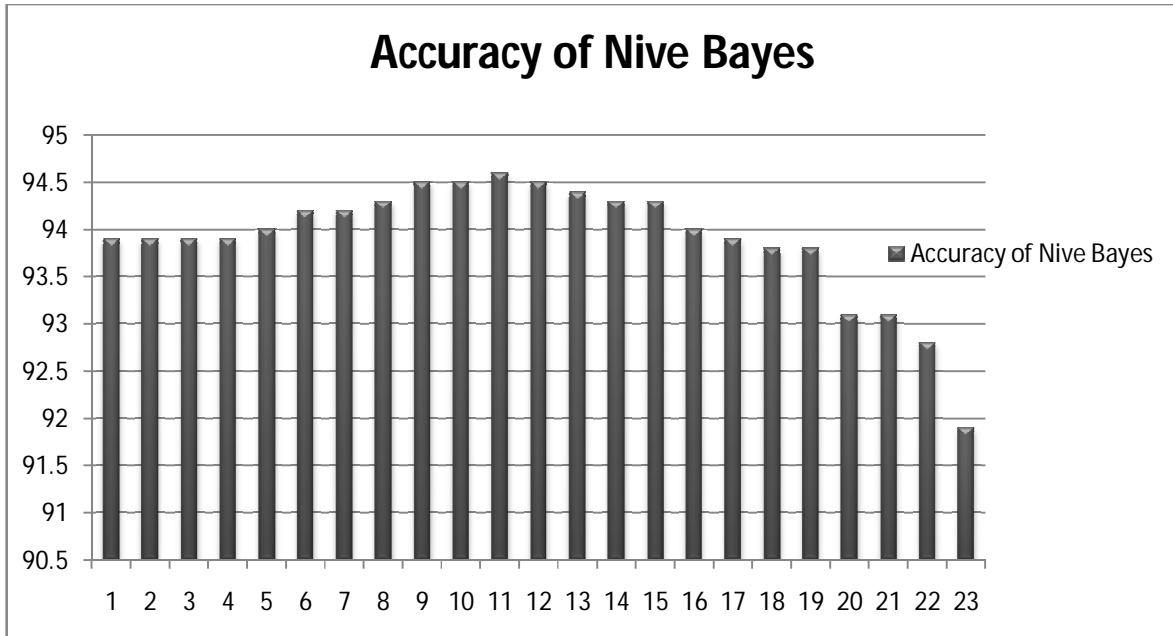
6	Rad_Surg	14.	15.	16.
7	Grade	17.	18.	19.
8	Histology	20.	21.	22.
9	Tumer_Size	23.	24.	25.
10	Lem_node	26.	27.	28.
11	Num_Pos_Node	29.	30.	31.
12	Surg_Prim	32.		33.
13	Ethnicity			34.
14	Gender			35.
15	Num_Primarys			36.
16	Age			
17	Node_E			
18	Scope_Reg_LN_Surg			
19	Stage			
20	behavior			
21	Primary_Site_Code			
22	Reson_Nosurg			
23	Tumer_Extention			
24	<b>Class</b>	<b>1.</b>	<b>2.</b>	<b>3.</b>



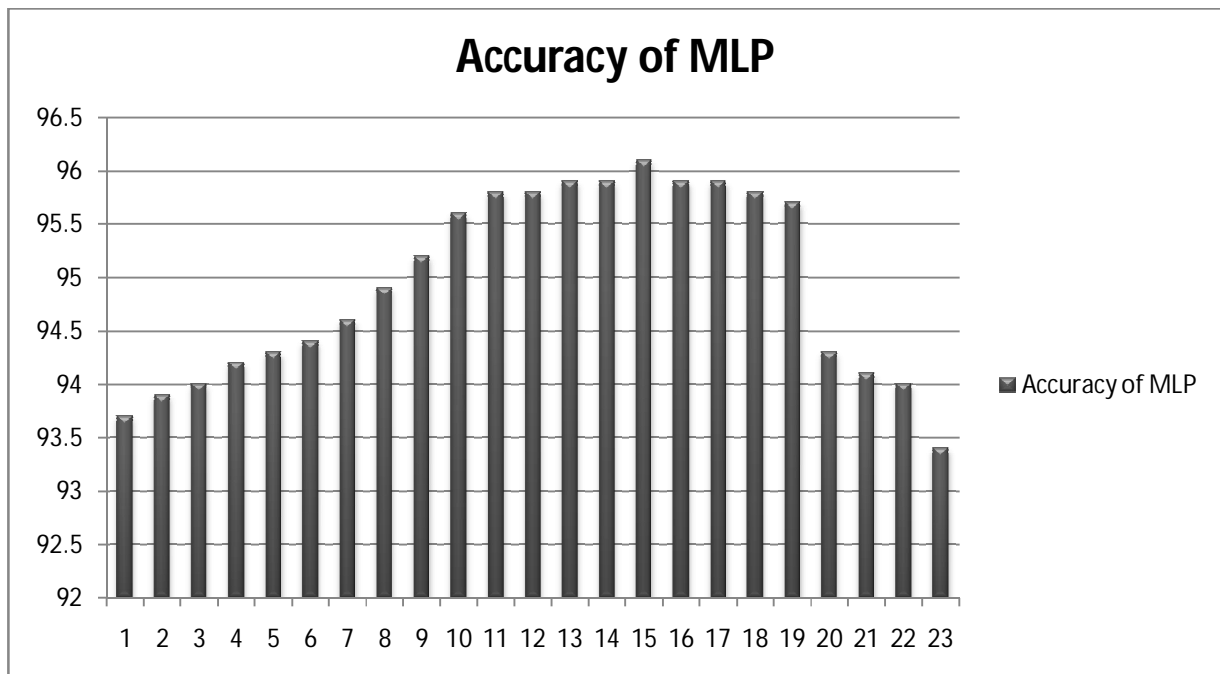
**Figure 3.1** this figure shows selected useful feature by J48 decision tree when using as classifier



**Figure 3.2** this figure shows the result of J48 when using different number of feature ranked RELIEF



**Figure 3.3** this figure shows the result of Naïve Bayes when using different number of feature ranked RELIEF.



**Figure 3.4** this figure shows the result of Multi Layer Perceptron when using different number of feature ranked RELIEF.

## 3.5.2 Classification Results

Table 3.3 shows the accuracy results before applying feature selection using J48, Naïve Bayes and Multi Layer Perceptron classifiers.

**Table 3.4** Accuracy Results before Feature Selection

<b>Classifier- Technique</b>	<b>Full Features</b>	<b>Accuracy</b>	<b>Time(Sec)</b>	<b>Tree Size</b>
J48	24	96.4	1.37	355
Naïve Bayes	24	91.9	0.25	-
Multi Layer Perceptron	24	93.4	20468.34	-

## 3.5.3 J48 Results

This section reports the accuracy of J48 with three feature selection methods CFS, RELIEF and Wrapper.

### 3.5.3.1 CFS

From the table 3.5 it can be seen that the CFS method degrades accuracy of J48, it reduced Accuracy from 96.4 to 95.2 but enhanced J48 by reduce time taken to build the model; it reduced time from 1.37 to 0.21 second. Also enhanced J48 by reduced tree size from 355 to 59, and reduced features number from 24 to 5 features.

### 3.5.3.2 RELIEF

RELIEF method not enhanced accuracy of J48, It leave Accuracy as same as accuracy before feature selection, it is remain 96.4, but enhanced J48 by reduce time taken to build the model; it reduced time from 1.37 to 0.44 second. But degrades J48 by increased tree size from 355 to 358 and reduced features number from 24 to 12 features.

### 3.5.3.3 Wrapper

Wrapper method degrades accuracy of J48, It reduced Accuracy from 96.4 to 90.7 but enhanced J48 by reduce time taken to build the model; it reduced time from 1.37 to 0.09 second. Also enhanced J48 by reduced tree size from 355 to 141, and reduced features number from 24 to 8 features.

**Table 3.5** Accuracy with J48

<b>FS-Method</b>	<b>Reduced Number of Features</b>	<b>Accuracy (%)</b>	<b>Time(Sec)</b>	<b>Tree Size</b>
CFS	5	95.2	0.21	59
RELIEF	12	96.0	0.44	358
WRAPPER	8	90.7	0.9	141

## 3.5.4 Naïve Bayes Results

This section reports the accuracy of Naïve Bayes with three feature selection methods CFS, RELIEF and Wrapper.

### 3.5.4.1 CFS



From the table 3.6 it can be seen that the CFS method enhanced accuracy of Naïve, it increased accuracy from 91.9 to 93.6 also enhanced Naïve Bayes by reduced time taken to build the model; it reduced time from 0.25 to 0.3 seconds, and reduced features number from 24 to 5 features.

### 3.5.4.2 RELIEF

RELIEF method enhanced accuracy of Naïve Bayes on breast cancer dataset, it increased accuracy from 91.9 to 94.6 also enhanced Naïve Bayes by reduced time taken to build the model; it reduced time from 0.25 to 0.5 seconds, and reduced features number from 24 to 11 features.

### 3.5.4.3 Wrapper

Wrapper method enhanced accuracy of Naïve, it increased accuracy from 91.9 to 95.2 also enhanced Naïve Bayes by reduced time taken to build the model; it reduced time from 0.25 to 0.6 seconds, and reduced features number from 24 to 8 features.

**Table 3.6** Accuracy with Naïve Bayes

<b>FS-Method</b>	<b>Reduced Number of Features</b>	<b>Accuracy (%)</b>	<b>Time(Sec)</b>
CFS	5	93.6	0.3
RELIEF	11	94.6	0.5
WRAPPER	8	95.2	0.6

## 3.5.5 Multilayer Perceptron Results

This section reports the accuracy of Multilayer Perceptron with three feature selection methods CFS, RELIEF and Wrapper.

### 3.5.5.1 CFS

From the table 3.7 it can be seen that the CFS method not enhanced and even not degrades accuracy of Multilayer Perceptron, it have same Accuracy with without feature selection method it is 93.9 but enhanced Multilayer Perceptron by reduce time taken to build the model; it reduced time from 20468.34to 168.24 second, and reduced features number from 24 to 5 features.

### 3.5.5.2 RELIEF

RELIEF method enhanced accuracy of Multilayer, it increased accuracy from 93.9 to 96.1 also enhanced Multilayer Perceptron by reduce time taken to build the model; it reduced time from 20468.34to 8252.44 seconds, and reduced features number from 24 to 15 features.

### 3.5.5.3 Wrapper

Wrapper method enhanced accuracy of Multilayer, it increased accuracy from 93.9 to 96.2, also enhanced Multilayer Perceptron by reduced time taken to build the model; it reduced time from 20468.34 to 2216.15 seconds, and reduced features number from 24 to 8 features.

**Table 3.7** Accuracy with Multilayer Perceptron

<b>FS-Method</b>	<b>Reduced Number of Features</b>	<b>Accuracy (%)</b>	<b>Time(Sec)</b>
CFS	5	93.9	168.24
RELIEF	15	96.1	8252.44
WRAPPER	8	96.2	2216.15

## 3.6 Discussion

The experiments are conducted on the above dataset before and after applying feature selection methods. The performance of selected classifiers before applying feature selection is shown in table 3.3 the performance of J48 with feature selection is shown in table 3.4, the performance of Naïve Bayes with feature selection is shown in table 3.5 and the performance of Multilayer Perceptron algorithm with feature selection is shown in table 3.6.

All feature selection methods used in this experiment showed increase in the accuracy for both Naïve Bayes and Multilayer Perceptron. However, J48 the accuracy decreases.

Naïve Bayes is show better results in these experiments. It produced a significant increase in accuracy from 91.9 to 93.6% with CFS method, 93.7 with RELIEF method, and 95.2 with Wrapper method.

## **Chapter 4**

### **4.1 Conclusion**

In this thesis three feature selection methods were applied on a selected dataset and three machine learning methods were applied on the task of classifying the selected dataset before and after applying feature selection. Results showed that Feature Selection greatly enhanced the classifiers accuracy. The performance of classifiers without feature selection in terms of accuracy, time taken to build the model on SEER is reported.

The study shows that Naïve Bayes and Multilayer perceptron have better performance on reduced data set, however, the performance of J48 is worse. This result is expected, as J48 classifier rank the feature by itself. And put in specific order in a decision tree. Therefore reducing features set defer hand way have bad effects in this process. The selection may not be the optimal one is another important result of J48 bad performance.

## **4.2 Future Work**

To take more global view, in future we plan to use other dataset in addition to cancer data. Also more feature selection methods would be consideration as well as more classification techniques such as support vector machine (SVM).

# References

- [1] Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining Concepts and Techniques", Third Edition, 2012, book.
- [2] V. S, S. R, A Dev, "A Comprehensive Study of Artificial Neural Networks," International Journal of Advanced Research in Computer Science and Software Engineering, p. 7, 2012.
- [3] Y. Singh Y, A.S. Chauhan, "Neural Networks in Data Mining", Journal of Theoretical and Applied Information Technology, 2005, pp. 37-42
- [4] J. Han and M. Kamber, "Data Mining; Concepts and Techniques", Morgan Kaufmann Publishers, 2006.
- [5] C. M. Bishop., "Neural Networks and Pattern Recognition," Oxford University Press, 1995
- [6] Sunita Beniwal, Jitender Arora, "Classification and Feature Selection Techniques" in Data Mining, August – 2012.
- [7] [Http://en.wikipedia.org/wiki/C4.5\\_algorithm](http://en.wikipedia.org/wiki/C4.5_algorithm), access date: [November, 2013]. [8] M. Sahami. Learning limited dependence Bayesian classifiers, 1996.
- [8] Ian H. Witten Eibe Frank, "Data Mining Practical Machine Learning Tools" and Third, Second Edition, 2012.
- [9] A. Darwiche, Modeling and Reasoning with Bayesian Networks, Cambridge University Press, 2009

[10] Mark A. Hall "Correlation-based Feature Selection for Machine Learning", 1999 thesis.

[11] "Breast cancer dataset">, Web Site: <http://seer.cancer.gov/> access date: [gugust, 2013].