Real Time Pattern Recognition Algorithm

By

Desai Samir Lalitkumar

08MCE001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AHMEDABAD-382481

May 2010

Real Time Pattern Recognition Algorithm

Major Project

Submitted in partial fulfillment of the requirements

For the degree of

Master of Technology in Computer Science and Engineering

By

Desai Samir Lalitkumar

08MCE001

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AHMEDABAD-382481

May 2010

Declaration

This is to declare that

i) The thesis comprises my original work towards the degree of Master of Technology in Computer science at Nirma University and has not been submitted elsewhere for a degree.

ii) Due acknowledgement has been made in the text to all other material used.

Desai Samir L

Certificate

This is to certify that the Major Project entitled "Real Time Pattern Recognition Algorithm" submitted by Desai Samir Lalitkumar (08MCE001), towards the partial fulfillment of the requirements for the degree of Master of Technology in Computer Science and Engineering of Nirma University of Science and Technology, Ahmedabad is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this major project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Dr.Rajanish Dass
Guide, Professor,
Wing 4E
IIM, Ahmedabad
Vastrapur Ahmedabad - 380015

Dr. S.N. Pradhan
Professor and PG Coordinator,
Department of Computer Engineering,
Institute of Technology,
Nirma University, Ahmedabad

Dr. Ketan Kotecha
Director,
Institute of Technology,
Nirma University, Ahmedabad,

Prof. D. J. Patel
Professor and Head,
Department of Computer Engineering,
Institute of Technology,
Nirma University, Ahmedabad

Abstract

Data mining can be defined as an activity that extracts some new nontrivial information contained in large databases. Real time Pattern recognition is a very active research area which overlaps with various other research fields such as Machine Learning, Artificial Intelligence, Data Mining, Probability Theory, Algebra and Calculus. In recent years the concept of data mining has emerged as one of them.The main focus of the experiment is on the mining algorithms to analyze a much accurate and efficient algorithm.

Data mining is the process of extracting patterns from data. Data mining is becoming an increasingly important tool to transform these data into information. It is commonly used in a wide range of profiling practices, such as marketing, surveillance, fraud detection and scientific discovery.Time series used for prediction of value.Different classifier method has been analyzed. First, in this project we are interested in the comparison of the quality of different mining algorithms.

Acknowledgements

It gives me immense pleasure in expressing my gratitude towards Dr . Rajanish Dass, Professor, IIM-Ahmedabad for his valuable guidance, support and motivation throughout the Major project. I am thankful to him for the valuable time he spent with me for my thesis, for suggestion that shaped the project.

I like to give my special thanks to Dr.S.N Pradhan,Institute of Technology for his suggestions to improve quality of work and providing constant motivation and support. I am also thankful to Dr. K Kotecha, Director, Institute of Technology for his kind support in all respect during my study.

I am thankful to all faculty members of Department of Computer Science and Engineering, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

Last, but not the least, no words are enough to acknowledge constant support and sacrifices of my family members because of whom I am able to complete the degree program successfully.

- Desai Samir L

08MCE001

Contents

List of Tables

List of Figures

Chapter 1

Project Introduction

1.1 Problem Statement

The main goal of the experiment is on the mining algorithms to analyze a much accurate and efficient algorithm.Also observation which best classifier for different all types of problems.

1.2 Objective

Data mining can be defined as an activity that extracts some new nontrivial information contained in large databases. The goal is to discover hidden patterns, unexpected trends or other subtle relationships in the data using a combination of techniques from machine learning, statistics and database technologies. This new discipline today finds application in a wide and diverse range of business, scientific and engineering scenarios. For example, large databases of loan applications are available which record different kinds of personal and financial information about the applicants (along with their repayment histories).The main focus of the experiment is on the mining algorithms to analyze a much accurate and efficient algorithm.

Chapter 2

Temporal data mining

2.1 Introduction

Temporal data mining is concerned with data mining of large sequential data sets. By sequential data, we mean data that is ordered with respect to some index. For example, time series constitute a popular class of sequential data, where records are indexed by time. Other examples of sequential data could be text, gene sequences, protein sequences, lists of moves in a chess game etc. Here, although there is no notion of time as such, the ordering among the records is very important and is central to the data description/modelling. Data mining is the process of extracting patterns from data.[6] Data mining is becoming an increasingly important tool to transform these data into information.[5] It is commonly used in a wide range of profiling practices, such as marketing, surveillance, fraud detection and scientific discovery.

Data mining is the impact of extracting patterns from data. It is becoming a progressively important agency to transform these accumulations into information.This paper analysis on classification algorithm which used in pattern recogniton.Here also datset size increase and getting interested result.For example, Naive-Bayes seems to be a good performer in medical domains.[20] Quinlan identifies families of parallel and sequential domains and claims that neural-networks are likely to perform well

in parallel domains, while decision-tree algorithms are likely to perform well in sequential domains.[22] Therefore, a single induction algorithm cannot build the most accurate classifiers in all situations, some algorithms will perform better in specific domains.[22] It is commonly utilized in a panoramic arrange of profiling practices, much as marketing, surveillance, fraud detection and scientific discovery. Temporal data mining is concerned with mining of large sequential data sets. Sequential data consists of data that is ordered with respect to some index. For example, time series constitutes a popular class of sequential data, where records are indexed by time. Other examples of sequential data could be text, gene sequences, protein sequences, lists of moves in a chess game etc. Here, although there is no notion of time as such, the ordering among the records is very important and is central to the data description/modeling. Data mining is the process of extracting patterns from data.[1] Data mining is becoming an increasingly important tool to transform these data into information. Data mining should be circumscribed as an activity that extracts whatever new nontrivial aggregation is contained in large databases. The goal is to discover hidden patterns, unexpected trends or other subtle relationships in the accumulation using a combination of techniques from machine learning, statistics and database technologies. This new discipline today finds application in a panoramic and different arrange of business, scientific and engineering scenarios. For example, large databases of loan applications are acquirable which record different kinds of individualized and financial aggregation about the applicants. These databases should be mined for exemplary patterns leading to defaults which should determine whether a forthcoming loan application must be accepted or rejected. Several terabytes of remote-sensing image accumulation are gathered from satellites around the globe.[14] Data mining should support reveal potential locations of whatever (as yet undetected) natural resources or assist in antiquity early warning systems for ecological disasters like lubricator slicks etc. Other situations where accumulation defense should be of use include psychotherapy of medical records of hospitals in a town to predict, for example, potential outbreaks of infectious diseases, psychotherapy of customer

transactions for mart investigate applications etc[2].

Time series analysis has quite a long history. Techniques for statistical modelling and spectral analysis of real or complex-valued time series have been in use for more than fifty years .Weather forecasting, financial or stock market prediction and automatic process control have been some of the oldest and most studied applications of such time series analysis.[4] Time series matching and classification have received much attention since the days speech recognition research saw heightened activity. These applications saw the advent of an increased role for machine learning techniques like Hidden Markov Models and time-delay neural networks in time series analysis[2].

2.2 Models and patterns

A model is a global, high-level and often abstract representation for the data. Typically, models are specified by a collection of model parameters which can be estimated from the given data. Often, it is possible to further classify models based on whether they are predictive or descriptive. Predictive models are used in forecast and classification applications while descriptive models are useful for data summarization. For example, auto regression analysis can be used to guess future values of a time series based on its past. Markov models constitute another popular class of predictive models that has been extensively used in sequence classification applications[24]. On the other hand, spectrograms (obtained through time-frequency analysis of time series) and clustering are good examples of descriptive modelling techniques. These are useful for data visualization and help summarize data in a convenient manner. In contrast to the (global) model structure, a pattern is a local structure that makes a specific statement about a few variables or data points. Spikes, for example, are patterns in a real-valued time series that may be of interest.Similarly, in symbolic sequences, regular expressions constitute a useful class of well-defined patterns. In biology, genes, regarded as the classical units of genetic information, are known to appear as local patterns interspersed between chunks of non-coding DNA. Matching

and discovery of such patterns are very useful in many applications. Due to their readily interpretable structure, patterns play a particularly dominant role in data mining[6]. Finally, we note that, while this distinction between models and patterns is useful from the point of view comparing and categorizing data mining algorithms, there are cases when such a distinction becomes blurred.

2.3 Pattern recognition

Pattern recognition is "the act of taking in raw data and taking an action based on the category of the pattern". Most research in pattern recognition is about methods for supervised learning and unsupervised learning. Pattern recognition aims to classify data (patterns) based either on a priori knowledge or on statistical information extracted from the patterns.[1] The patterns to be classified are usually groups of measurements or observations, defining points in an appropriate multidimensional space. This is in contrast to pattern matching, where the pattern is rigidly specified A complete pattern recognition system consists of a sensor that gathers the observations to be classified or described, a feature extraction mechanism that computes numeric or symbolic information from the observations, and a classification or description scheme that does the actual job of classifying or describing observations, relying on the extracted features. The classification or description scheme is usually based on the availability of a set of patterns that have already been classified or described. This set of patterns is termed the training set, and the resulting learning strategy is characterized as supervised learning. Learning can also be unsupervised, in the sense that the system is not given an a priori labeling of patterns, instead it itself establishes the classes based on the statistical regularities of the patterns. The classification or description scheme usually uses one of the following approaches: statistical (or decision theoretic) or syntactic (or structural). Statistical pattern recognition is based on statistical characterizations of patterns, assuming that the patterns are generated by a probabilistic system. Syntactical (or structural) pattern recognition is

based on the structural interrelationships of features.[17] A wide range of algorithms can be applied for pattern recognition, from simple naive Bayes classifiers and neural networks to the powerful KNN decision rules. Pattern recognition is more complex when templates are used to generate variants. For example, in English, sentences often follow the "N-VP" (noun - verb phrase) pattern, but some knowledge of the English language is required to detect the pattern. Pattern recognition is studied in many fields, including psychology, ethology, cognitive science and computer science.

2.4 Temporal data mining tasks

Data mining has been used in a wide range of applications. However, the possible objectives of data mining, which are often called tasks of temporal data mining, these tasks may be grouped as follows: prediction,

a. classification

b. clustering

c. search retrieval

d. pattern discovery

Once again, as was the case with models and patterns, this categorization is neither unique nor exhaustive, the only objective being to facilitate an easy discussion of the numerous techniques in the field.[33]

2.4.1 Classification

Classification means prediction of categorical class labels.Here classifies data based on the training set and the vales in classifying attribute and uses it in classifying new data.Classification is a frequently encountered data mining task in enterprise applications. The goal of classification in data mining is to build a model from a given set

of training data records in which the classes of the objects are known and uses the model to assign classes to new records. For example, a risk classification model can be built from a dataset of previous credit card customers and applied to classify the risk levels of new customers. Several data mining techniques are available for building classification models.Classification is one type of supervised learning where the training data are acompanied by labels indicating the class of the observations.New data is classified based on the training set.

2.4.2 Pattern discovery

In this section we consider the temporal data mining task of pattern discovery. Unlike in search and retrieval applications, in pattern discovery there is no specific query in hand with which to search the database. The objective is simply to unearth all patterns of interest. In that sense, pattern discovery, with its exploratory and unsupervised nature of operation, is something of a sole preserve of data mining. For this reason, this review lays particular emphasis on the profane data mining task of pattern discovery.[12]

In this section, we first introduce the notion of frequent patterns and point out its relevance to rule discovery. Then we discuss, at some length, two popular frameworks for frequent pattern discovery, namely sequential patterns and episodes. In each case we explain the basic algorithm and then state some recent improvements.We end the section by discussing another important pattern class, namely, partially periodic patterns. As mentioned earlier, a pattern is a local structure in the data. It would typically be like a substring or a substring with some 'don't care' characters in it etc. The problem of pattern discovery is to unearth all 'interesting' patterns in the data. There are many ways of defining what constitutes a pattern and we shall discuss some generic methods of defining patterns which one can look for in the data.There is no universal notion for interestingness of a pattern either. However, one concept that is found very useful in data mining is that of frequent patterns.[13] A frequent

pattern is one that occurs many times in the data. Much of data mining literature is afraid with formulating useful pattern structures and developing efficient algorithms for discovering all patterns which occur frequently in the data. Methods for finding frequent patterns are considered important because they can be used for discovering useful rules.[11] These rules can in turn be used to infer some interesting regularities in the data. A rule consists of a pair of Boolean-valued propositions, namely, a left-hand side proposition and a right-hand side proposition. The rule states that when the antecedent is true, then the consequent will be true as well. Rules have been popular representations of knowledge in machine learning and AI for many years. Decision tree classifiers, for example, yield a set of classification rules to reason data. In data mining, association rules are used to capture correlations between different attributes in the data . In such cases, the conditional probability of the consequent occurring given the antecedent, is referred to as certainty of the rule. For example, in a sequential data stream, if the pattern B̈ follows Aäppears f1 times and the pattern C̈ follows B follows Aäppears f2 times, it is possible to infer a profane association rule ẅhenever B follows A, C will follow tooẅith a certainty f2/f1. A rule is usually of interest, only if it has high certainty and it is applicable sufficiently often in the data, i. e., in addition to the certainty f2/f1 being high, frequency of the consequent f2 should also be high.[33]

2.5 Time series

In statistics, signal processing, and some other fields, a time series is a sequence of data points, measured typically at successive times, spaced at instance intervals. Time series analysis comprises methods that attempt to understand such time series, ofttimes either to understand the underlying environment of the data points (Where did they come from? What generated them?), or to make forecasts (predictions). Time series forecasting is the use of a model to forecast future events based on known past events: to forecast future data points before they are measured.[4] A standard

example in econometrics is the opening price of a share of stock based on its past performance. The term instance series analysis is used to characterize a problem, firstly from more ordinary data analysis problems (where there is no natural ordering of the environment of individual observations), and secondly from spatial data analysis where there is a environment that observations (often) relate to geographical locations. There are additional possibilities in the form of space-time models (often called spatial-temporal analysis).Methods for instance series analyses are often divided into two classes: frequency-domain methods and time-domain methods. A instance series model will generally reflect the fact that observations close together in instance will be more closely related than observations further apart.

2.5.1 Notation

A number of different notations are in use for time-series analysis:

$X = \{X1, X2, ...\}$is a common notation which specifies a time series X which is indexed by the natural numbers.

$$Y_t = \alpha_0 Y_{t-1} + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + ... + \alpha_p Y_{t-p} + \epsilon_t \tag{2.1}$$

To denote this, we will index the observations with the letter t rather than the letter i. Our data will be observations on Y1, Y2, ...Yt, ...where t indexes the day, month, year, or any time interval.Time series is about dependence. We use correlation as a measure of dependence. Although we have only one variable, we can compute the correlation between Y_t and Y_{t-1} or between Y_t and Y_{t-2}.The correlations between Ys at different times are called autocorrelations.[10]

2.5.2 General exploration

Graphical examination of data series. Autocorrelation analysis to examine serial dependence. Spectral analysis to examine cyclic behaviour which need not be related

to seasonality

2.5.3 Prediction and forecasting

Fully-formed statistical models for stochastic simulation purposes, so as to generate alternative versions of the time series, representing what might happen over non-specific time-periods in the future (prediction). Simple or fully-formed statistical models to describe the likely outcome of the time series in the immediate future, given knowledge of the most recent outcomes (forecasting).

2.6 Cross Validation

The goal of using cross validation is to know the good parameters so that the classifier can accurately predict unknown data. A common way is to separate training data into two parts, one of which is considered unknown in training the classifier. Then the prediction accuracy on this set can more precisely reflect the performance on classifying unknown data. An improved version of this procedure is cross-validation. In v-fold cross-validation, the training set is divided into v subsets of equal size. Sequentially one subset is tested using the classifier trained on the remaining v - 1 subsets. Thus, each instance of the whole training set is predicted once so the cross-validation accuracy is the percentage of data which are correctly classified. The cross-validation procedure can prevent the over fitting problem.

2.7 Comparing Algorithms

In medical domains, many measurements (attributes) that doctors have developed over the years tend to be independent: if the attributes are highly correlated, only one attribute will be chosen. In such domains, a certain class of learning algorithms might outperform others. For example, Naive-Bayes seems to be a good performer in medical domains.[20] Quinlan identifies families of parallel and sequential domains and claims

that neural-networks are likely to perform well in parallel domains, while decision-tree algorithms are likely to perform well in sequential domains.[22] Therefore, although a single induction algorithm cannot build the most accurate classifiers in all situations, some algorithms will perform better in specific domains.[22]

2.8 Drawbacks of Existing Algorithms

This field is still in it's infancy and is constantly evolving. The first people who gave a serious thought to the problem of data mining were those researching in the Database field , since they were the first to face this problem. Whereas most of the tools and techniques used for data mining come from other related fields like pattern recognition, statistics and complexity theory.[22] It is only recently that the researchers of these various fields have been interacting to solve the mining issue.

Chapter 3

Dataset

3.1 Data Set Format

A data set consists of a number of observations/patterns. An observation is represented by a 1xd dimensional vector, where d is the dimensionality of the feature space. Each observation has a corresponding class label, the class labels have values from 0,...,C-1 where C is the total number of classes.

Each row in a data set represents a single observation with the class label appended at the end. The variables and class label in an observation are separated by a comma. The following is an example of a data set.

1, 10, 100, 1000, 0

4, 20, 80, 5, 1

100, 50,10,40,2

The data set contains 3 observations, the dimensionality of the feature space is 4 and each observation belongs to a different class. Dataset taken from UCI Machine Learning Repository. Data from a breast cancer study is used in this study. The data consists of 22 cDNA microarrays, each representing 3226 genes based on biopsy specimens of primary breast tumors of 7 patients with germ-line mutations of BRCA1, 8 patients with germ-line mutations of BRCA2, and 7 with sporadic cases[18].

Chapter 4

Algorithm

4.1 SVM Model Classifier

A Support Vector Machine (SVM) performs classification by constructing an N-dimensional hyperplane that optimally separates the data into two categories. SVM models are closely related to neural networks.Using a kernel function, SVM's are an alternative training method for polynomial, radial basis function and multi-layer perceptron classifiers in which the weights of the network are found by solving a quadratic programming problem with linear constraints, rather than by solving a non-convex, unconstrained minimization problem as in standard neural network training.[31] In the parlance of SVM, a predictor variable is called an attribute, and a transformed attribute that is used to define the hyperplane is called a feature. The task of choosing the most suitable representation is known as feature selection. A set of features that describes one case (i.e., a row of predictor values) is called a vector. So the goal of SVM modeling is to find the optimal hyperplane that separates clusters of vector in such a way that cases with one category of the target variable are on one side of the plane and cases with the other category are on the other size of the plane.[7] The vectors near the hyperplane are the support vectors.[21] Implementation of a C-Support Vector classification algorithm with a radial basis function kernel.[8] The

optimal penalty factor(C) and basis function width(g) are determined by performing a grid search with the Golden Ratio Search algorithm. Some common kernels include,[27]

Linear:

$$k(x_i, x_j) = x_i^T x_j \tag{4.1}$$

Ploynomial:

$$k(x_i, x_j) = (x_i, x_j)^d \tag{4.2}$$

Radial Basis:

$$k(x_i, x_j) = exp(-\left\|x_i - x_j\right\|^2 / 2\sigma^2) \tag{4.3}$$

Sigmoid:

$$k(x_i, x_j) = \tanh(k(x_i.x_j + \Theta)) \tag{4.4}$$

4.2 k-nearest neighbor algorithm

In pattern recognition, the k-nearest neighbor algorithm (k-NN) is a method for classifying objects based on closest training examples in the feature space. k-NN is a type of instance-based learning, or lazy-learning where the function is only approximated locally and all computation is deferred until classification. The training examples are mapped into multidimensional feature space. The space is partitioned into regions by class labels of the training samples. A point in the space is assigned to the class c if it is the most frequent class label among the k nearest training samples. Usually Euclidean distance is used. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. In the actual classification phase, the same features as before are computed for the test sample (whose class is not known). Distances from the new vector to all stored vectors are computed and k closest samples are selected. The new point is predicted to belong to the most numerous class within the set. The best choice of k depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but

make boundaries between classes less distinct. A good k can be selected by parameter optimization using, for example, cross-validation. The special case where the class is predicted to be the class of the closest training sample (i.e. when k = 1) is called the nearest neighbour algorithm. The accuracy of the k-NN algorithm can be severely degraded by the presence of noisy or irrelevant features, or if the features scales are not consistent with their relevance. Van der Walt and Barnard has also shown that the optimal value of k is influenced by the amount of output noise in the data.[9] They had also show that the archilles heel of the kNN classifier is the constant distance metric that it uses.Much research effort has been placed into selecting or scaling features to improve classification. A particularly popular approach is the use of evolutionary algorithms to optimize feature scaling.The algorithm is easy to implement, but it is computationally intensive, especially when the size of the training set grows. Many optimizations have been proposed over the years; these generally seek to reduce the number of distances actually computed. Some optimizations involve partitioning the feature space, and only computing distances within specific nearby volumes.

$Posterior = prior \times likelihood/setminusevidence$ Implementation of a kNN classifier. The kNN classifier classifies a sample by determining the k nearest data points to the given point in Euclidian space. N Euclidian distances are thus calculated for each sample that has to be classified, where N are the total number of samples in the training set. After the k nearest points (neighbours) have been determined, their corresponding class labels are used to make a classification prediction for the new observation.

Values of k between 1 and 10 are used with 10-fold cross-validation to determine the optimal k value.

4.3 Gaussian Classifier

Gauss - Gaussian Classifier A Gaussian classifier implementation. Each class is assumed to have a Gaussian distribution. Maximum likelihood estimation is performed

to determine the means and covariances of each class. The prior probabilities of each class are determined by the number of occurrences of each class in the training set. Bayes' rule is used to determine the posterior probabilities of the test observations belonging to each class. A classification decision is made for each observation in the test set by selecting the class with the highest posterior probability for each observation.

4.4 GMM - Gaussian Mixture Model Classifier

A simple implementation of a GMM classifier. The number of mixtures per class is given to the classifier. For each class the Expectation Maximization (EM) algorithm is used to determine the mixture/group means and covariances. Equal group prior probabilities are assumed. The class-conditional probability density function of each class is determined by substituting the mean vector and covariance matrix of each mixture into the multivariate gaussian distribution equation and the adding all these probability values. The class prior probabilities are determined by the proportion of samples belonging to a specific class in the training set. To classify a new observation, the class posterior probabilities are calculated by using the Bayes' rule.

The number of mixtures per class are iterated from 1 to 10 to determine the optimal number of mixtures per class.

It should be noted that the Gaussian Mixture Classifier makes use of diagonal covariance matrices for the mixtures.

4.5 Neural Network

In general a natural neuronal meshwork is composed of a group or groups of chemically connected or functionally related neurons. A single neuron may be connected to many another neurons and the total number of neurons and connections in a meshwork may be extensive. Connections, called synapses, are usually formed from axons to dendrites, though dendrodendritic microcircuits and other connections are possi-

ble. Apart from the electrical signaling, there are another forms of signaling that arise from neurotransmitter diffusion, which have an effect on electrical signaling. As such, neural networks are extremely complex. Artificial intelligence and cognitive modeling try to simulate some properties of neural networks. While similar in their techniques, the past has the aim of solving particular tasks, patch the latter aims to build mathematical models of natural neuronal systems. In the artificial intelligence field, artificial neuronal networks have been applied successfully to speech recognition, image analysis and adaptive control, in order to construct software agents video gamesartificial neuronal networks for artificial intelligence are based on statistical estimation, optimization and control theory. The cognitive modelling field involves the physical or mathematical modeling of the behaviour of neuronal systems; ranging from the individual neuronal level of neurons to a stimulusmodelling the release and effects of dopamine in the essential gangliato the complete organism organism's salutation to stimuliparallel with input variables and consequently handle large sets of data swiftly. The principal strength with the meshwork is its ability to find patterns and irregularities as well as detecting multi-dimensional non-linear connections in data. The latter quality is extremely useful for modelling dynamical systems, e.g. the stock market. Apart from that, neuronal networks are frequently used for pattern acceptance tasks and non-linear regression[2].An Error Correction Neural Network is built and implemented for an empirical study. Standard benchmarks are used to evaluate the networks ability to make forecasts.The principal motivation for the neural network approach in stock prevision is twofold: 1.stock data is highly complex and hornlike to model, therefore a non-linear model is beneficial. 2. a large set of interacting input series is often required to explain a specific stock, which suites neuronal networks It is also possible to approach the prevision task from the angle of economics.Thus a neuronal meshwork is a complete statement of the financial market in itself[6].

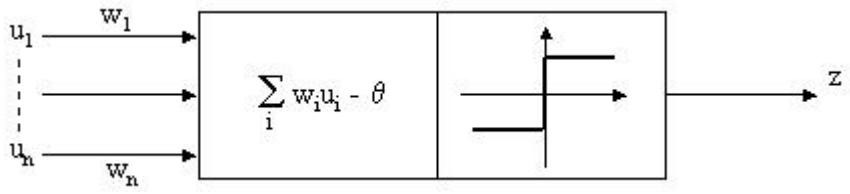

Figure 4.1: The artificial neuron with a threshold function.

4.5.1 The brain, neural networks and computers

Neural networks, as used in artificial intelligence, have traditionally been viewed as simplified models of neural processing in the brain, even though the relation between this model and brain biological architecture is debated[citation needed]. A subject of current research in theoretical neuroscience is the question surrounding the degree of complexity and the properties that individual neural elements should have to reproduce something resembling animal intelligence. Historically, computers evolved from the von Neumann architecture, which is based on sequential processing and execution of explicit instructions. On the other hand, the origins of neural networks are based on efforts to model information processing in biological systems, which may rely largely on parallel processing as well as implicit instructions based on recognition of patterns of 'sensory' input from external sources. In other words, at its very heart a neural network is a complex statistical processor (as opposed to being tasked to sequentially process and execute).

4.5.2 Neural networks and artificial intelligence

An artificial neural network (ANN), also called a simulated neural network (SNN) or commonly just neural network (NN) is an interconnected group of artificial neurons that uses a mathematical or computational model for information processing based on a connectionistic approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that flows through the network. In more practical terms neural networks are non-linear statis-

tical data modeling or decision making tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data.[19]

4.5.3 Applications

The utility of artificial neural network models lies in the fact that they can be used to infer a function from observations and also to use it. This is particularly useful in applications where the complexity of the data or task makes the design of such a function by hand impractical. The tasks to which artificial neural networks are applied tend to fall within the following broad categories: Function approximation, or regression analysis, including time series prediction and modelling. Classification, including pattern and sequence recognition, novelty detection and sequential decision making. Data processing, including filtering, clustering, blind signal separation and compression.[11]

4.5.4 Learning paradigms

There are three major learning paradigms, each corresponding to a particular abstract learning task. These are supervised learning, unsupervised learning and reinforcement learning. Usually any given type of network architecture can be employed in any of those tasks.

4.5.5 Supervised learning

In supervised learning, we are given a set of example pairs and the aim is to find a function f in the allowed class of functions that matches the examples. In other words, we wish to infer how the mapping implied by the data and the cost function is related to the mismatch between our mapping and the data.[26]

4.5.6 Unsupervised learning

In unsupervised learning we are given some data x, and a cost function which is to be minimized which can be any function of x and the network's output, f. The cost function is determined by the task formulation. Most applications fall within the domain of estimation problems such as statistical modeling, compression, filtering, blind source separation and clustering.

4.5.7 Neural networks and neuroscience

Theoretical and computational neuroscience is the field concerned with the theoretical analysis and computational modeling of biological neural systems. Since neural systems are intimately related to cognitive processes and behaviour, the field is closely related to cognitive and behavioural modeling. The aim of the field is to create models of biological neural systems in order to understand how biological systems work. To gain this understanding, neuroscientists strive to make a link between observed biological processes (data), biologically plausible mechanisms for neural processing and learning (biological neural network models) and theory (statistical learning theory and information theory).

4.5.8 Types of models

Many models are used in the field, each defined at a different level of abstraction and trying to model different aspects of neural systems. They range from models of the short-term behaviour of individual neurons, through models of how the dynamics of neural circuitry arise from interactions between individual neurons, to models of how behaviour can arise from abstract neural modules that represent complete subsystems. These include models of the long-term and short-term plasticity of neural systems and its relation to learning and memory, from the individual neuron to the system level.[29]

4.6 MLP - Multilayer Perceptron

Implementation of a feed-forward, back propagation neural network. The MLP is a specific case of Neural Networks. Neural Networks refer to Radial Basis Function Networks as well as Multilayer Perceptrons. A single hidden layer is used in this implementation. The number of nodes in the hidden layer is iterated from 2 to 10 and 10-fold cross-validation is performed to determine the optimal number of hidden nodes.

4.7 Genetic Algorithm

A genetic algorithm is a search technique used in computing to find exact or approximate solutions to improvement and search problems. Genetic algorithms are categorized as global search heuristics. Genetic algorithms are a particular collection of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and crossover.[3]
subsectionMethodology Genetic algorithms are implemented in a computer simulation in which a accumulation of abstract representations (called chromosomes or the genotype of the genome) of candidate solutions (called individuals, creatures, or phenotypes) to an improvement problem evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but another encodings are also possible. The evolution usually starts from a accumulation of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the accumulation is evaluated, multiple individuals are stochastically designated from the current accumulation (based on their fitness), and modified (recombined and possibly arbitrarily mutated) to modify a new population. The new accumulation is then used in the next process of the algorithm. Commonly, the algorithm terminates when either a maximum sort of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm

has terminated cod to a maximum sort of generations, a satisfactory resolution haw or haw not have been reached. Genetic algorithms find application in bioinformatics, phylogenetics, computational science, engineering, economics, chemistry, manufacturing, mathematics, physics and another fields. A typical genetic algorithm requires: a genetic state of the solution domain, a fitness duty to evaluate the resolution domain.[25] A standard state of the resolution is as an array of bits. Arrays of another types and structures should be used in essentially the same way. The main property that makes these genetic representations convenient is that their parts are easily aligned cod to their fixed size, which facilitates simple crossover operations. Variable length representations haw also be used, but crossover feat is more complex in this case.[30] Tree-like representations are explored in genetic planning and graph-form representations are explored in evolutionary programming. The fitness duty is defined over the genetic state and measures the quality of the represented solution. The fitness duty is always problem dependent.[15] For instance, in the knapsack problem one wants to tap the total value of objects that should be put in a knapsack of whatever fixed capacity. A state of a resolution might be an array of bits, where each bit represents a different object, and the value of the bit (0 or 1) represents whether or not the object is in the knapsack. Not every such state is valid, as the size of objects haw exceed the capacity of the knapsack. The fitness of the solution is the sum of values of every objects in the knapsack if the representation is valid, or 0 otherwise.Once we have the genetic representation and the fitness duty defined, GA proceeds to initialize a accumulation of solutions randomly, then improve it through repetitive application of mutation, crossover, inversion and selection operators.[34] In whatever problems, it is hard or even impossible to define the fitness expression; in these cases, interactive genetic algorithms are used.[28]

4.7.1 Initialization

Initially many individual solutions are randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions.

4.7.2 Selection

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as this process may be very time-consuming. Most functions are stochastic and designed so that a small proportion of less fit solutions are selected. This helps keep the diversity of the population large, preventing premature convergence on poor solutions. Popular and well-studied selection methods include roulette wheel selection and tournament selection.[23]

4.7.3 Reproduction

The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation. For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each new child, and the process continues until a new population of solutions of appropriate size is generated. Although reproduction methods that are based on the use of two parents are more "biology inspired", recent researches suggested more

than two "parents" are better to be used to reproduce a good quality chromosome. These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above.[30]

4.7.4 Termination

This generational process is repeated until a termination condition has been reached. Common terminating conditions are: A solution is found that satisfies minimum criteria. Fixed number of generations reached. Allocated budget (computation time/money) reached .The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results. Manual inspection. Combinations of the above Simple generational genetic algorithm pseudocodes. Choose the initial population of individuals Evaluate the fitness of each individual in that population Repeat on this generation until termination: (time limit, sufficient fitness achieved, etc.) Select the best-fit individuals for reproduction Breed new individuals through crossover and mutation operations to give birth to offspring Evaluate the individual fitness of new individuals Replace least-fit population with new individuals.[34]

4.8 DT - Decision Tree Classifier

A Bayes Decision Tree classifier. Decision trees capture dependencies between variables without assuming total independence. The dependencies are however also limited to prevent too complex models. The probability distribution of each class is modelled as a tree dependent distribution. The probability distribution is expressed as the product of pair wise conditional probability densities between variables. After the class-conditional probability density functions of each class are determined, clas-

sification can be performed by substituting a new observation into each probability density selecting the most probable class.

4.9 Naive Bayes classifier

A simple Naive Bayes classifier implementation. Each class is assumed to have a Gaussian distribution with independent variables. Maximum likelihood estimation is performed to determine the means and covariances of each class. The prior probabilities of each class are determined by the number of occurrences of each class in the training set. Bayes' rule is used to determine the posterior probabilities of the test observations belonging to each class. A classification decision is made for each observation in the test set by selecting the class with the highest posterior probability for each observation.

Chapter 5

Algorithm comparison I

5.1 Experimental Evaluation

Table includes datasets which are used in experiment to compare these algorithms in terms of Accuracy and generation time. Data sets are taken from the UCI Machine learning Repository , which have been discretised/ normalized.

Table I: Dataset

S.No.	Dataset	size
1	Iris	3KB
2	Heart dieases	10KB
3	German credit score	53KB
4	Australian	29KB
5	Diabetes	24KB

All the experiments are performed on my laptop using Matlab platform Intel celron M Processor 1.30GHZ,760MB RAM . Comparison is based on Accuracy and generation time. Analysis is shown below.

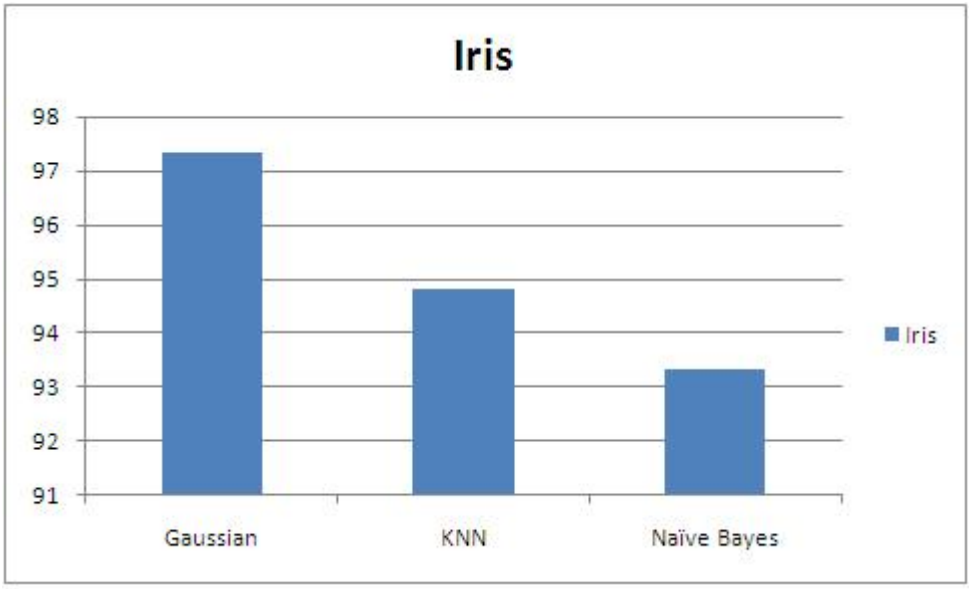

Figure 5.1: Iris dataset accuracy comparision

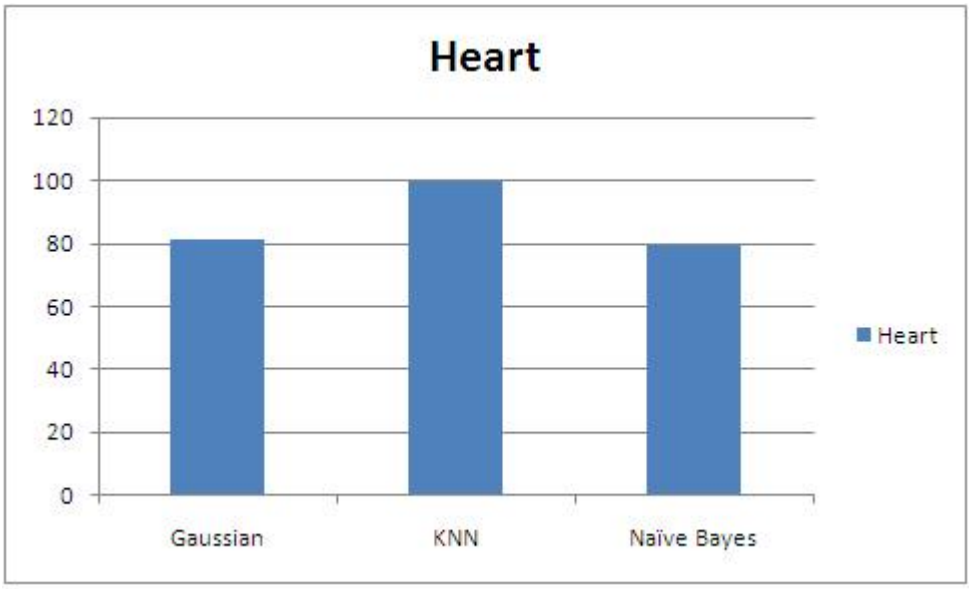

Figure 5.2: Heartdieases dataset accuracy comparision

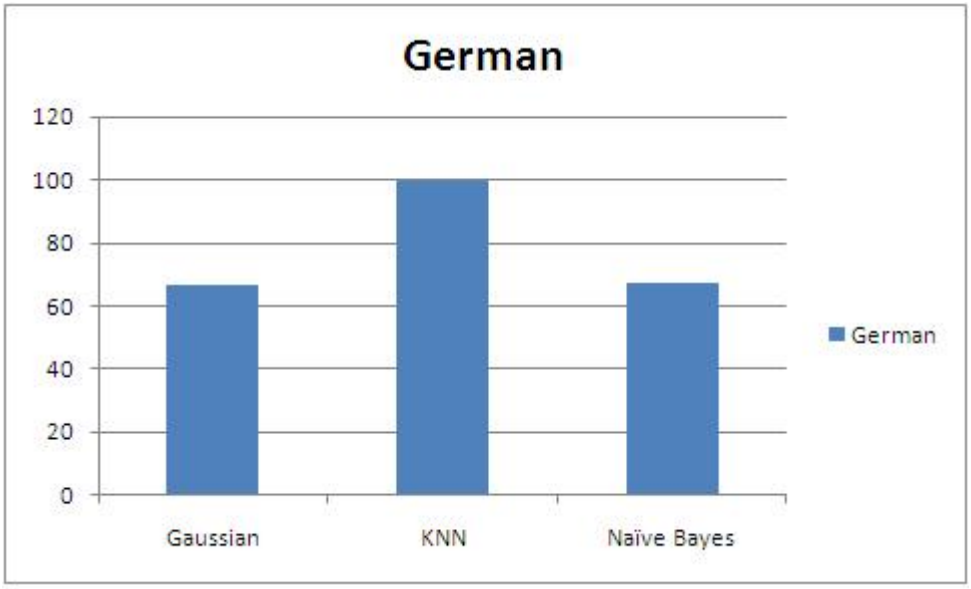

Figure 5.3: Germancredit dataset accuracy comparision

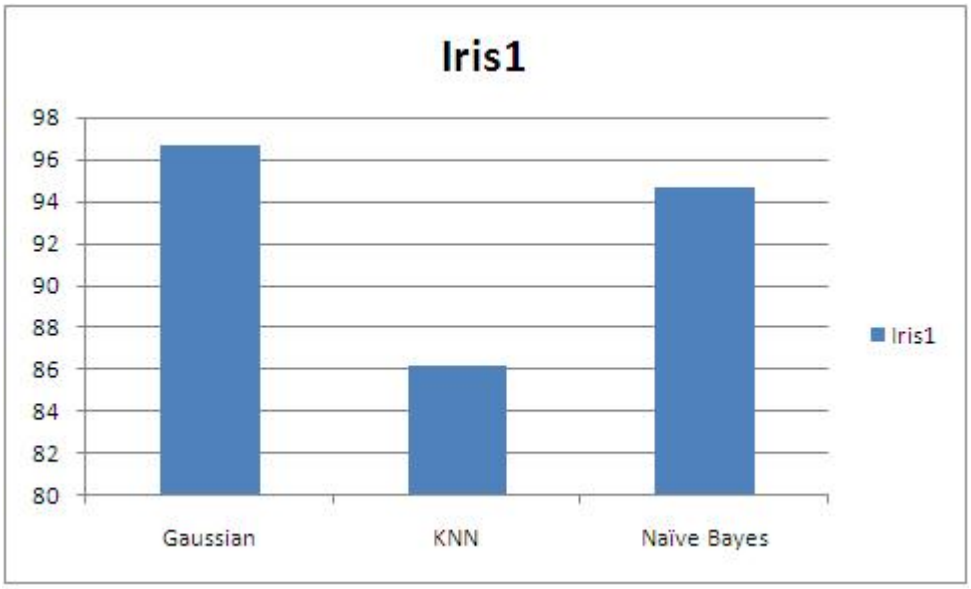

Figure 5.4: Iris1 dataset accuracy comparision

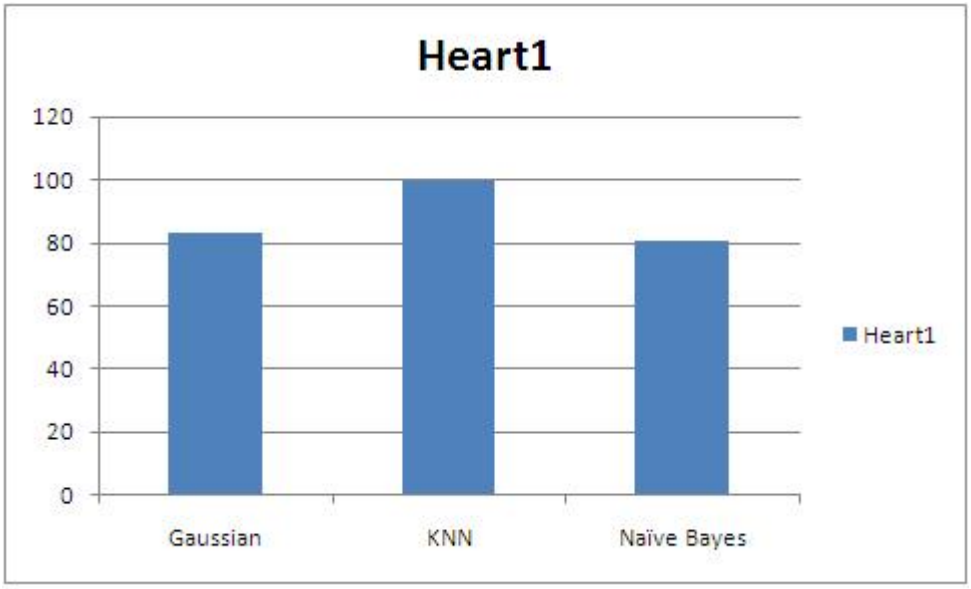

Figure 5.5: Heartdieases1 dataset accuracy comparision

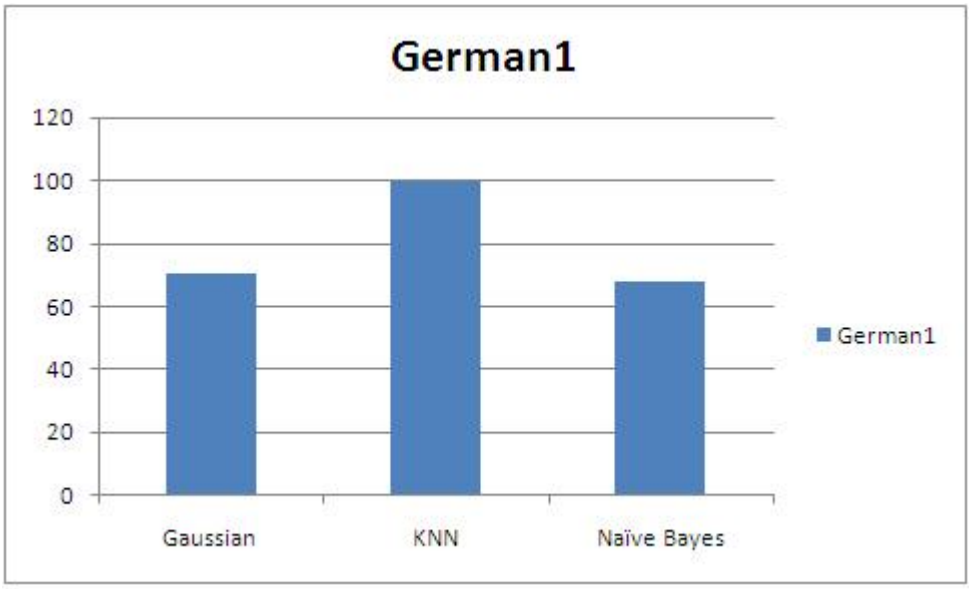

Figure 5.6: Germancredit1 dataset accuracy comparision

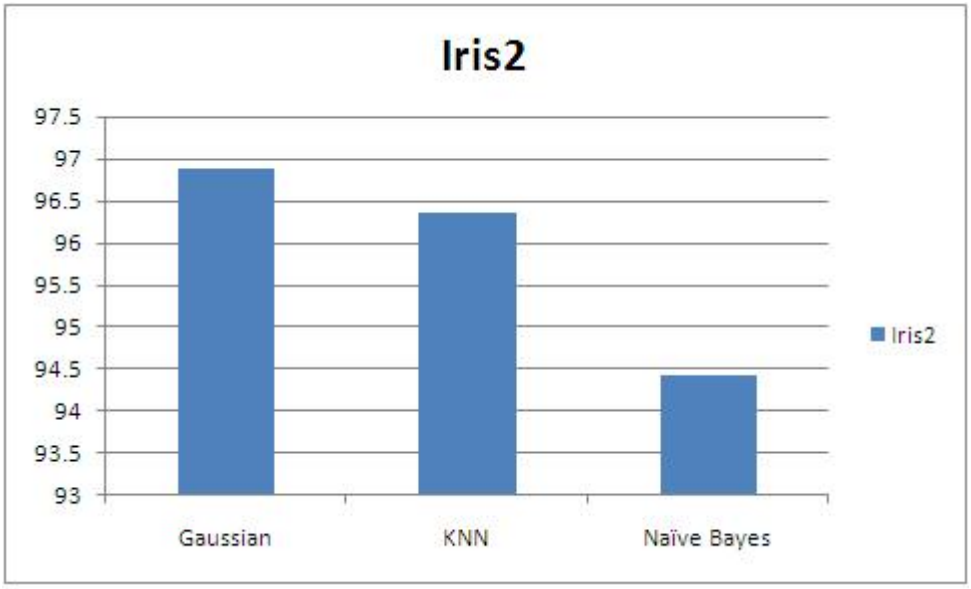

Figure 5.7: Iris2 dataset accuracy comparision

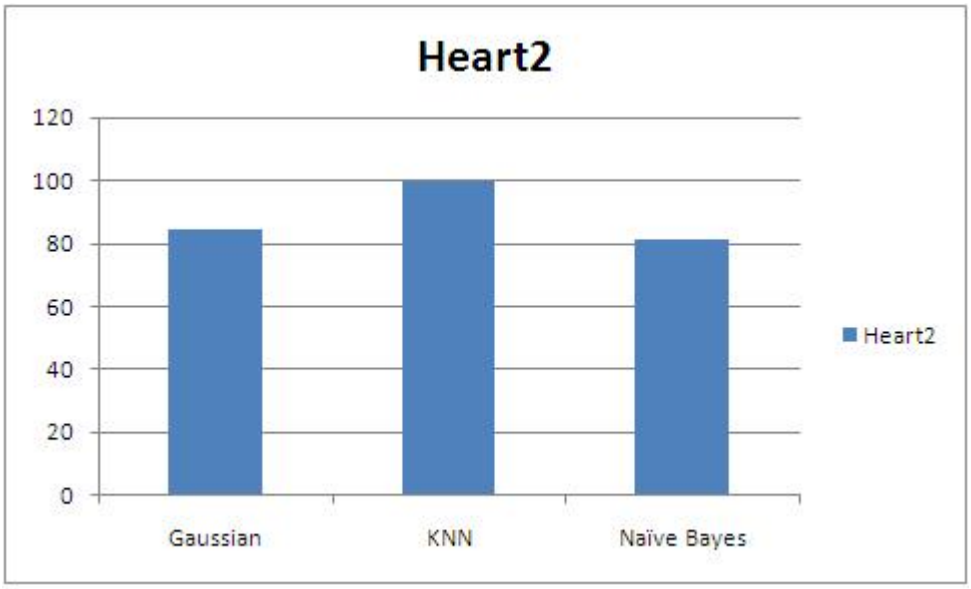

Figure 5.8: Heartdieases2 dataset accuracy comparision

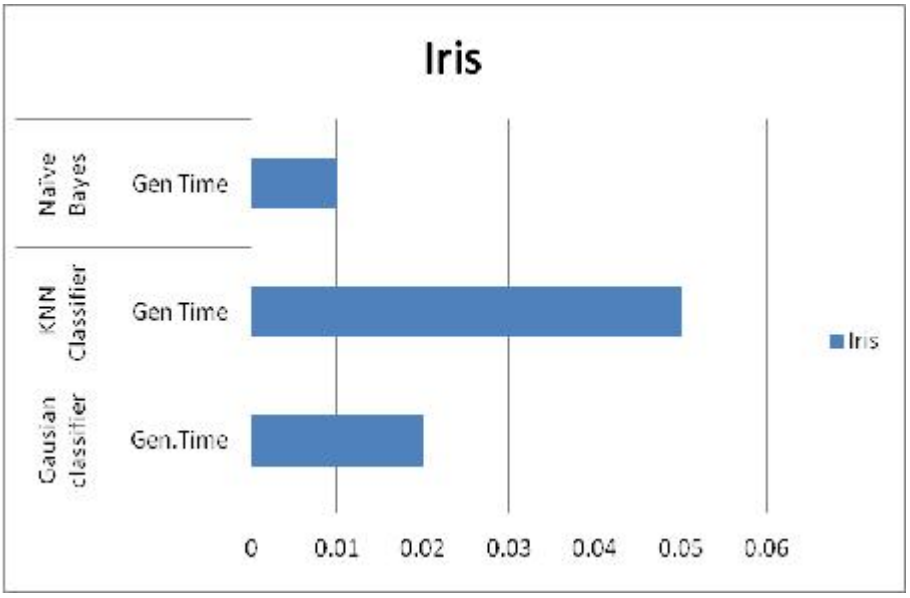

Figure 5.9: Iris dataset Time comparision

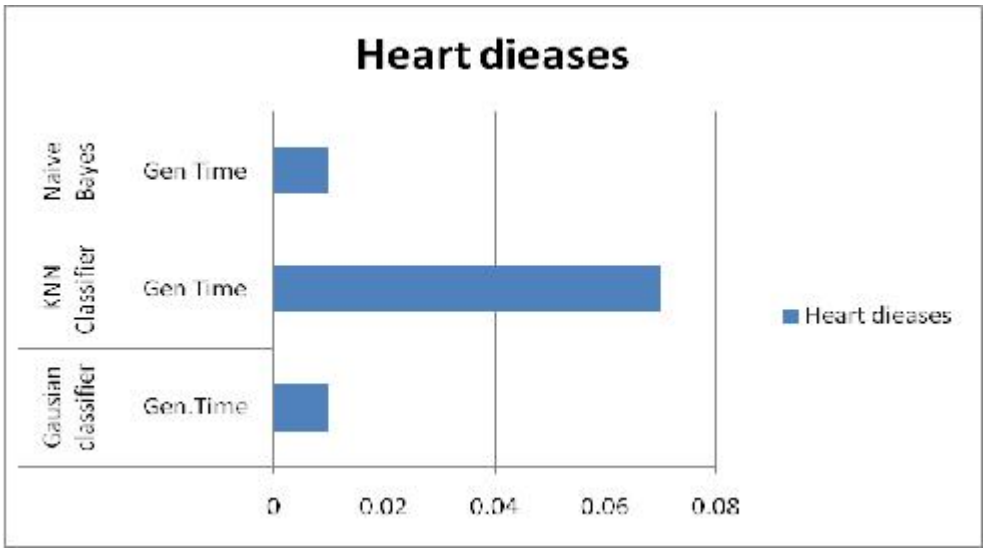

Figure 5.10: Heartdeases dataset Time comparision

Table II: Accuracy percentage

S.No	*Dataset*	*Gaussian*	*KNN*	Naive bayes
1	Iris	97.33	94.8	93.33
2	Heart	81.11	100	79.62
3	German credit score	66.66	100	67.60
4	Iris1(double)	96.66	86.2	94.66
5	Heart1(double)	83.33	100	80.74
6	German credit score1(double)	70.70	100	67.80
7	Iris2(triple)	96.88	96.36	94.44
8	Heart2(triple)	84.19	100	81.33

Table III: Time

S.No	*Dataset*	*Gaussian*	*KNN*	Naive bayes
1	Iris	0.02 sec	0.05 sec	0.01 sec
2	Heart	0.01 sec	0.07 sec	0.01sec
3	German credit score	0.01sec	67 sec	0.01sec
4	Iris1	0.05 sec	60 sec	0.03 sec
5	Heart1	0.02 sec	61 sec	0.03sec
6	German credit score1	0.02sec	564 sec	0.04sec
7	Iris2	0.09 sec	94 sec	0.08 sec
8	Heart2	0.05 sec	97 sec	0.09sec

5.1.1 Observation(size single)

The experiment has observed for Iris dataset. The accuracy performance good for Gausian method when dataset size single. For Heart disease and German credit scoring dataset good method is KNN clasifier. The experiment has also observed for time wise Naive bayes method is good for all three dataset.

5.1.2 Observation (size double)

The experiment has observed for Iris dataset. The accuracy performance good for Gausian method when dataset size double. For Heart disease and German credit scoring dataset good method is KNN clasifier. The experiment has also observed for time wise Gaussian method is good for all three dataset.

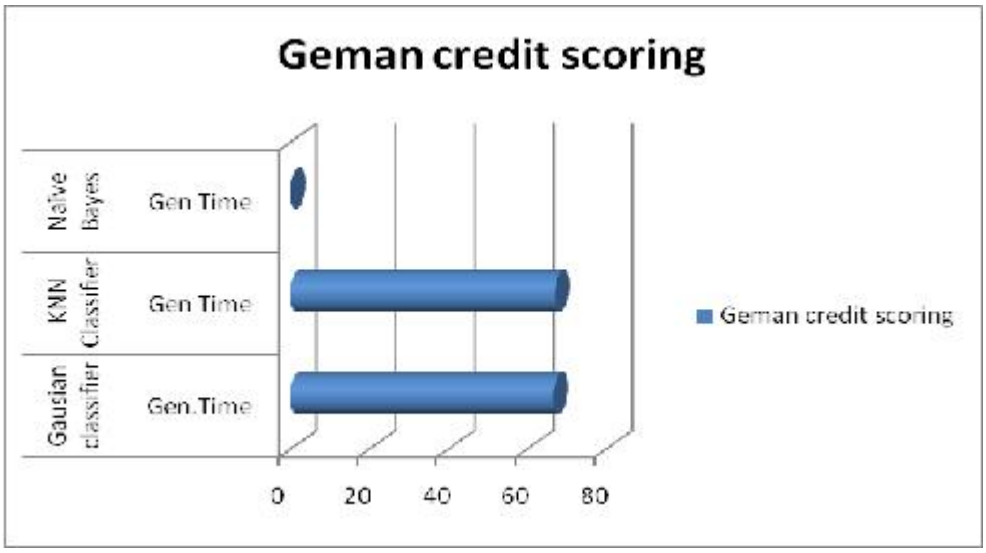

Figure 5.11: German dataset Time comparision

5.1.3 Observation (size triple)

The experiment has observed for Iris dataset. The accuracy performance good for Gausian method when dataset size triple. For Heart disease and German credit scoring dataset good method is KNN clasifier. The experiment has also observed for time wise Naive bayes method is good for all three dataset.KNN classifier is good for heart dataset.

It has also compared with respect to time.Naive bayes is good for iris dataset.Gaussian method is good for heart dataset.

Table IV: Error comparision

S.No	*Dataset*	*KNN*	*Gaussian*	*Linear*	*GMM*	*DT*
1	Iris	0.033	0.0266	0.023	0.033	0.146
2	Heart	0.285	0.188	0.185	0.214	0.433
3	German credit score	0.274	0.334	0.25	0.325	0.34
4	Australian	0.268	0.172	0.153	0.23	0.41
5	Diabetes	0.235	0.49	0.307	0.485	0.334

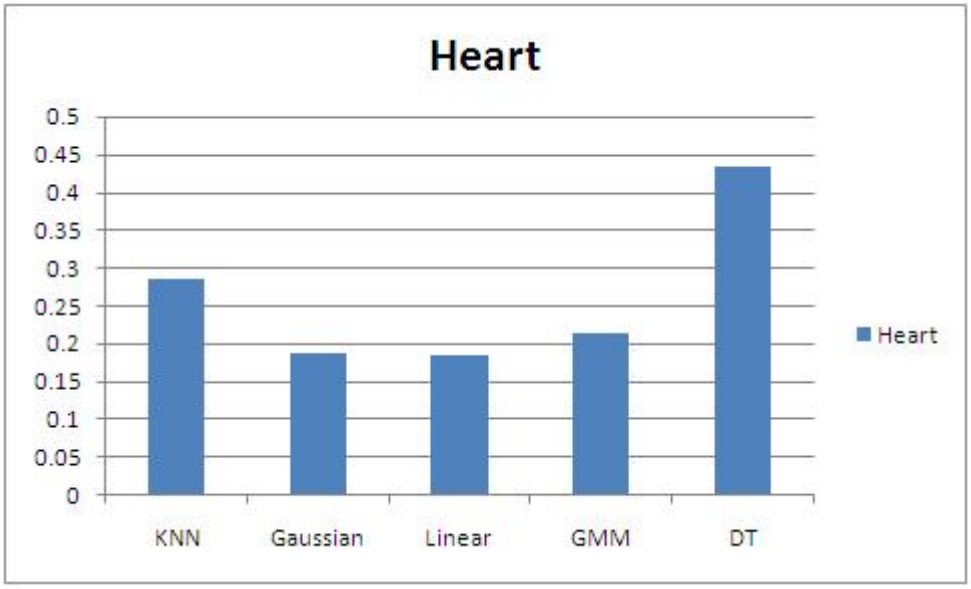

Figure 5.12: Heart dataset Error comparision

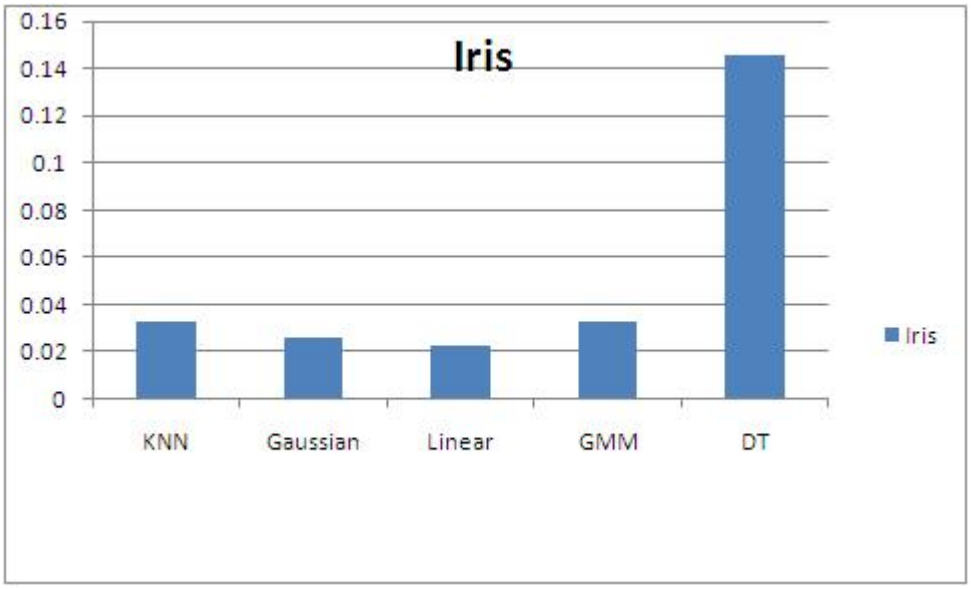

Figure 5.13: Iris dataset Error comparision

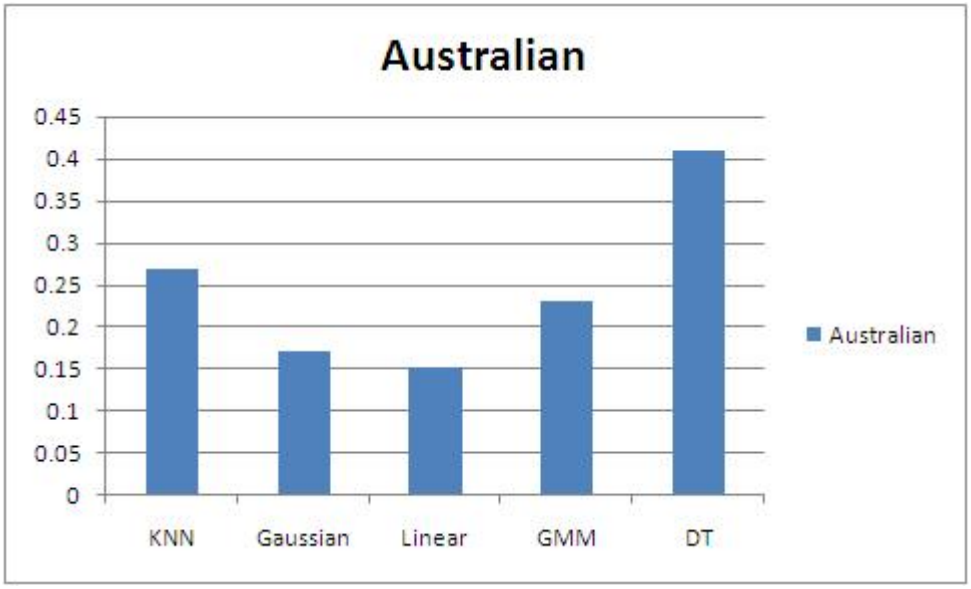

Figure 5.14: Australian dataset Error comparision

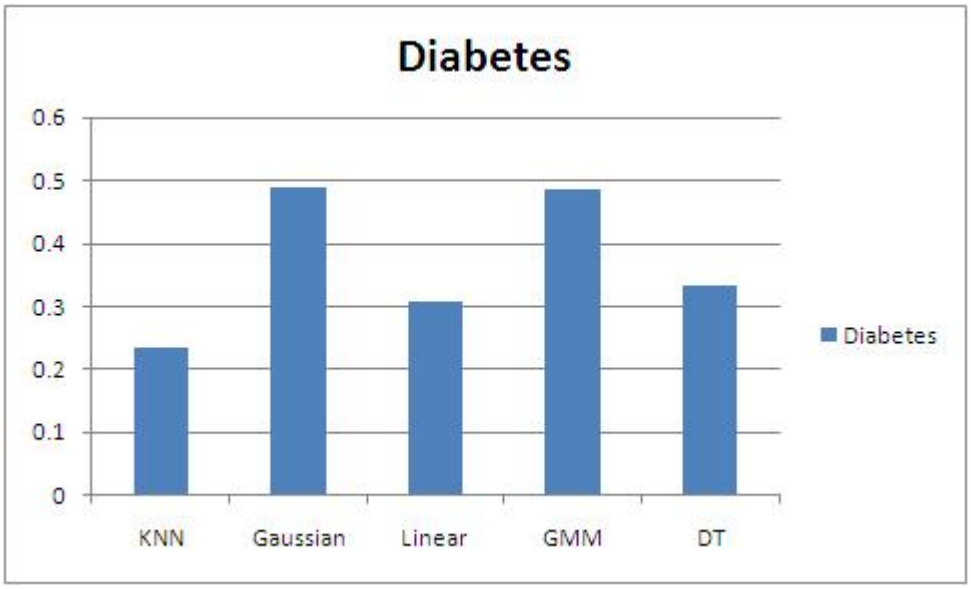

Figure 5.15: Diabetes dataset Error comparision

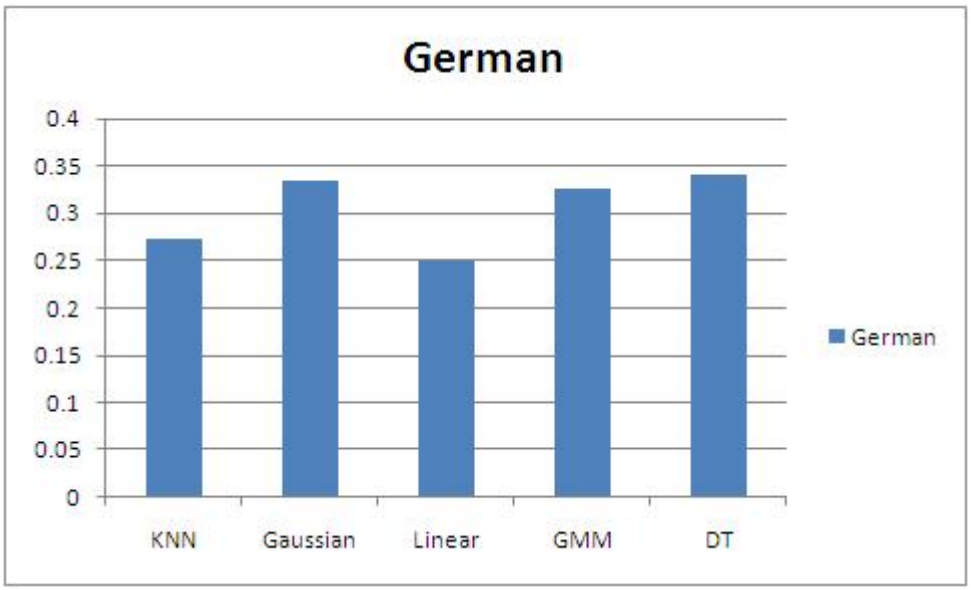

Figure 5.16: German dataset Error comparision

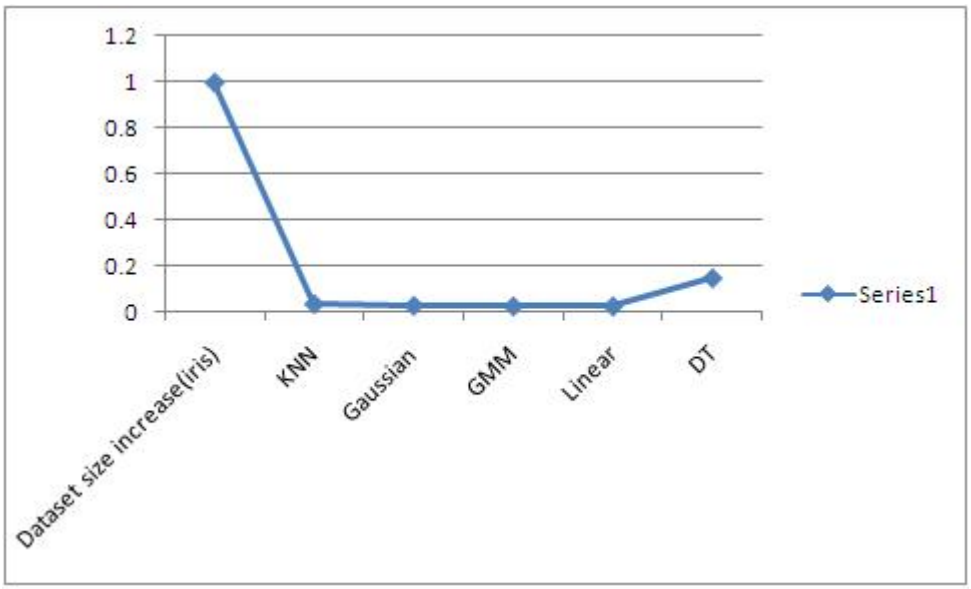

Figure 5.17: Iris dataset(Single) Error comparision

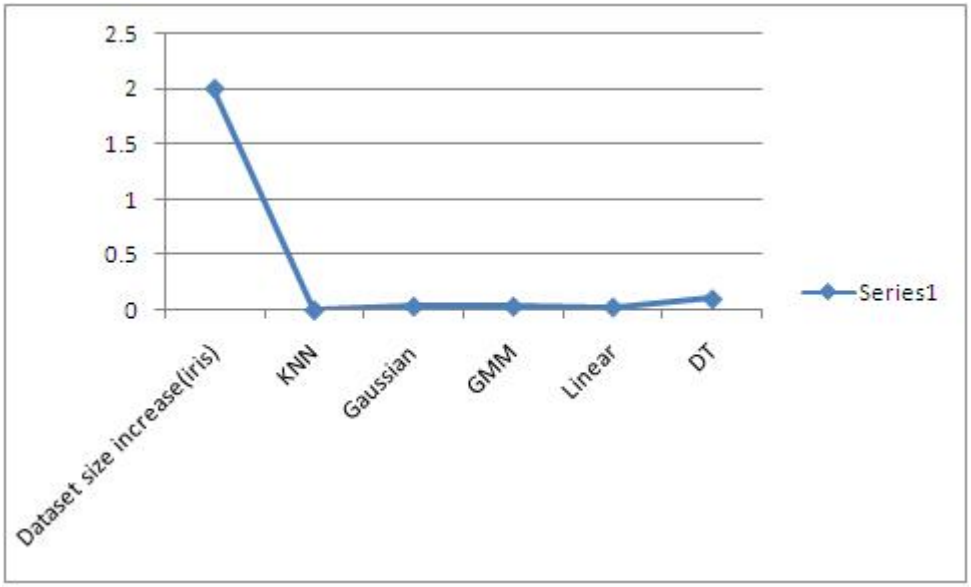

Figure 5.18: Iris dataset(Double) Error comparision

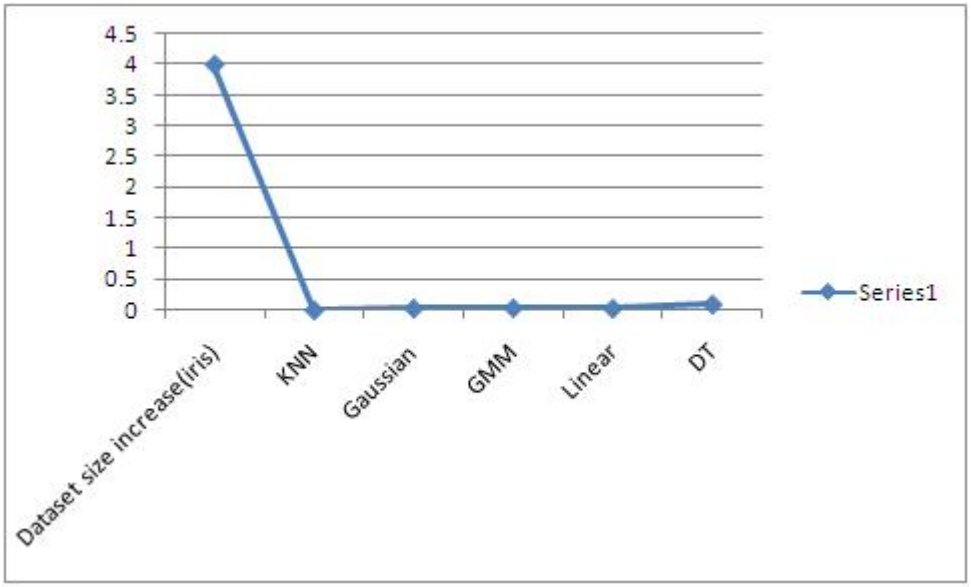

Figure 5.19: Iris dataset(Fourth) Error comparision

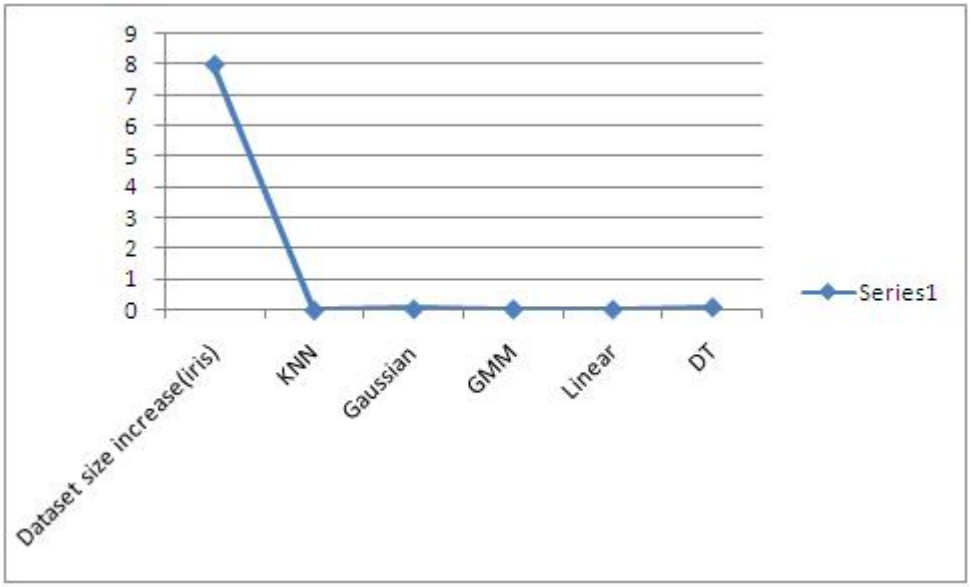

Figure 5.20: Iris dataset(Eighth) Error comparision

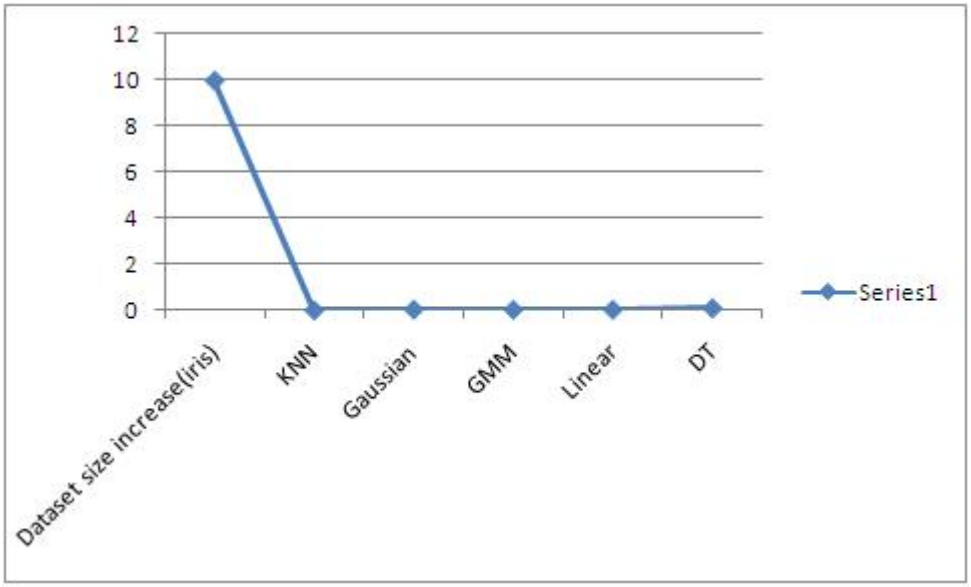

Figure 5.21: Iris dataset(Tenth) Error comparision

Table V: Error comparision

S.No	*Datasetsizeincrease(iris)*	*KNN*	*Gaussian*	*GMM*	*Linear*	*DT*
1	1	0.033	0.0266	**0.023**	0.023	0.146
2	2	**0**	0.033	0.033	0.026	0.1
3	4	**0**	0.028	0.033	0.025	0.093
4	8	**0**	0.033	0.032	0.025	0.097
5	10	**0**	0.033	0.026	0.026	0.092

5.1.4 Conclusion

Here it is observed that for different dataset accuracy vise different method is good.The experiment has observed that when size double accuracy result good compare first result for Naive bayes method.It has observed that when size increase(triple) dataset tested for KNN classifier and Naive bayes,accuracy result good compare to first result.

Chapter 6

Algorithm comparison II

6.1 Experimental Evaluation

Table includes datasets which are used in experiment to compare these algorithms in terms of Accuracy and generation time. Data sets are taken from the UCI Machine learning Repository , which have been discretised/ normalized.

Table I: Dataset

S.No.	Dataset	size
1	Diabetes	24KB
2	Adult	113KB
3	BRCA1 BRCA2	416KB
4	BRCA1 Sporadic	580KB
5	BRCA2 Sporadic	625KB
6	Australian	70KB
7	Liver disorders	23KB
8	Fourclass	23KB
9	Ionosphere	100KB

All the experiments are performed on my laptop using LibSVM(Java platform) Intel celron MProcessor 1.30GHZ,760MB RAM . Comparison is based on Accuracy. Analysis is shown below.

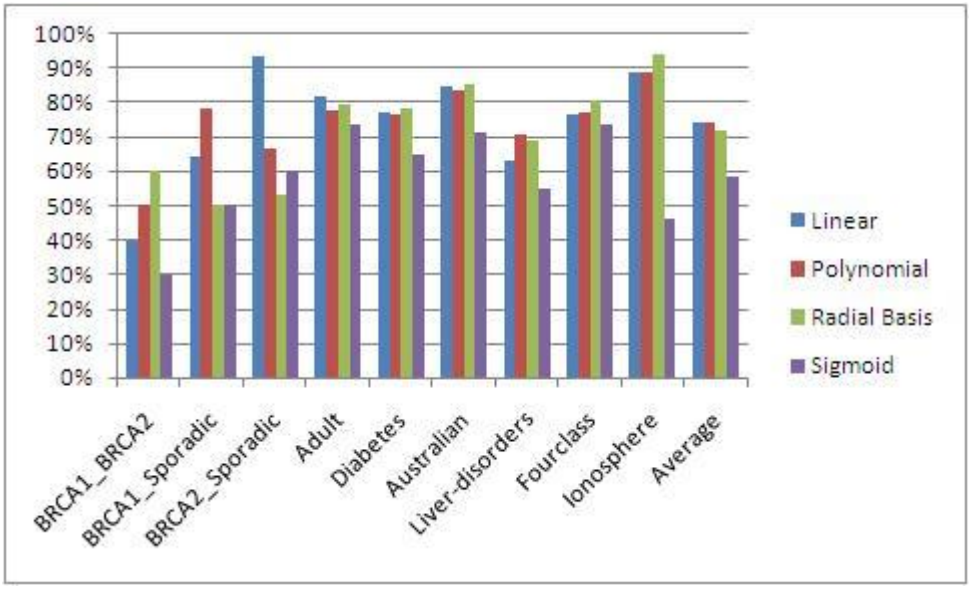

Figure 6.1: Single dataset accuracy comparision

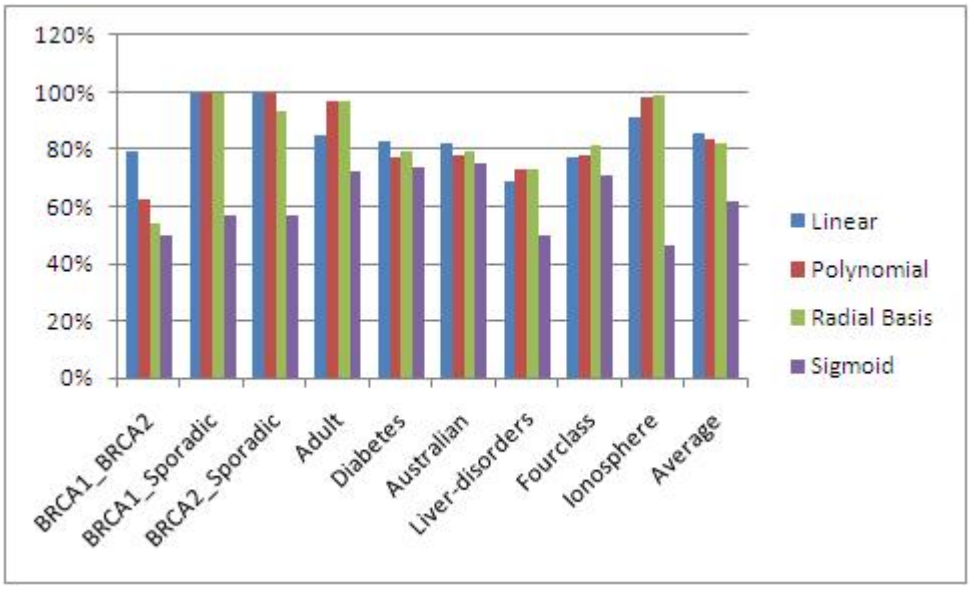

Figure 6.2: Double dataset accuracy comparision

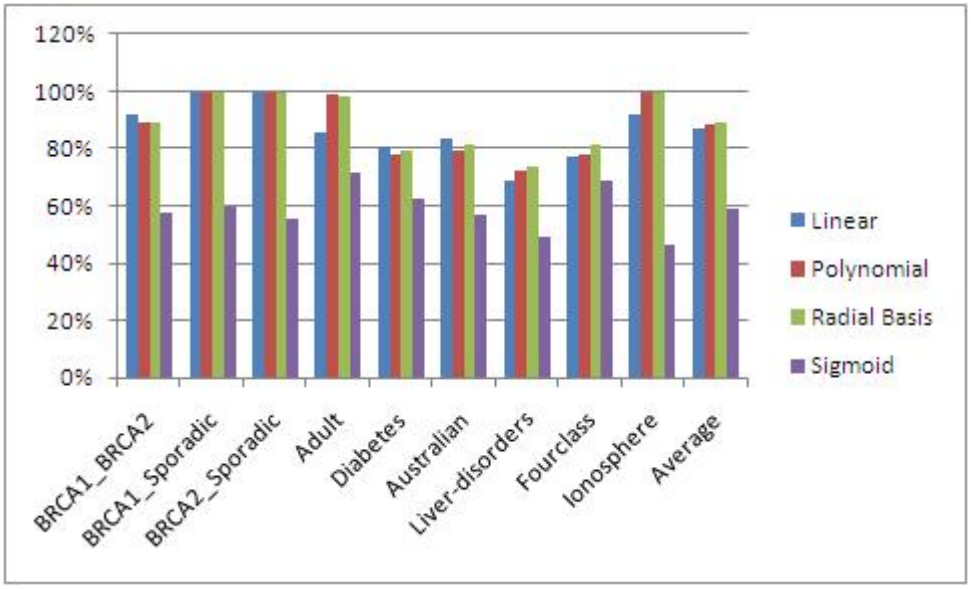

Figure 6.3: Triple dataset accuracy comparision

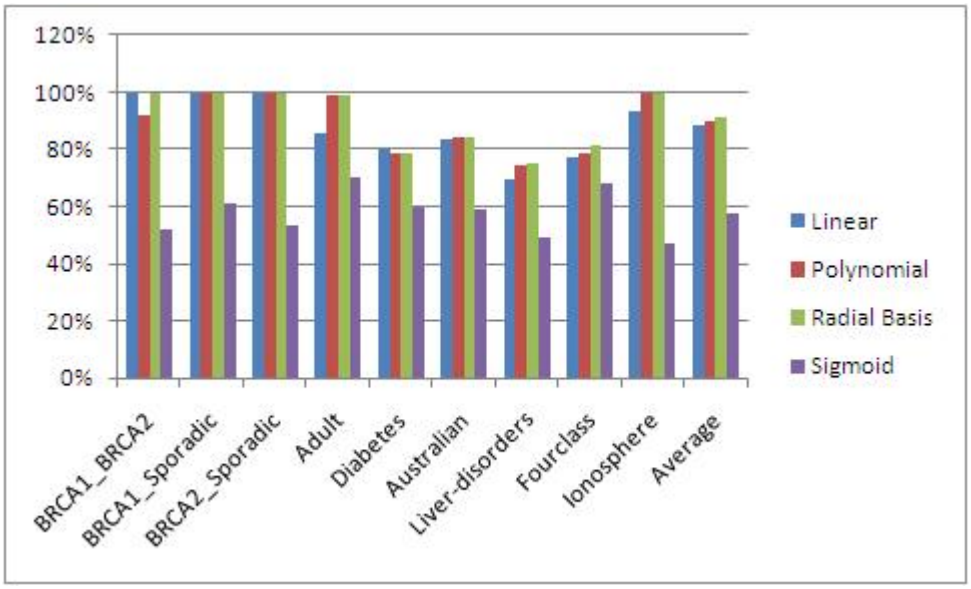

Figure 6.4: Fourth size dataset accuracy comparision

Table II: SVM model single data set size (Accuracy in percentage)

S.No	*Dataset*	*Linear*	*Polynomial*	*RadialBasis*	*Sigmoid*
1	Adult	82	77.94	79.43	73.45
2	Diabetes	77.07	76.69	78.13	64.71
3	Australian	85.07	83.91	85.36	71.44
4	Liver-disorders	63.18	70.72	68.98	54.78
5	BRCA1 BRCA2	40	50	60	30
6	BRCA1 Sporadic	64.28	78.57	50	64.28
7	BRCA2 Sporadic	93.33	66.66	53.33	60
8	Fourclass	76.79	77.26	80.51	73.43
9	Ionosphere	88.88	88.60	94.01	46.43
	Average	75	74	72	60

Table III: SVM model data set double size (Accuracy in percentage)

S.No	*Dataset*	*Linear*	*Polynomial*	*RadialBasis*	*Sigmoid*
1	Adult	85.01	96.54	96.79	72.39
2	Diabetes	82.99	77.19	79.06	73.45
3	Australian	82.36	77.88	79.25	74.76
4	Liver-disorders	68.69	72.75	73.33	50.00
5	BRCA1 BRCA2	79	63	54	50
6	BRCA1 Sporadic	100	100	100	39.28
7	BRCA2 Sporadic	100	100	93.33	56.66
8	Fourclass	76.85	77.78	81.14	70.53
9	Ionosphere	91.02	98.57	99	46.29
	Average	85	84	82	60

6.1.1 Observation

The experiment has observed for different dataset.The accuracy performance good for Linear method when dataset size single.

6.1.2 Observation size double

The experiment has observed for different dataset.The accuracy performance good for Linear method when dataset size double.

Table IV: SVM model dataset Triple size (Accuracy in percentage)

S.No	*Dataset*	*Linear*	*Polynomial*	*RadialBasis*	*Sigmoid*
1	Adult	85.35	98.62	98.40	71.50
2	Diabetes	80.44	77.91	79.05	62.45
3	Australian	83.18	79.04	81.61	57.16
4	Liver-disorders	68.98	72.96	74	49.27
5	BRCA1 BRCA2	92	89	89	58
6	BRCA1 Sporadic	100	100	100	59.52
7	BRCA2 Sporadic	100	100	100	55.55
8	Fourclass	76.95	78.22	81.39	68.63
9	Ionosphere	91.83	99.52	99.43	46.43
	Average	87	88	89	59

Table V: SVM model dataset fourth size (Accuracy in percentage)

S.No	*Dataset*	*Linear*	*Polynomial*	*RadialBasis*	*Sigmoid*
1	Adult	85.65	98.75	98.69	70.52
2	Diabetes	80.06	78.66	78.73	59.63
3	Australian	83.58	84.18	84.05	58.79
4	Liver-disorders	69.56	74.20	74.85	49.05
5	BRCA1 BRCA2	100	92	100	52
6	BRCA1 Sporadic	100	100	100	60.71
7	BRCA2 Sporadic	100	100	100	53.33
8	Fourclass	76.95	78.65	81.52	67.83
9	Ionosphere	93.16	99.71	99.43	46.86
	Average	88	90	91	58

6.1.3 Observation size triple

The experiment has observed for different dataset.The accuracy performance good for Radial Basis method when dataset size triple.

6.1.4 Observation size fourth

The experiment has observed for different dataset.The accuracy performance good for Radial Basis method when dataset size fourth times.

6.2 Conclusion

The experiment has observed SVM model for different dataset.The accuracy performance good for Linear method when dataset size single and double.The experiment has observed for different dataset.The accuracy performance good for Radial Basis method when dataset size triple and fourth times.Here also observed that accuracy performance increase when dataset size copy two times,three times.

The experiment has observed here SVM model for different dataset.So it is observed Linear method is good when dataset single and double.When it has sized triple that time Radial Basis method is good.So We can't say one method is good for all dataset.

Chapter 7

Conclusion and Future Scope

7.1 Conclusion

It is clear from the literature that there is no best classifier for all types of problems.The experiment for SVM method observed that when dataset size increase it's accuracy performance increase.The experiment has observed here SVM model for different dataset.So it is observed the result Linear method is good when dataset single and double.When it has sized triple that time Radial Basis method is good.Here it is observed that for different dataset accuracy vise different method is good.It has observed that when size double accuracy result good compare first result for Naive bayes method.It has observed that when size increase(triple) dataset tested for KNN classifier and Naive bayes,accuracy result good compare to first result.So We can't say one method is good for all dataset.For example, neural-networks are likely to perform well in parallel domains, while decision-tree algorithms are likely to perform well in sequential domains. Therefore, although a single induction algorithm cannot build the most accurate classifiers in all situations, some algorithms will perform better in specific domains.

7.2 Future work

The main focus of the experiment is on the mining algorithms to analyze a much accurate and efficient algorithm which takes less time.

Appendix A

Temporal data mining tasks

Data mining has been used in a wide range of applications. However, the possible objectives of data mining, which are often called tasks of temporal data mining, these tasks may be grouped as follows: prediction,

a. classification

b. clustering

c. search retrieval

d. pattern discovery

Once again, as was the case with models and patterns, this categorization is neither unique nor exhaustive, the only objective being to facilitate an easy discussion of the numerous techniques in the field.

A.0.1 Notation

A number of different notations are in use for time-series analysis:

$X = \{X1, X2, ...\}$is a common notation which specifies a time series X which is indexed by the natural numbers.

$$Y_t = \alpha_0 Y_{t-1} + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + ... + \alpha_p Y_{t-p} + \epsilon_t \tag{A.1}$$

To denote this, we will index the observations with the letter t rather than the letter i. Our data will be observations on Y1, Y2, ...Yt, ...where t indexes the day, month, year, or any time interval.Time series is about dependence. We use correlation as a measure of dependence. Although we have only one variable, we can compute the correlation between Y_t and Y_{t-1} or between Y_t and Y_{t-2}.The correlations between Ys at different times are called autocorrelations.

Website References

[1] en.wikipedia.org/wiki/Cross-validation

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/

[3] http://www.biomedcentral.com/content/pdf/1471-2105-7-S4-info.pdf

[4] http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1780131/

[5] www.ics.uci.edu/~mlearn//

[6] http://mfgn.usm.edu/ebl/svm/

[7] en.wikipedia.org/wiki/Pattern_recognition

References

[1] Agrawal R, Srikant R 1995 Mining sequential patterns. In Proc. 11th Int. Conf. on Data Engineering, (Washington, DC: IEEE Comput. Soc.)

[2] Azoff, E.M. Neural Network Time Series Forecasting of Financial Markets John Wiley and Sons Ltd, 1994.

[3] Banzhaf, Wolfgang; Nordin, Peter; Keller, Robert; Francone, Frank (1998) Genetic Programming - An Introduction, Morgan Kaufmann, San Francisco, CA.

[4] Box, George; Jenkins, Gwilym (1976), Time series analysis: forecasting and control, rev. ed., Oakland, California: Holden-Day

[5] Brachman, R.J., and Anand, T. The Process Of Knowledge Discovery In Databases: A Human-Centered Approach. In Advances In Knowledge Discovery And Data Mining , eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, pp. 37-57.

[6] Buntine, W. Graphical Models For Discovering Knowledge. In Advances In Knowledge Discovery And Data Mining, eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, pp. 59-82.

[7] Chih-Chung Chang, Chih-Jen Lin. LIBSVM, a library for support vector machines. 2001.

[8] Cortes C, Vapnik V. Support-vector network. Machine Learning. 1995;20:273297.

[9] C.M. van der Walt and E. Barnard,"Data characteristics that determine classifier perfromance", in Proceedings of the Sixteenth Annual Symposium of the Pattern Recognition Association of South Africa, pp.160-165, 2006

[10] FARMER, D., SIDOROWICH, J. Predicting chaotic time series. Physical Review Letters, v. 59, p. 845-848, 1987.

[11] Fausett L. Fundamentals of Neural Networks: Architectures, Algorithms and Applications, Prentice-Hall, New Jersey, 1994.

[12] Fayyad, U.M., Piatetsky-Shapiro, G., and Smyth, P. From Data Mining To Knowledge Discovery: An Overview. In Advances In Knowledge Discovery And Data Mining , eds. U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, AAAI Press/The MIT Press, Menlo Park, CA., 1996, pp. 1-34.

[13] Flajolet P, Guivarch Y, Szpankowski W, Vallee B 2001 Hidden pattern statistics. In Lecture notes in computer science; Proc. 28th Int. Colloq. on Automata, Languages and Programming (London: Springer-Verlag) vol. 2076, pp 152165

[14] Fogel, David B (2006), Evolutionary Computation: Toward a New Philosophy of Machine Intelligence, IEEE Press, Piscataway, NJ. Third Edition

[15] Fraser, Alex S. (1957). "Simulation of Genetic Systems by Automatic Digital Computers. I. Introduction". Australian Journal of Biological Sciences 10: 484491.

[16] Gershenfeld, Neil (2000), The nature of mathematical modeling, Cambridge: Cambridge Univ. Press, ISBN 978-0521570954, OCLC 174825352

[17] Geurts, P. (2001). Pattern extraction for time series classification. In proceedings of Principles of Data Mining and Knowledge Discovery, 5 th European Conference. Freiburg, Germany, Sept 3-5. pp 115-127.

[18] Hedenfalk I, Duggan D, Chen Y, Radmacher M, Bittner M, Simon R, Meltzer P, Gusterson B, Esteller M, Raffeld M, et al. Gene-expression profiles in hereditary breast cancer. N Engl J Med. 2001;344:539548. doi: 10.1056/NEJM200102223440801.

[19] Holland, John H (1975), Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor

[20] I. Kononenko (1993) Inductive and Bayesian learning in medical diagnosis, Applied Artificial Intelligence, Vol. 7, pp. 317-337.

[21] Joachims T. Making large-scale SVM learning practical. In: Scholkopf B, Burges CJC, Smola AJ, editor. Advances in Kernel Methods Support Vector Learning. Cambridge: MIT Press; 1998.

[22] Karuna Pande Joshi,Analysis of Data Mining Algorithms,March1997

[23] Koza, John (1992), Genetic Programming: On the Programming of Computers by Means of Natural Selection, MIT Press. ISBN 0-262-11170-5

[24] Laxman S, Sastry P S, Unnikrishnan K P 2005 Discovering frequent episodes and learning hidden markov models: A formal connection. IEEE Trans. Knowledge Data Eng. 17: 15051517 Palacios R. and Gupta

[25] Mitchell, Melanie, (1996), An Introduction to Genetic Algorithms, MIT Press, Cambridge, MA.

[26] Monterola, C., Roxas, R.M., and Carreon-Monterola, S. (2008). Characterizing the Effect of Seating Arrangement on Classroom Learning Using Neural Networks. Complexity, 14(4), 26-33. ISSN 1076-2782.

[27] Noble WS. Support vector machine applications in computational biology. In: Scholkopf B, Tsuda K, Vert JP, editor. Kernel Methods in Computational Biology. MIT Press; 2004. pp. 71-92.

[28] Poli, R., Langdon, W. B., McPhee, N. F. (2008). A Field Guide to Genetic Programming.

[29] Roger Bridgman's defence of neural networks

[30] Schmitt, Lothar M (2001), Theory of Genetic Algorithms, Theoretical Computer Science 259: 1-61

[31] Scholkopf B, Smola A, Williamson RC, Bartlett PL. New support vector algorithms. Neural Computation. 2000;12:12071245. doi: 10.1162/089976600300015565.

[32] Scholkopf B, Platt JC, Shawe-Taylor J, Smola AJ, Williamson RC. Estimating the support of a high-dimensional distribution. Neural Computation. 2001;13:14431471. doi: 10.1162/089976601750264965.

[33] SRIVATSAN LAXMAN and P S SASTRY,A survey of temporal data mining,Department of Electrical Engineering, Indian Institute of Science, Bangalore 560 012, India

[34] Vose, Michael D (1999), The Simple Genetic Algorithm: Foundations and Theory, MIT Press, Cambridge, MA