**CHAPTER 1**

**INTRODUCTION**

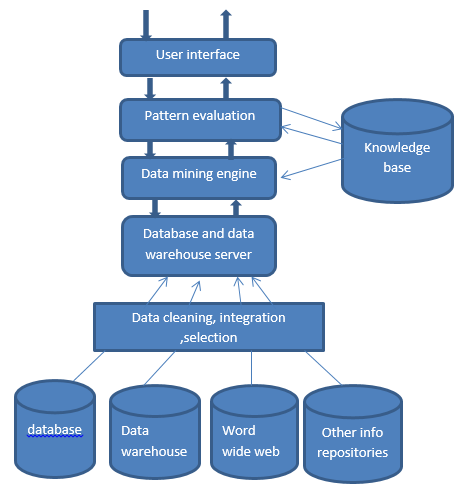
**1.1 OVERVIEW**

Data mining is the computing process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems. It is an interdisciplinary subfield of computer science. The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.

Aside from the raw analysis step, it involves database and data management aspects, data pre-processing, model and inference consideration metrics, complexity considerations, post-processing of discovered structures, visualization, and online updating.

Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD. The actual data mining task is the semi-automatic or automatic analysis of large quantities of data to extract previously unknown, interesting patterns such as groups of data records (cluster analysis), unusual records (anomaly detection), and dependencies (association rule mining, sequential pattern mining). This usually involves using database techniques such as spatial indices.

These patterns can then be seen as a kind of summary of the input data, and may be used in further analysis or, for example, in machine learning and predictive analytics. For example, the data mining step might identify multiple groups in the data, which can then be used to obtain more accurate prediction results by a decision support system. Neither the data collection, data preparation, nor result interpretation and reporting is part of the data mining step, but do belong to the overall KDD process as additional steps.



Data

Fig 1.1 Overview of data mining

**Real time examples of data mining:**

As the importance of data analytics continues to grow, companies are finding more and more applications for Data Mining and Business Intelligence. Here we take a look at 5 real life applications of these technologies and shed light on the benefits they can bring to your business.

**Service providers:**

The first example of Data Mining and Business Intelligence comes from service providers in the mobile phone and utilities industries. Mobile phone and utilities companies use Data Mining and Business Intelligence to predict ‘churn’, the terms they use for when a customer leaves their company to get their phone/gas/broadband from another provider. They collate billing information, customer services interactions, website visits and other metrics to give each customer a probability score, then target offers and incentives to customers whom they perceive to be at a higher risk of churning.

**Retail:**

Another example of Data Mining and Business Intelligence comes from the retail sector. Retailers segment customers into ‘Recency, Frequency, Monetary’ (RFM) groups and target marketing and promotions to those different groups. A customer who spends little but often and last did so recently will be handled differently to a customer who spent big but only once, and also some time ago. The former may receive a loyalty, upsell and cross-sell offers, whereas the latter may be offered a win-back deal, for instance.

**E-commerce:**

Perhaps some of the most well -known examples of Data Mining and Analytics come from E-commerce sites. Many E-commerce companies use Data Mining and Business Intelligence to offer cross-sells and up-sells through their websites. One of the most famous of these is, of course, Amazon, who use sophisticated mining techniques to drive there, ‘People who viewed that product, also liked this’ functionality.

**Supermarkets:**

Supermarkets provide another good example of Data Mining and Business Intelligence in action. Famously, supermarket loyalty card programmers are usually driven mostly, if not solely, by the desire to gather comprehensive data about customers for use in data mining. One notable recent example of this was with the US retailer Target. As part of its Data Mining programed, the company developed rules to predict if their shoppers were likely to be pregnant. By looking at the contents of their customers’ shopping baskets, they could spot customers who they thought were likely to be expecting and begin targeting promotions for nappies (diapers), cotton wool and so on. The prediction was so accurate that Target made the news by sending promotional coupons to families who did not yet realize (or who had not yet announced) they were pregnant! You can read the full story.

**Crime agencies:**

The use of Data Mining and Business Intelligence is not solely reserved for corporate applications and this is shown in our final example. Beyond corporate applications, crime prevention agencies use analytics and Data Mining to spot trends across myriads of data – helping with everything from where to deploy police manpower (where is crime most likely to happen and when?), who to search at a border crossing (based on age/type of vehicle, number/age of occupants, border crossing history) and even which intelligence to take seriously in counter-terrorism activities.

**1.1.1 KNOWLEDGE DISCOVERY IN DATABASE KDD PROCESS**

The term **Knowledge Discovery in Databases**, or KDD for short, refers to the broad process of finding knowledge in data, and emphasizes the "high-level" application of particular data mining methods.

* **Data Cleaning** In this step, the noise and inconsistent data is removed.
* **Data Integration** In this step, multiple data sources are combined.
* **Data Selection** In this step, data relevant to the analysis task are retrieved from the database.
* **Data Transformation** In this step, data is transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations.
* **Data Mining** In this step, intelligent methods are applied in order to extract data patterns.
* **Pattern Evaluation** In this step, data patterns are evaluated.
* **Knowledge Presentation** In this step, knowledge is represented.

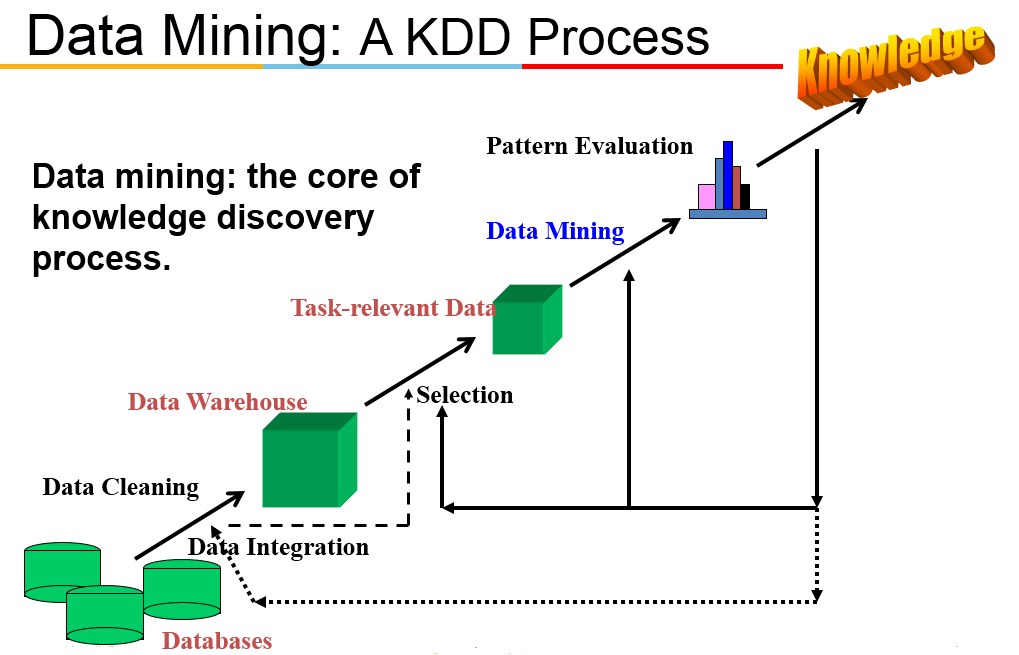


Fig 1.2 KDD PROCESS

**1.1.2 SCOPE OF DATA MINING**

**Data mining in market basket analysis**

Data mining technique is used in MBA(Market Basket Analysis).When the customer want to buying some products then this technique helps us finding the associations between different items that the customer put in their shopping buckets. Here the discovery of such associations that promotes the business technique .In this way the retailers uses the data mining technique so that they can identify that which customers intension (buying the different pattern).In this way this technique is used for profits of the business and also helps to purchase the related items.

**Data mining in Medical Science**

The data mining effectively used in the diagnosis of lung abnormality that may be cancerous or benign the data mining algorithms significantly reduce patient’s risks and diagnosis costs. Using the prediction algorithms the observed prediction accuracy was 100% for 91.3% cases. The use of data mining in health care is the widely used application of data mining. The medical data is complex and difficult to analyze. A REMIND (Reliable Extraction and Meaningful Inference from Non-structured Data) system integrates the structured and unstructured clinical data in patient records to automatically create high quality structured clinical data. To adopt the high quality technique, we can mined the existing patient records to support guidelines and give compliance to improve patient care.

**In the education system**

A huge number of universalities are established by the order of UG. Every day a millennium of students are enrolls across the country. We believe that data mining technology can help bridging knowledge gap in higher educational systems. The hidden patterns, associations, and anomalies that are discovered by data mining techniques from educational data can improve decision making processes in higher educational systems.

**In Sports data mining**

In the world, a huge number of games are available where each and every day the national and international games are to be scheduled, where a huge number of data’s are to be maintained .The data mining tools are applied to give the information as and when we required.

**1.1.3 DATA MINING TECHNIQUES**

Below are 5 data mining techniques that can help you create optimal results.

**Classification**This analysis is used to retrieve important and relevant information about data, and metadata. It is used to classify different data in different classes. Classification is similar to clustering in a way that it also segments data records into different segments called classes. [3] But unlike clustering, here the data analysts would have the knowledge of different classes or cluster. So, in classification analysis you would apply algorithms to decide how new data should be classified. A classic example of classification analysis would be our Outlook email. In Outlook, they use certain algorithms to characterize an email as legitimate or spam.

**Association** It refers to the method that can help you identify some interesting relations (dependency modeling) between different variables in large databases. [4] This technique can help you unpack some hidden patterns in the data that can be used to identify variables within the data and the concurrence of different variables that appear very frequently in the dataset. Association rules are useful for examining and forecasting customer behavior. It is highly recommended in the retail industry analysis. This technique is used to determine shopping basket data analysis, product clustering, and catalogue design and store layout. In IT, programmers use association rules to build programs capable of machine learning.

Association (or relation) is probably the better known and most familiar and straightforward data mining technique. Here, you make a simple correlation between two or more items, often of the same type to identify patterns. For example 1, when tracking people's buying habits, you might identify that a customer always buys cream when they buy strawberries, and therefore suggest that the next time that they buy strawberries they might also want to buy cream. Building association or relation-based data mining tools can be achieved simply with different tools. Example 2, within Info Sphere Warehouse a wizard provides configurations of an information flow that is used in association by examining your database input source, decision basis, and output information.

**Anomaly** This refers to the observation for data items in a dataset that do not match an expected pattern or an expected behavior. [1] Anomalies are also known as outliers, novelties, noise, deviations and exceptions. Often they provide critical and actionable information. An anomaly is an item that deviates considerably from the common average within a dataset or a combination of data. These types of items are statistically aloof as compared to the rest of the data and hence, it indicates that something out of the ordinary has happened and requires additional attention. This technique can be used in a variety of domains, such as intrusion detection, system health monitoring, fraud detection, fault detection, event detection in sensor networks, and detecting eco-system disturbances. Analysts often remove the anomalous data from the dataset top discover results with an increased accuracy.

**Clustering**   
 The cluster is actually a collection of data objects those objects are similar within the same cluster. [2] That means the objects are similar to one another within the same group and they are rather different or they are dissimilar or unrelated to the objects in other groups or in other clusters. Clustering analysis is the process of discovering groups and clusters in the data in such a way that the degree of association between two objects is highest if they belong to the same group and lowest otherwise. A result of this analysis can be used to create customer profiling.

**Regression**  
 In statistical terms, a regression analysis is the process of identifying and analyzing the relationship among variables. [5] It can help you understand the characteristic value of the dependent variable changes, if any one of the independent variables is varied. This means one variable is dependent on another, but it is not vice versa. It is generally used for prediction and forecasting.

**Prediction**

Prediction is a wide topic and runs from predicting the failure of components or machinery, to identifying fraud and even the prediction of company profits. [6] Used in combination with the other data mining techniques, prediction involves analyzing trends, classification, pattern matching, and relation. By analyzing past events or instances, you can make a prediction about an event. Using the credit card authorization, for example, you might combine decision tree analysis of individual past transactions with classification and historical pattern matches to identify whether a transaction is fraudulent. Making a match between the purchase of flights to the US and transactions in the US, it is likely that the transaction is valid.

**Sequential patterns**

Often used over longer-term data, sequential patterns are a useful method for identifying trends, or regular occurrences of similar events. [8] For example, with customer data you can identify that customers buy a particular collection of products together at different times of the year. In a shopping basket application, you can use this information to automatically suggest that certain items be added to a basket based on their frequency and past purchasing history.

**Decision trees**

Related to most of the other techniques (primarily classification and prediction), the decision tree can be used either as a part of the selection criteria, or to support the use and selection of specific data within the overall structure.[10] Within the decision tree, you start with a simple question that has two (or sometimes more) answers. Each answer leads to a further question to help classify or identify the data so that it can be categorized, or so that a prediction can be made based on each answer.

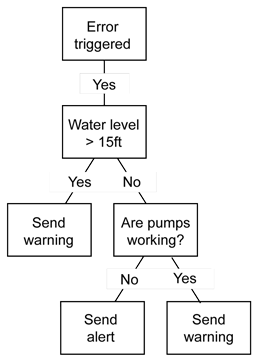


Fig 1.3 decision tree

**Classification:**

You can use classification to build up an idea of the type of customer, item, or object by describing multiple attributes to identify a particular class.[11] For example, you can easily classify cars into different types (sedan, 4x4, convertible) by identifying different attributes (number of seats, car shape, driven wheels). Given a new car, you might apply it into a particular class by comparing the attributes with our known definition. You can apply the same principles to customers, for example by classifying them by age and social group. Additionally, you can use classification as a feeder to, or the result of, other techniques. For example, you can use decision trees to determine a classification. Clustering allows you to use common attributes in different classifications to identify clusters.

## **Data implementations and preparation:**

Data mining itself relies upon building a suitable data model and structure that can be used to process, identify, and build the information that you need. Regardless of the source data form and structure, structure and organize the information in a format that allows the data mining to take place in as efficient a model as possible.

Consider the combination of the business requirements for the data mining, the identification of the existing variables (customer, values, and country) and the requirement to create new variables that you might use to analyze the data in the preparation step. [5].

You might compose the analytical variables of data from many different sources to a single identifiable structure (for example, you might create a class of a particular grade and age of customer, or a particular error type).

Depending on your data source, how you build and translate this information is an important step, regardless of the technique you use to finally analyze the data. This step also leads to a more complex process of identifying, aggregating, simplifying, or expanding the information to suit your input data.

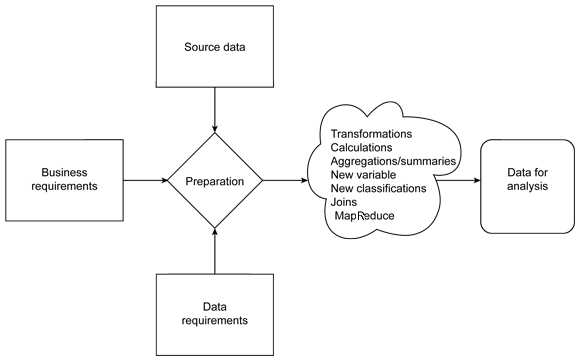


Fig 1.4 data preparation

**1.1.4 DATA MINING MODEL**

These patterns and trends can be collected and defined as a **data mining model*.*** Mining models can be applied to specific scenarios, such as:

* **Forecasting**: Estimating sales, predicting server loads or server downtime
* **Risk and probability**: Choosing the best customers for targeted mailings, determining the probable break-even point for risk scenarios, assigning probabilities to diagnoses or other outcomes
* **Recommendations**: Determining which products are likely to be sold together, generating recommendation
* **Finding sequences**: Analyzing customer selections in a shopping cart, predicting next likely events
* **Grouping**: Separating customers or events into cluster of related items, analyzing and predicting affinities.

**1.2 ABOUT THE PROJECT**

The main objectives of this project is the decision tree is to create a model that predicts the value of a target variable based on several input variables. This type of learning is one of the most successful techniques for supervised classification learning. Nowadays, there is an ever-increasing migration of people to urban areas. Health care service is one of the most challenging aspects that is greatly affected by the vast influx of people to city centers. Consequently, cities around the world are investing heavily in digital transformation in an effort to provide healthier ecosystems for people. In such a transformation, millions of homes are being equipped with smart devices which generate massive volumes of fine-grained and indexical data that can be analyzed to support smart city services. In this paper, we propose a model that utilizes big data as a means of learning and discovering human activity patterns for health care applications. We propose the use of cluster analysis, and prediction to determine the patients who are in need of immediate care. The comparison of two algorithms are presented in detail in this paper along with the accuracy of short and long-term predictions.

**1.3 SCOPE OF THE PROJECT**

**Healthcare Data solutions**

With the help of big data, the vast amount of data can be stored systematically. Now doctors and other healthcare practitioners can make informed decisions as they have access to a wide range of data. Of course, the data generated will grow by leaps and bounds, and newer systems will be able to process it quickly and cost effectively.

**Big data to fight cancer**

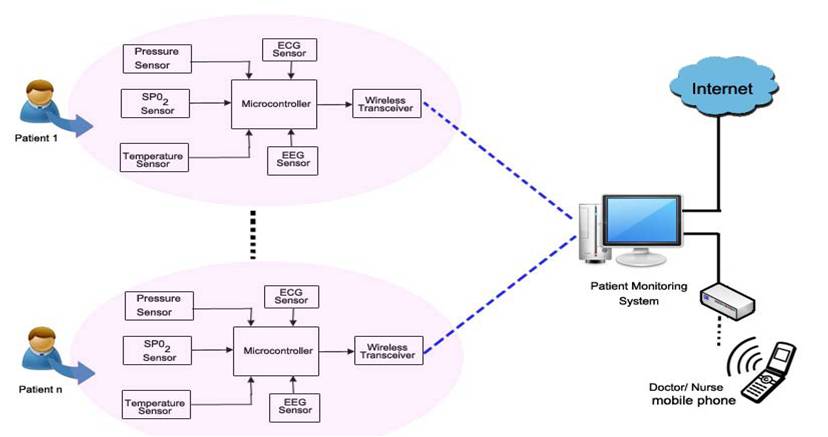
Cancer is rapidly crippling people across the world. Big data can help to fight cancer more effectively. Healthcare providers will have enhanced ability to detect and diagnose diseases in their early stages, assigning more effectual therapies based on a patient’s genetic makeup, and regulate drug doses to minimize side effects and improve effectiveness. It will also provide great support for parallelization and help in mapping the 3 billion DNA base pairs.

**Monitoring patient vitals**

The application of big data makes it easier for hospital staff to work more efficiently. Sensors are used besides patient beds to continuously monitor blood pressure, heartbeat and respiratory rate. Any change in pattern is quickly alerted to doctors and healthcare administrators.

**Smoother Hospital Administration**

Healthcare administration becomes much smoother with the help of big data. It helps to reduce the cost of care measurement, provide the best clinical support, and manage the population of at-risk patients. It also helps medical experts analyse data from diverse sources. It helps healthcare providers conclude the deviations among patients and the effects treatments have on their health.

 Fig 1.5 Monitoring patient vitals

**Healthcare Intelligence**

Big Data can be used for healthcare Intelligence application. This will help hospitals, payers and healthcare agencies augment their competitive advantages by developing smart business solutions.

**Fraud Prevention and Detection**

Big data helps to prevent a wide range of errors on the side of health administrators in the form of wrong dosage, wrong medicines, and other human errors. It will also be particularly useful to insurance companies. They can prevent a wide range of fraudulent claims of insurance.

**1.4 Organizations of Chapters**

The report is organized as follows:

**Chapter 2: Literature survey**

This chapter describe about survey of the various base works and the Pre-processing concepts to improve the Accuracy of prediction and cluster analysis on various medical datasets.

**Chapter 3: Proposed work**

In this section, we illustrate our proposed system for predicting cancer patients and the comparison of algorithms based on accuracy.

**Chapter 4: Software and Hardware requirements**

In this chapter, the requirements are specified which are used to develop the software.

**Chapter 5: Modules description**

This chapter describes about the modules and module description.

**Chapter 6: Testing**

This chapter explains about the different types of testing that can be performed after the development stages of the project.

**Chapter 7: Parameter evaluation**

In this chapter, it describes the performance of the proposed system using simulation graph.

**Chapter 8: Conclusion**

The conclusion of the report is presented in this chapter. It also specifies the enhancements that could be done in the days yet to come.

**CHAPTER 2**

**LITERATURE SURVEY**

In this section, we review existing work in the literature, which employ various data mining algorithms to group and predict from a given dataset.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| s.no | Year | Name and author | Technology/  Algorithm | Disadvantage |
| 1.  2.  3.  4.  5.  6  7  8  9  10  11 | 2017  2015  2017  2017  2016  2014  2015  2014  2015  2016  2017 | An output-based knowledge transfer approach and its application in bladder cancer prediction Guanjin Wang1,2, Guangquan Zhang1, Kup-Sze Choi2, Kin-Man Lam3, and Jie Lu1  “Smart meter profiling for health applications. C. Chalmers, W. Hurst, M. Mackay, and P. Fergus,  A study on prediction of breast cancer recurrence using data mining techniques. Uma Ojha  Dr. Savita Goel.  Prediction of Breast Cancer using Voting Classifier Technique U. Karthik Kumar, M.B. Sai Nikhil  Big Data Analytics for Demand Response Clustering Over Space and Time. Charalampos Chelmis Jahanvi Kolte Viktor K. Prasanna.  Comparative Study on D  Mining Classification Methods for Cervical Cancer Prediction Yulia Ery Kurniawati Adhistya Erna Permanasari Silmi Fauziati  Intelligent Breast Cancer Prediction Model Using Data Mining Techniques  Runjie Shen Yuanyuan Yang Fengfeng Shao  Detecting Household Activity Patterns from Smart Meter Data Jing Liao, Lina Stankovic  Prediction of breast cancer recurrence using data mining techniques. Uma Ojha Dr. Savita Goel.  Leveraging Smart Grid Technology for Home Health Care. Thomas Cashen, and Samuel Russ.  Mining Human Activity Patterns from Smart Home Big Data for Healthcare Applications  Abdulsalam Yassine, Shailendra Singh, Atif Alamri | Least squares-support vector machine(LS-SVM)  Voted Perception Classifier(VPC)  Decision tree C5.0 and Support vector machine.  Voting classifier technique.  Classification and Regression Tree (CART).  A feature selection method, INTERACT and SVM.  Time series clustering.  NALM (Non-Intrusive Appliance Load Monitoring) and Dempster-Shafer (D-S) theory of evidence.  Random Neural Network(RNN)  Kmeans, Naïve Bayes, SVM.  Incremental Data-Mining, Association Rules, Prediction. | Not robust.  No real time system for detection of depression. Complex.    Complex on large dataset. Not robust.  Over fitting problem.  Does not handle imbalanced dataset.  Not robust.  Cluster formed depend on the choice of seed.  Over fitting problem and not robust.  Does not handle imbalanced dataset.  Less accuracy, complex and not robust.  Dataset is imbalanced, not robust, scalability issues. |

**2.2 EXISTING SYSTEM:**

The system presents a model for recognizing human activities patterns from low resolution smart meters data. Occupants’ habits and behavior follow a pattern that could be used in health applications to track the wellbeing of individuals living alone or those with self-limiting conditions. Most of these activities can be learned from appliance-to appliance and appliance-to-time associations. We presented incremental frequent mining and prediction model based on Bayesian network. In our current work, through experiments, we built the model to operate on any quantum of time. From the experiment results we have demonstrated the applicability of the proposed model to correctly detect multiple appliance usage and make short and long term prediction at high accuracy.

**2.2.1 SYSTEM ARCHITECTURE**

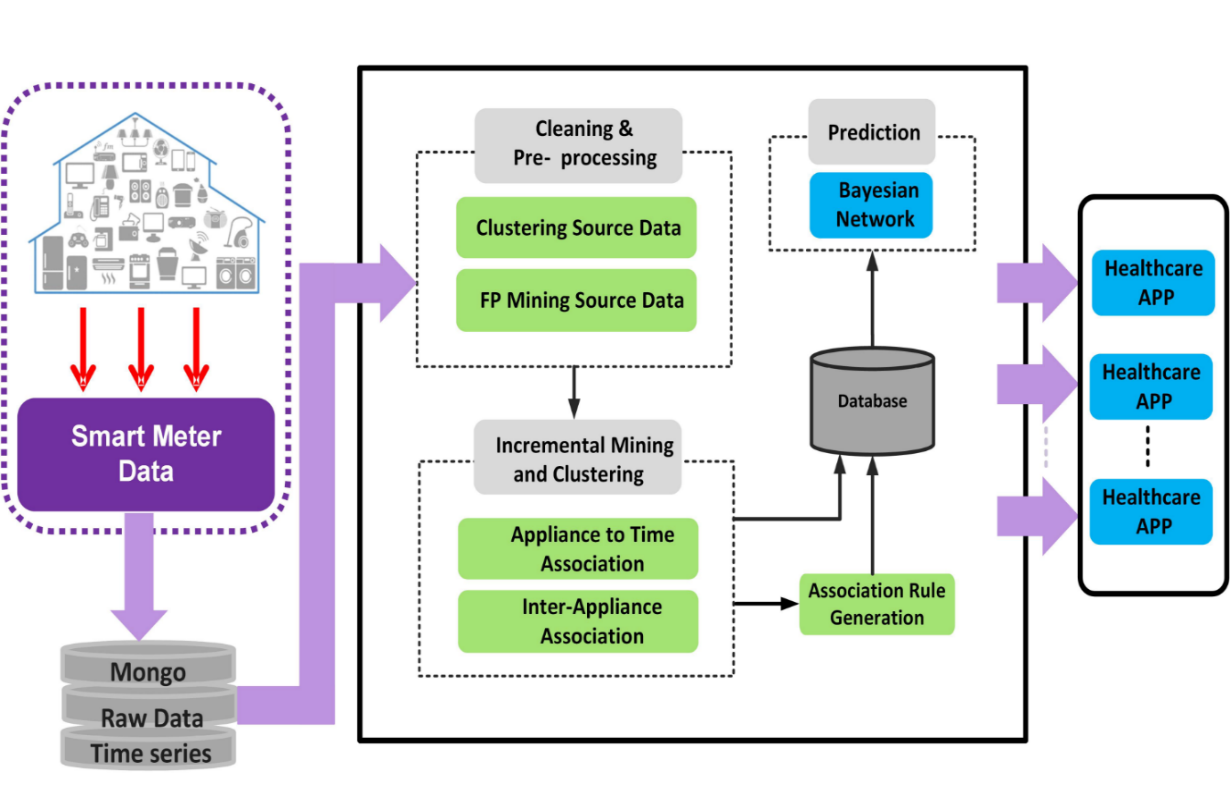


Fig 2.1 System architecture

**2.2.2 ALGORITHM USED:**

ALGORITHM1: INCREMENTAL FREQUENT PATTEN MINING

Require: transaction database (DB), frequent pattern discovered database (FP\_DB)

Ensues: incremental discovery of frequent patterns, stored in frequent patterns discovery

database (FP\_DB)

1: for all transaction data slice db24 in quanta of 24 hours in database DB do {data is processed in slices of 24 hour period}

2: determine database size

Database\_sizedb24 for data slice/quantum db24

3: mine frequency patterns in FP\_DB db24 using extended FP-growth approach

4: for all frequency pattern FP in FP\_DB db24 do

5: search a frequent pattern FP\_DB

6: If frequent pattern found then

7: Update frequent pattern in FP\_DB

8: else

9: add a new frequency pattern to FP\_DB

10: end if

11: end for

12: for all frequency patterns is database FP\_DB increment database size by database\_size db24

13: end for



Fig 2.2 Bayesian network for activity prediction

**Naive Bayes**

The Naive Bayes algorithm is widely used algorithm for document classification. Two types of feature (POS and word association) were extracted and integrated and classified using the base classifier. The feature selection methods used for fast selection and classification performance using the proposed ensemble methods showed a contradiction in the performance of SVM.

The focus is on writing Cantonese, a written variety of Chinese. The machine learning model classifies the lexicon as a positive or negative one extracted from the review. The seed set is provided initially. The naive Bayes classifier surprisingly achieves better performance than SVM.

**2.2.3 DRAWBACKS IN EXISTING SYSTEM**

* Medical datasets are often not balanced in their class labels.
* Most of the existing classification methods tend to perform poorly on dataset which is extremely imbalanced.
* An increasing number of applications deployed over the cloud operate on datasets which is large and complex.
* It becomes difficult to gather, store, analyze and visualize. So there arise a scalability issue.

**CHAPTER 3**

**PROPOSED SYSTEM**

**3.1 INTRODUCTION**

In this project, we suggest by comparing J48 Random Forest and Naive Bayesian as classifier ensemble that can incorporate different base classifiers into classifier ensembles models for classification problems. This project suggest that the impact of using different base classifiers on classification accuracy of Random Forest classifier ensemble. Classifier ensembles with five base classifier has used on five medical data sets. These results evaluated and compared choosing different type of decision tree algorithms for base classifier. The reliability of classification for most of datasets and classifier ensembles is increased when we select the appropriate j48 random forest classifier achieves the minimum time required to build models.

**3.2 PROPOSED MODEL**

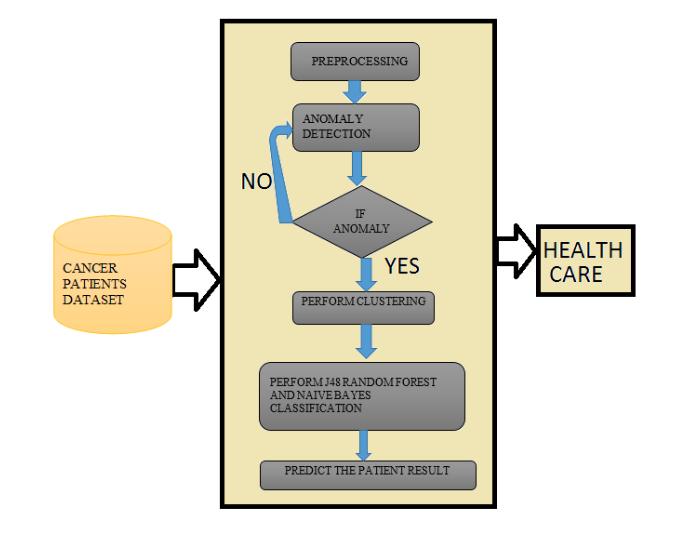
****

Fig 3.1 Proposed model

**3.3 ALGORITHM USED**

**Supervised Learning**

Classification is most frequently used popular data mining technique. Classification used to predict the possible outcome from a given data set on the basis of a defined set of attributes and a given predictive attributes. The given dataset is found to be the training dataset consist of independent variables (dataset related properties) and a dependent attribute (predicted attribute). A training dataset created model test on the test corpus contains the same attributes but no predicted attribute. Accuracy of model checked that how accurate it is making a prediction. Product features and sentenced words are extracted using Double Propagation Algorithm.

**Machine Learning Based Approaches**

In a machine learning based classification, documents require two sets: one is the training and the other is the test set. For training a set automatic classifier is used that learns various characteristics of documents and a test set is used to validate the automatic classifier performance. A number of machine learning techniques have been adopted to classify the reviews. Machine learning techniques like Naïve Bayes (NB), maximum entropy (ME) and support vector machines (SVM) have achieved great success in text categorization.

**J48 Random Forest**

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of over fitting to their training set.

The first algorithm for random decision forests was created by Tin Kam Ho using the random subspace method, which, in Ho's formulation, is a way to implement the "stochastic discrimination" approach to classification proposed by Eugene Kleinberg.

An extension of the algorithm was developed by Leo Breiman and Adele Cutler, and "Random Forests" is their trademark. The extension combines Breiman's "bagging" idea and random selection of features, introduced first by Ho and later independently by Amit and German in order to construct a collection of decision trees with controlled variance.

Although random forests have been inherently designed to work only with multidimensional data, it has been shown that one can also use them for arbitrary objects (like graphs or time series) with the use of only pairwise similarities between objects. This variant is referred to as a similarity forest.

**Adaboost**

Boosting is a family of methods for improving the performance of a “weak” classifier by using it within an ensemble structure, the most prominent member of which is AdaBoost.

In Boosting methods, a set of weights is maintained across the objects in the data set, so that objects that have been difficult to classify acquire more weight, forcing subsequent classifiers to focus on them.

These methods works by repeatedly running a learning algorithm on various distributions over the training data, and then combining the classifiers produced by the learner into the single composite classifier.

**3.4 DATASET USED**

Catalogue of Somatic Mutations in Cancer (COSMIC) is an online database of somatically acquired mutations found in human cancer. Somatic mutations are those that occur in non-germ line cells that are not inherited by children. COSMIC, an acronym of Catalogue of Somatic Mutations in Cancer, curates data from papers in the scientific literature and large scale experimental screens from the Cancer Genome Project at the Sanger Institute.

Census Dataset is the procedure of systematically acquiring and recording information about the members of a given population. It is a regularly occurring and official count of a particular population. The term is used mostly in connection with national population and housing censuses, other common censuses include agriculture, business, and traffic censuses. It recommends that population censuses be taken at least every 10 years to cover census topics to be collected, official definitions, classifications and other useful information to co-ordinate international practice.

* 1. **ADVANTAGES OF PROPOSED SYSTEM**
* It is simple to understand and interpret and able to handle both numerical and categorical data, which requires little data preparation, for possible to validate a model using statistical tests, performs well with large datasets.
* It is robust, which means that performs well even if its assumptions are somewhat violated by the true model from which the data were generated.

**CHAPTER 4**

**SYSTEM REQUIREMENTS**

**4.1 MINIMUM HARDWARE REQUIREMENTS:**

* System : Pentium IV 2.4 GHz.
* Hard Disk : 40 GB.
* Monitor : 15 inch VGA Color.
* Mouse : Logitech Mouse.
* Ram : 512 MB
* Keyboard : Standard Keyboard

**4.2 MINIMUM SOFTWARE REQUIREMENTS:**

* Operating System : Windows XP.
* Platform : Java TECHNOLOGY
* Tool : NetBeans 6.9.1, Hadoop 2.3
* Front End : Jdk 1.7

**CHAPTER 5**

**MODULE DESCRIPTION**

The modules used to implement the proposed model are,

* Preprocessing
* Feature extraction
* Clustering
* J48 Random forest classification
* Naïve Bayes classification
* Result comparison

**5.1 Preprocessing:**

* In this step initially the data set were loaded .And normalized the data .were the empty data set, redundant data were removed in this process .And send to the anomaly detection.

**5.2 Feature extraction:**

* In this step, we perform Anomaly detection where the data sets of the cancer patients are classified into various stages. In this stage 4 patients are filtered as they are said to be in the critical state.
* Performs Anomaly detection and groups the patients belonging to stage4.
* Stage 4 refers to the state where the patient needs immediate care.

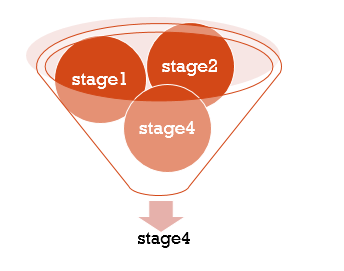


Fig 5.1 feature extraction

**5.3 Clustering:**

* By clustering the data sets were further classified as stage 4starting state, stage 4 middle state, stage 4 end state with the use of K means algorithm.

Stage 4

High level

Middle level

Starting level

Fig 5.2 clustering

**5.4 J48-Random forest classification:**

* In this project we are using J48-random forest algorithm. By using this the accuracy of the result were predicted easily and efficiently. On this various metrics were used for prediction (precision, recall, TP rate F -measure) with this metrics confusion matrix were prepared.

**Confusion matrix:**

* A confusion matrix is a table that is often used to describe the performance of a classification model (or “classifier”) on a set of test data for which the true values are known.
* It allows the visualization of the performance of an algorithm.  
  It allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. Most performance measures are computed from the confusion matrix.
* A confusion matrix is a summary of prediction results on a classification problem.  
  The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix.
* The confusion matrix shows the ways in which your classification model It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.
* In the field of [machine learning](https://en.wikipedia.org/wiki/Machine_learning) and specifically the problem of [statistical classification](https://en.wikipedia.org/wiki/Statistical_classification), a **confusion matrix**, also known as an error matrix,[[4]](https://en.wikipedia.org/wiki/Confusion_matrix#cite_note-4) is a specific table layout that allows visualization of the performance of an algorithm, typically a [supervised learning](https://en.wikipedia.org/wiki/Supervised_learning) one (in [unsupervised learning](https://en.wikipedia.org/wiki/Unsupervised_learning) it is usually called a **matching matrix**). Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class (or vice versa).[[2]](https://en.wikipedia.org/wiki/Confusion_matrix#cite_note-Powers2011-2) The name stems from the fact that it makes it easy to see if the system is confusing two classes (i.e. commonly mislabeling one as another).
* It is a special kind of [contingency table](https://en.wikipedia.org/wiki/Contingency_table), with two dimensions ("actual" and "predicted"), and identical sets of "classes" in both dimensions (each combination of dimension and class is a variable in the contingency table).



Table 5.1 class for confusion matrix

**Definition of terms**:

|  |  |  |
| --- | --- | --- |
| Positive (P) | : | Observation is positive (for example: is an apple). |
| Negative (N) | : | Observation is not positive (for example: is not an apple). |
| True Positive (TP) | : | Observation is positive, and is predicted to be positive. |
| FalseNegative (FN) | : | Observation is positive, but is predicted negative. |
| True Negative (TN) | : | Observation is negative, and is predicted to be negative. |
| False Positive (FP) | : | Observation is negative, but is predicted positive. |

**Classification rate/accuracy:**

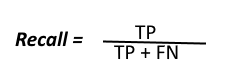
Classification Rate or Accuracy is given by the relation:



However, there are problems with accuracy. It assumes equal costs for both kinds of errors. A 99% accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem.

**Recall:**

Recall can be defined as the ratio of the total number of correctly classified positive divide to the total number of positive examples. High Recall indicates the class is correctly.



{\displaystyle {\text{recall}}={\frac {|\{{\text{relevant documents}}\}\cap \{{\text{retrieved documents}}\}|}{|\{{\text{relevant documents}}\}|}}}

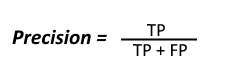
For example, for a text search on a set of documents, recall is the number of correct results divided by the number of results that should have been returned.

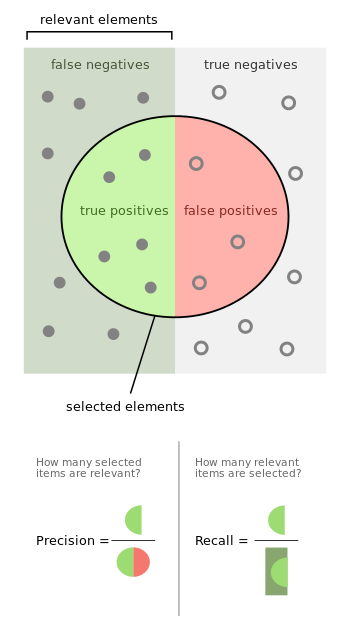
In binary classification, recall is called sensitivity. It can be viewed as the probability that a relevant document is retrieved by the query.

It is trivial to achieve recall of 100% by returning all documents in response to any query. Therefore, recall alone is not enough but one needs to measure the number of non-relevant documents also, for example by also computing the precision.

**Precision:**

To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labeled as positive is indeed positive (small number of FP).





**Fig 5. Precision and recall**

**Precision and recall:**

In [pattern recognition](https://en.wikipedia.org/wiki/Pattern_recognition), [information retrieval](https://en.wikipedia.org/wiki/Information_retrieval) and [binary classification](https://en.wikipedia.org/wiki/Binary_classification), **precision** (also called [positive predictive value](https://en.wikipedia.org/wiki/Positive_predictive_value)) is the fraction of relevant instances among the retrieved instances, while **recall** (also known as [sensitivity](https://en.wikipedia.org/wiki/Sensitivity_and_specificity)) is the fraction of relevant instances that have been retrieved over the total amount of relevant instances. Both [precision and recall](https://en.wikipedia.org/wiki/Precision_and_recall) are therefore based on an understanding and measure of [relevance](https://en.wikipedia.org/wiki/Relevance). Suppose a computer program for recognizing dogs in photographs identifies eight dogs in a picture containing 12 dogs and some cats. Of the eight dogs identified, five actually are dogs (true positives), while the rest are cats (false positives).

The program's precision is 5/8 while its recall is 5/12. When a [search engine](https://en.wikipedia.org/wiki/Search_engine_(computing)) returns 30 pages only 20 of which were relevant while failing to return 40 additional relevant pages, its precision is 20/30 = 2/3 while its recall is 20/60 = 1/3. So, in this case, precision is "how useful the search results are", and recall is "how complete the results are".

In [statistics](https://en.wikipedia.org/wiki/Statistics), if the [null hypothesis](https://en.wikipedia.org/wiki/Null_hypothesis) is that all items are *irrelevant* (where the hypothesis is accepted or rejected based on the number selected compared with the sample size), absence of [type I and type II errors](https://en.wikipedia.org/wiki/Type_I_and_type_II_errors) corresponds respectively to maximum precision (no false positive) and maximum recall (no false negative). The above pattern recognition example contained 8 − 5 = 3 type I errors and 12 − 5 = 7 type II errors. Precision can be seen as a measure of exactness or *quality*, whereas recall is a measure of completeness or *quantity*.

In simple terms, high precision means that an algorithm returned substantially more relevant results than irrelevant ones, while high recall means that an algorithm returned most of the relevant results.

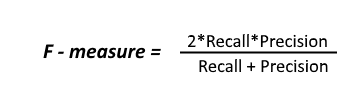
This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives.

**Low recall, high precision:**

This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP)

**F-measure:**   
Since we have two measures (Precision and Recall) it helps to have a measurement that represents both of them. We calculate an F-measure which uses Harmonic Mean in place of

The **F measure** (F1 score or **F** score) is a **measure** of a test's accuracy and is defined as the weighted harmonic mean of the precision and recall of the test.



**5.5 Naïve Bayes classification:**

The Naïve Bayes algorithm is applied on the clustered dataset and the confusion matrix is generated to determine the performance metrics.

**5.6 Result comparison:**

At last with the predicted value, the graph was prepared .on this graph the accuracy of both (J48-Random forest algorithm, Naïve Bayes classification algorithm) was compared.

**CHAPTER 6**

**TESTING**

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

**6.1 System Testing**

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified?. Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Validation **:** Are we doing the right job?

Verification **:** Are we doing the job right?

Software testing should not be confused with debugging. Debugging is the process of analyzing and localizing bugs when software does not behave as expected. Although the identification of some bugs will be obvious from playing with the software, a methodical approach to software testing is a much more thorough means for identifying bugs. Debugging is therefore an activity which supports testing, but cannot replace testing.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis

Looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it vital success of the system testing objectives, there are several rules that can serve as testing objectives. They are

Testing is a process of executing a program with the intend of finding an error.A good test case is one that has high possibility of finding an undiscovered error.A successful test is one that uncovers an undiscovered error.

     If a testing is conducted successfully according to the objectives as stated above, it would uncovered errors in the software also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

* For correctness
* For implementation efficiency
* For computational complexity

     Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

**6.2 UNIT TESTING**

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm’s execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified.

**6.3 INTEGRATION TESTING**

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop.

After unit testing in Sell-Soft System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

**6.4 OUTPUT TESTING OR USER ACCEPTANCE TESTING**

The system considered is tested for user acceptance; here it should satisfy the firm’s need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points

* Input Screen Designs,
* Output Screen Designs,
* Online message to guide the user and the like.

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

**CHAPTER 7**

**PERFORMANCE EVALUATION**

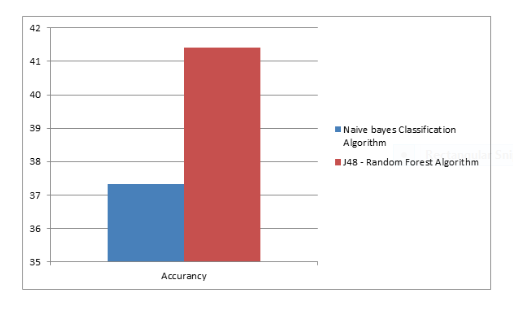
By finding the confusion matrix these are parameters to find it:

* **Accuracy:** the proportion of the total number of predictions that were correct.
* **Positive Predictive Value or Precision:** the proportion of positive cases that were correctly identified.
* **Negative Predictive Value**: the proportion of negative cases that were correctly identified.
* **Sensitivity or Recall:** the proportion of actual positive cases which are correctly identified.
* **Specificity:** the proportion of actual negative cases which are correctly identified.

**PRECISION**:

SPECIFICITY:

**ACCURACY:**



**CHAPTER 8**

**CONCLUSION**

We suggested comparing J48 Random Forest and Naive Bayesian as classifier ensemble that can incorporate different base classifiers into classifier ensembles models for classification problems. This project suggest that the impact of using different base classifiers on classification accuracy of Random Forest classifier ensemble. Classifier ensembles with five base classifier has used on five medical data sets. These results evaluated and compared choosing different type of decision tree algorithms for base classifier. The reliability of classification for most of datasets and classifier ensembles is increased when we select the appropriate j48 random forest classifier achieves the minimum time required to build models.

**REFERENCE**

[1] N. United, “World urbanization prospect.” United Nation, 2014. [Online]. Available: http://dl.acm.org/citation.cfm?id=308574. 308676

[2] M. S. Hossain, “Cloud-supported cyber-physical localization framework for patients monitoring,” IEEE Systems Journal, vol. 11, no. 1, pp. 118–127, March 2017.

[3] M. S. Hossain, G. Muhammad, W. Abdul, B. Song, and B. Gupta, “Cloud-assisted secure video transmission and sharing framework for smart cities,” Elsevier, Future Generation Computer Systems Journal, April 2017.

[4] J. Liao, L. Stankovic, and V. Stankovic, “Detecting household activity patterns from smart meter data,” in Intelligent Environments (IE), 2014 International Conference on, 6 2014, pp. 71–78.

[5] A. Yassine, A. A. N. Shirehjini, and S. Shirmohammadi, “Smart meters big data: Game theoretic model for fair data sharing in deregulated smart grids,” IEEE Access, vol. 3, pp. 2743–2754, 2015.

[6] A. Yassine and S. Shirmohammadi, “Measuring users’ privacy payoff using intelligent agents,” in 2009 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, May 2009, pp. 169–174.

[7] “A business privacy model for virtual communities,” Inderscience Publishers International journal of web based communities, Clifton, Christopher (2010).

[8] V. Karthikeyani, I. Parvin, K. Tajudin, and I. Shahina Begam, “Comparative of Data Mining Classification Algorithm in Diabetes Disease Prediction”. International journal of computer application 2012.12.26-31

[9] F. Mutaher and Ba-Alwi, “Comparative Study for Analysis the Prognostic in Hepatitis Data: Data Mining Approach”, International Journal of Scientific and Engineering Research, Vol 4, Issue 8, August 2013 680ISSN 2229-5518.

[10] C. Shah and A. Jivani, “Comparison of Data Mining Classification Algorithms for Breast Cancer Prediction”.4th ICCCNT 2013, July 4-6, India

[11] http://www.cs.waikato.ac.nz/ml/weka/ 2017 IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials.

[12] B. Padmapriya, T. Velmurugan, “A Survey on Breast Cancer Analysis Using Data Mining Techniques”, Computational Intelligence and Computing Research (ICCIC), 2014 IEEE International Conference on. IEEE, 2014.

[13] S. Gosh, S. Mondal, B. Ghosh, “A Comparative Study of Breast Cancer Detection Based on SVM And MLP BPN Classifier”, Automation, Control, Energy and Systems (ACES), 2014 First International Conference on. IEEE, 2014.

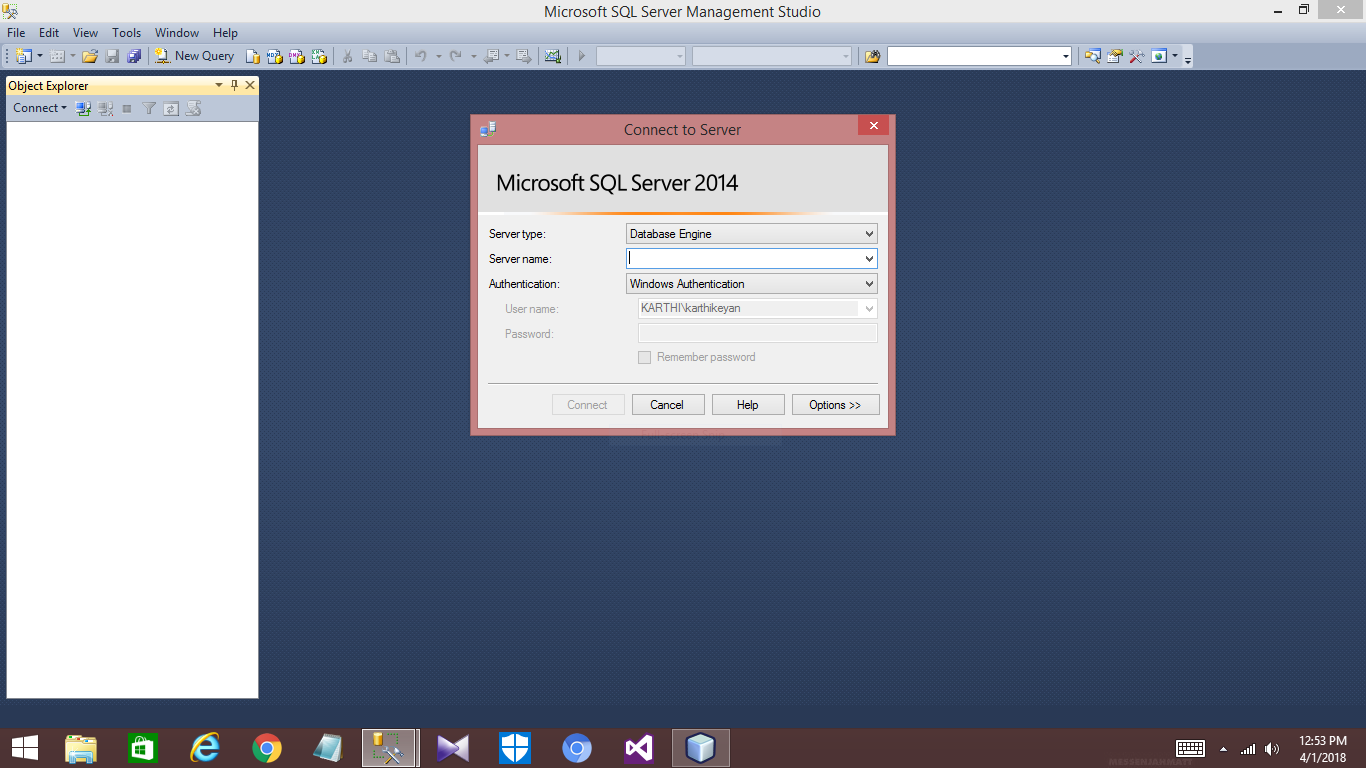
[14] N. Rathore, D. Tomar, S. Agarwal, “Predicting the Survivability Of Breast Cancer Patients Using Ensemble Approach.” Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014 International Conference on. IEEE, 2014.

[15] J. Han, M. Kamber, “Data Mining Concepts and Techniques”, third edition, Morgan Kaufmann Publishers an imprint of Elsevier.

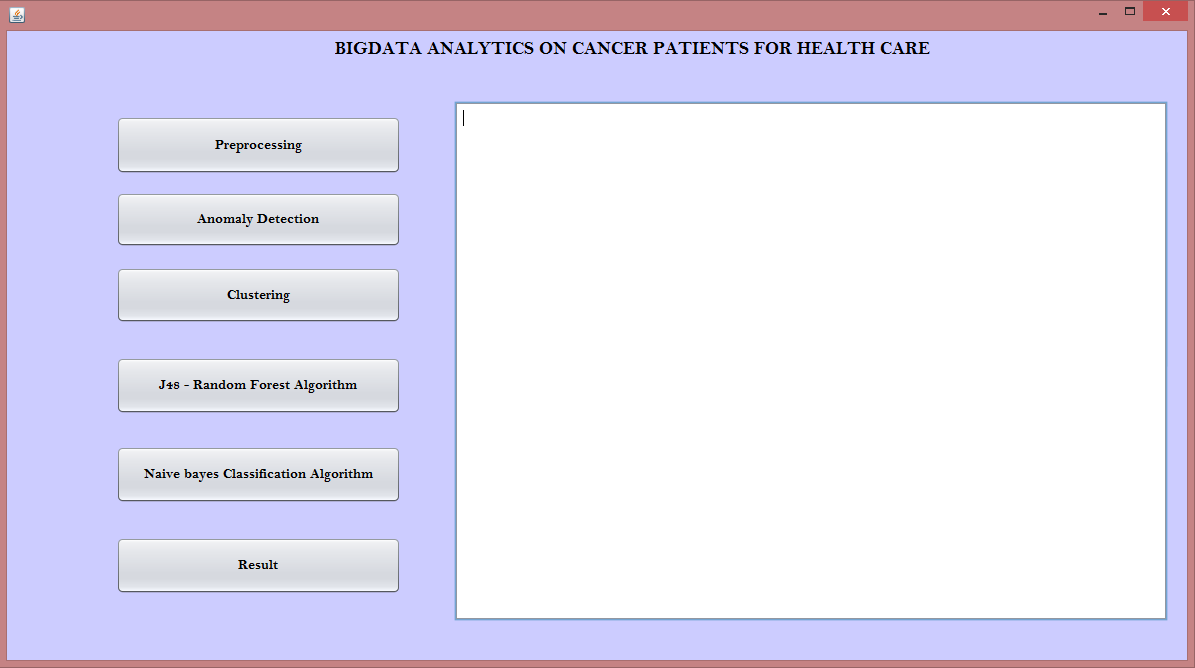
**APPENDIX**

**SCREENSHOTS OF THE PROJECT**

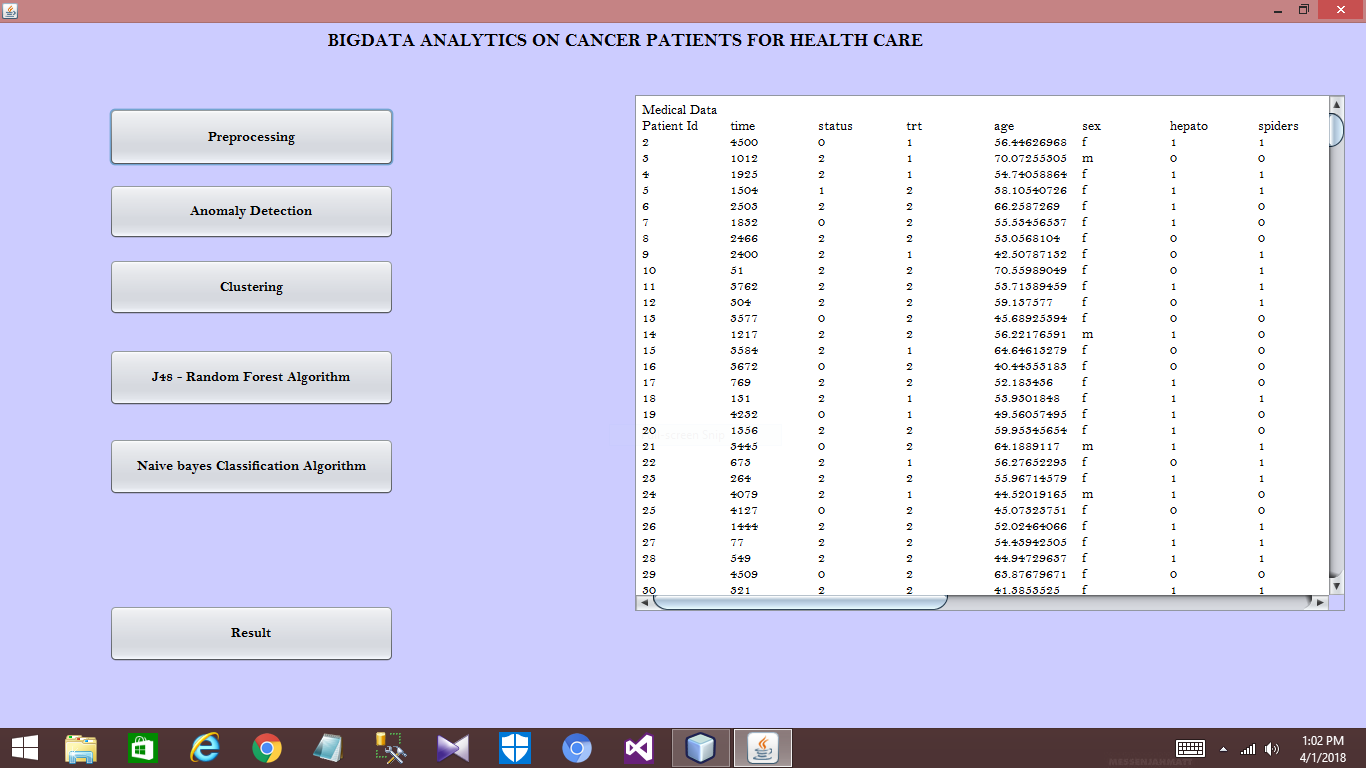
Connecting to SQL server

****

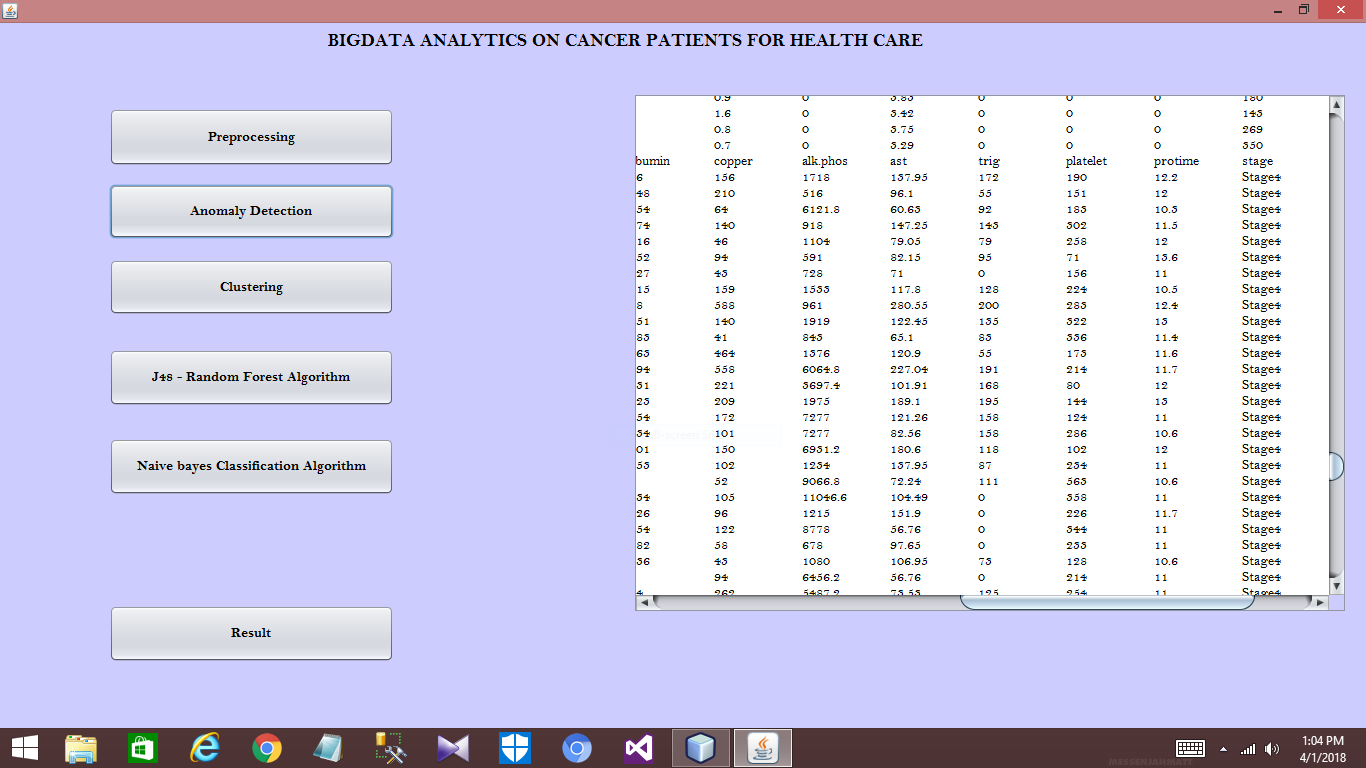
Home



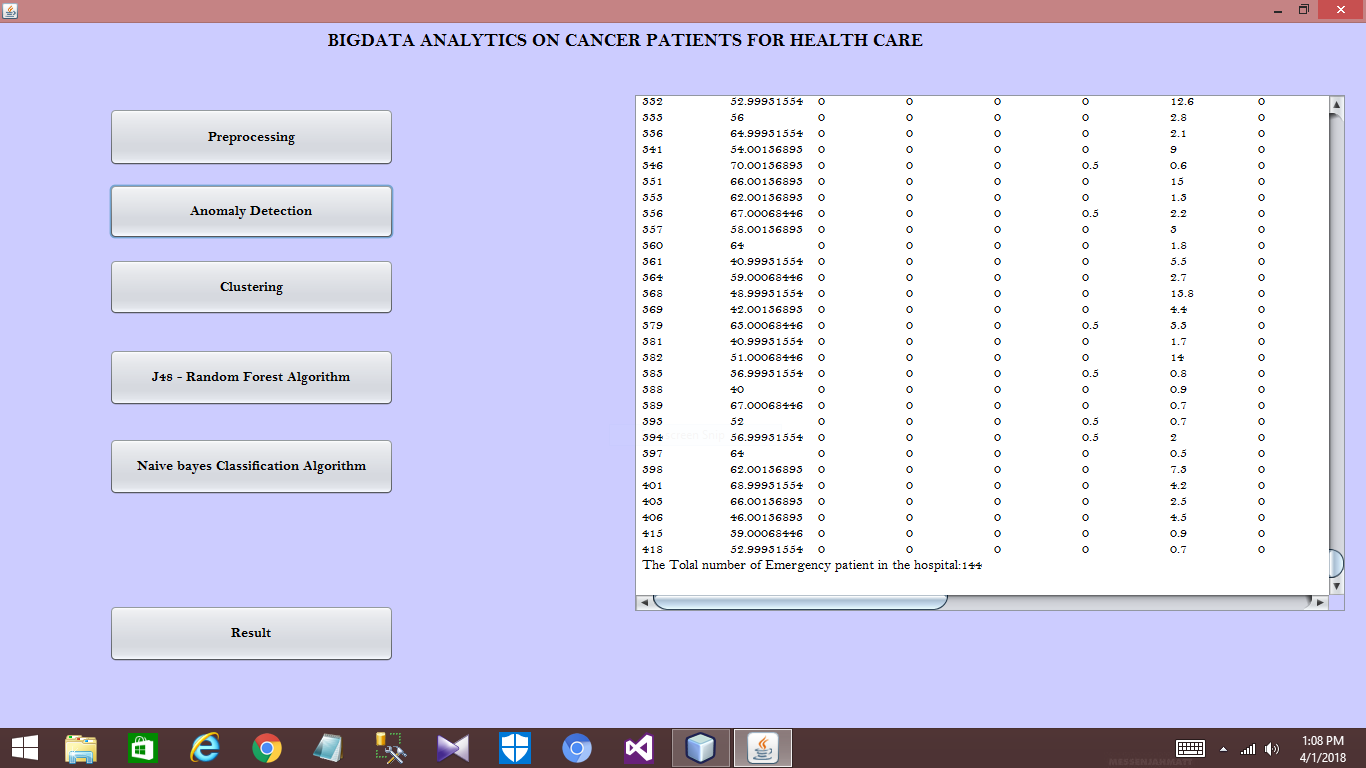
Preprocessing

****

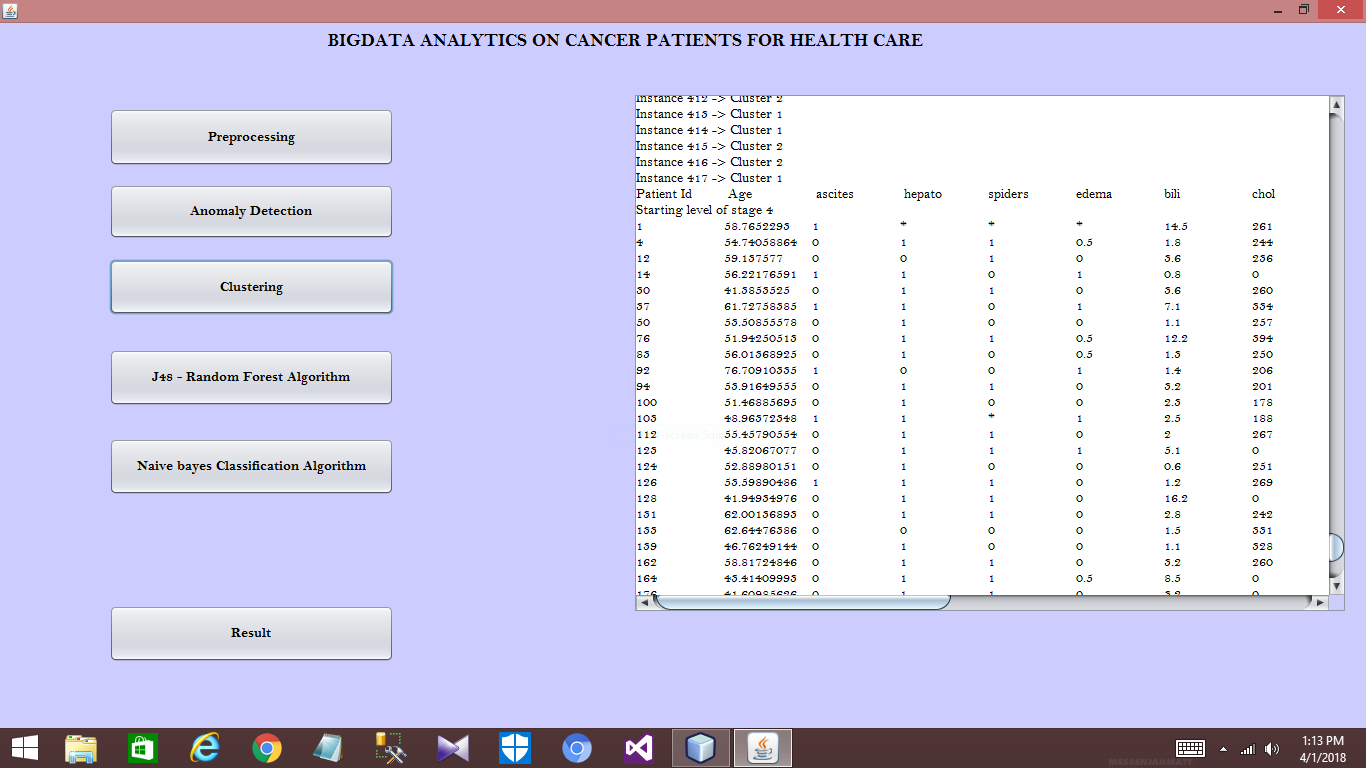
Feature extraction



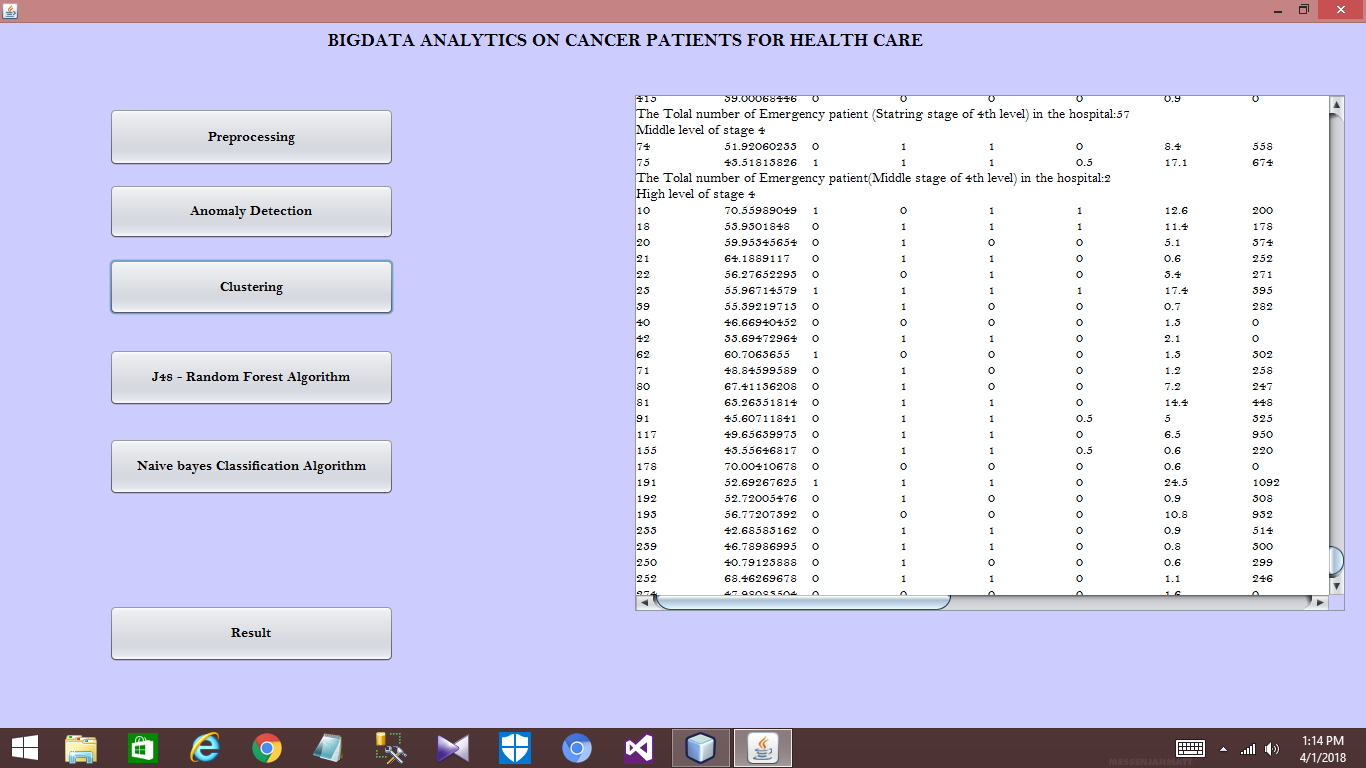
Feature extraction



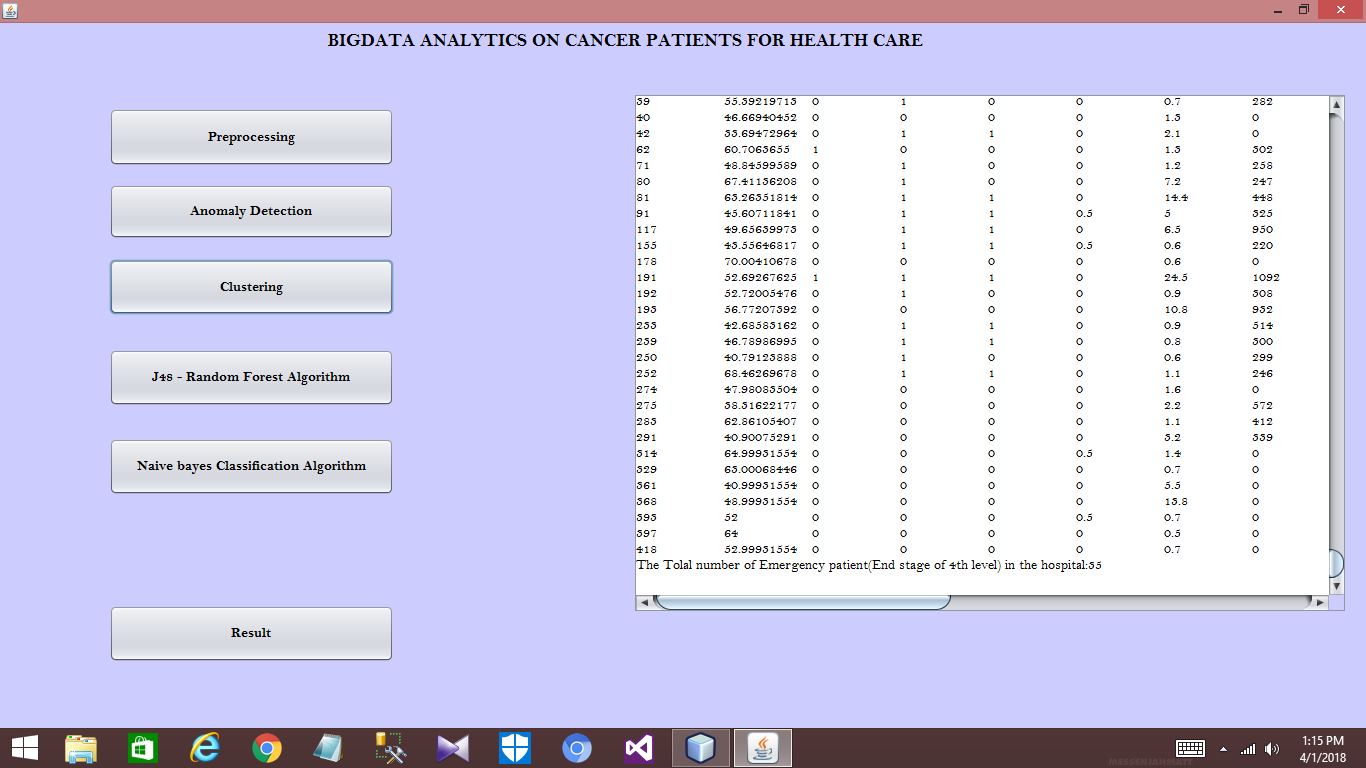
Clustering



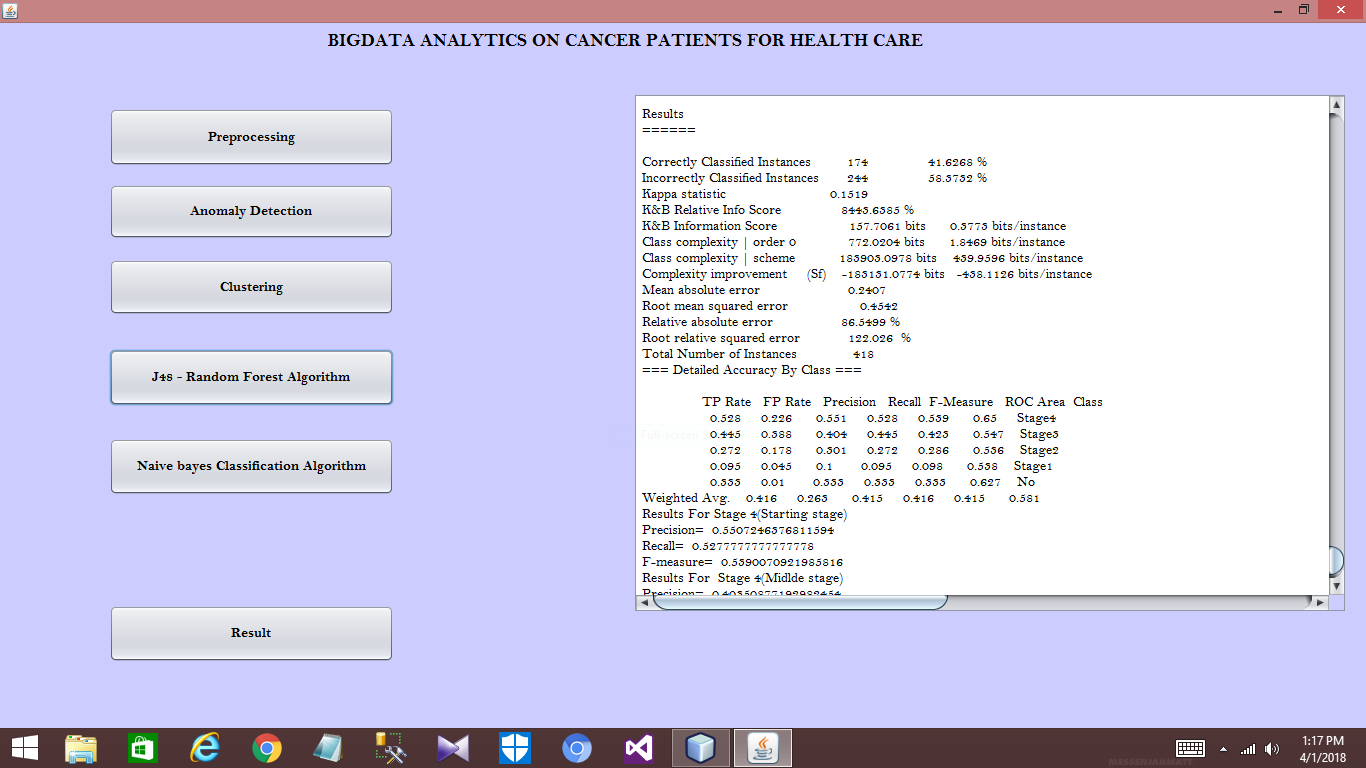
Clustering



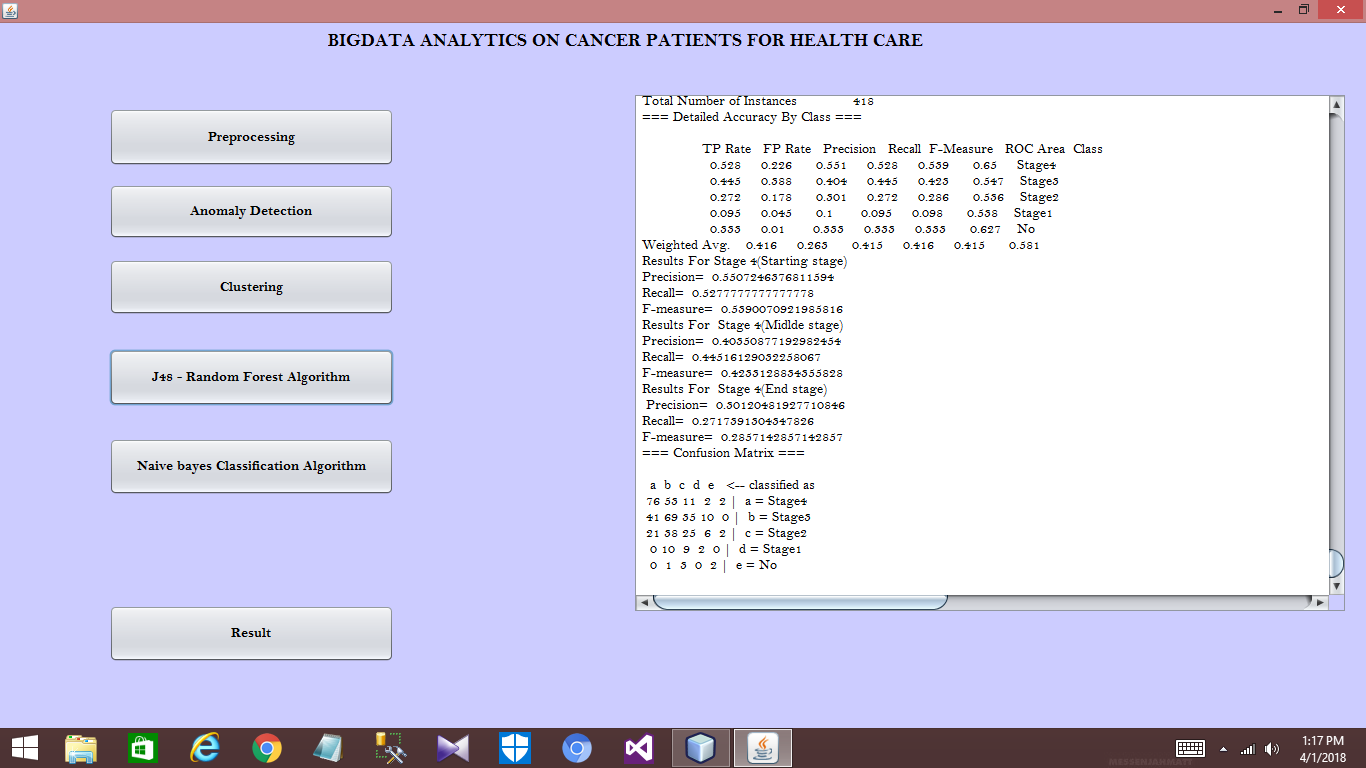
Clustering



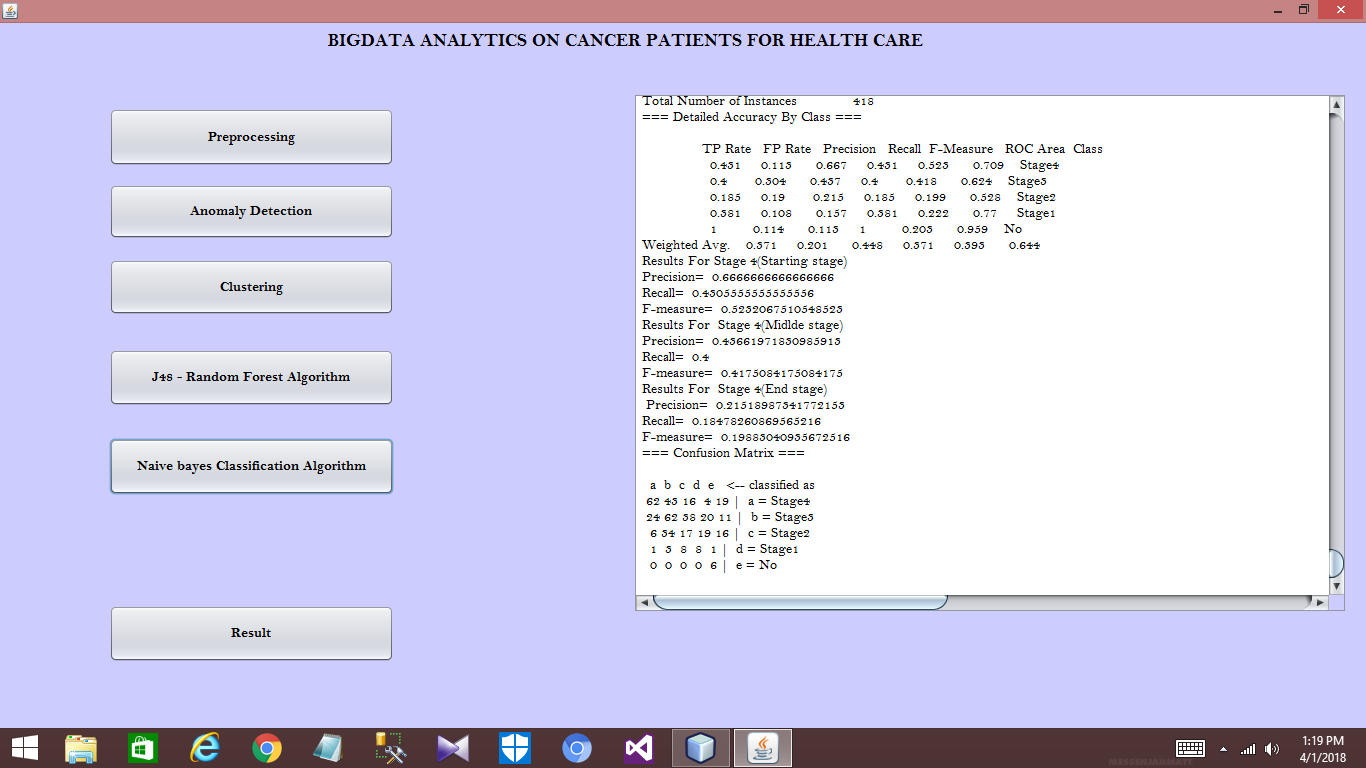
J48 Random forest classification



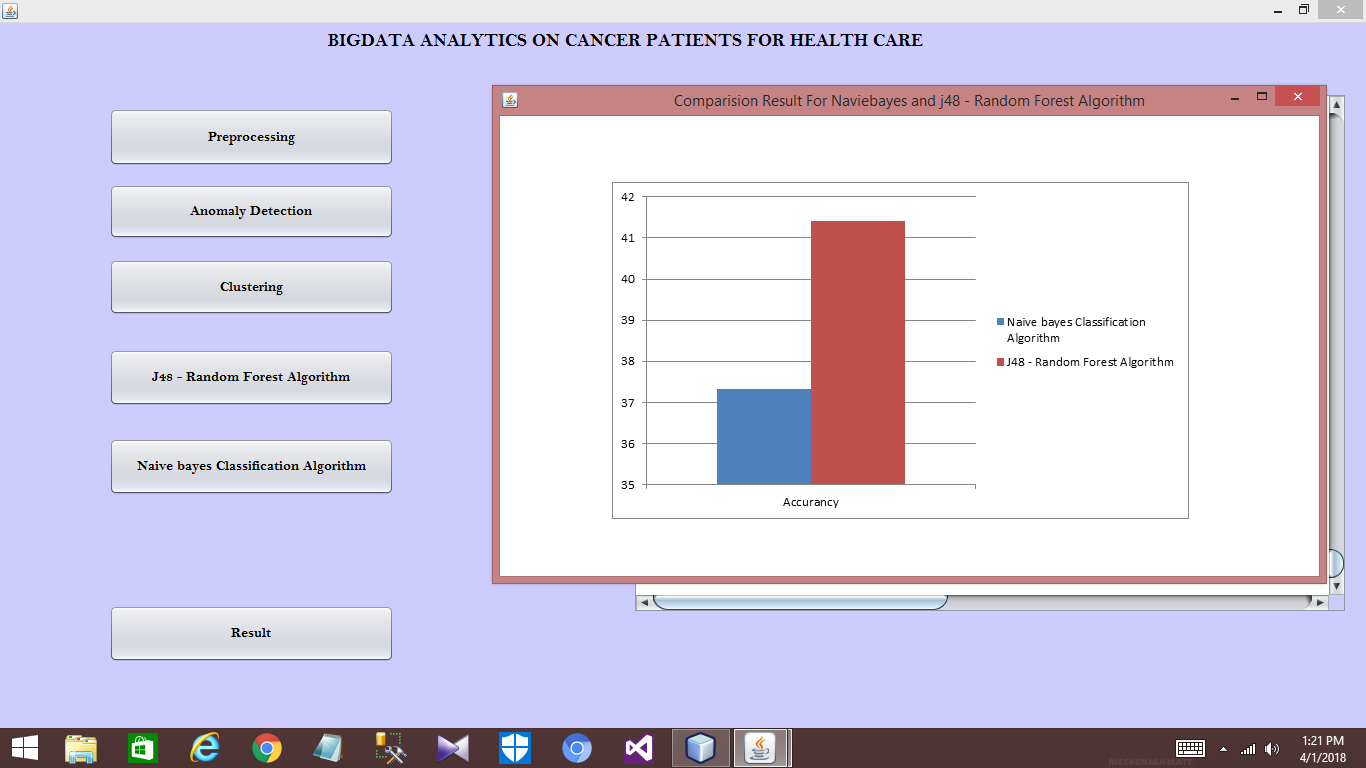
Confusion matrix for J48 classification



Confusion matrix for Naïve Bayes classification



Result comparison



**PROGRAM CODING:**

package datamining;

/\*\*

\*

\* @author ISI 2

\*/

import java.io.BufferedReader;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

import java.lang.Object;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.awt.BorderLayout;

import java.awt.image.BufferedImage;

import java.io.\*;

import java.lang.String;

import java.util.ArrayList;

import java.util.Random;

import javax.imageio.ImageIO;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.FileSystem;

import org.apache.hadoop.fs.Path;

import java.util.Random;

import weka.classifiers.Evaluation;

import weka.classifiers.bayes.NaiveBayes;

import weka.classifiers.bayes.NaiveBayesUpdateable;

import weka.classifiers.rules.OneR;

import weka.classifiers.trees.DecisionStump;

import weka.classifiers.trees.J48;

import weka.classifiers.trees.RandomForest;

import weka.core.Instances;

import weka.gui.treevisualizer.PlaceNode2;

import weka.gui.treevisualizer.TreeVisualizer;

import weka.clusterers.SimpleKMeans;

import weka.core.Instances;

public class main extends javax.swing.JFrame {

public main() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jButton4 = new javax.swing.JButton();

jScrollPane1 = new javax.swing.JScrollPane();

jTextArea1 = new javax.swing.JTextArea();

jLabel1 = new javax.swing.JLabel();

jButton5 = new javax.swing.JButton();

jButton6 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

jPanel1.setBackground(new java.awt.Color(204, 204, 255));

jButton1.setFont(new java.awt.Font("Bell MT", 1, 14)); // NOI18N

jButton1.setText("Preprocessing");

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setFont(new java.awt.Font("Bell MT", 1, 14)); // NOI18N

jButton2.setLabel("Anomaly Detection");

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

jButton3.setFont(new java.awt.Font("Bell MT", 1, 14)); // NOI18N

jButton3.setLabel("Clustering");

jButton3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton3ActionPerformed(evt);

}

});

jButton4.setFont(new java.awt.Font("Bell MT", 1, 14)); // NOI18N

jButton4.setText("J48 - Random Forest Algorithm");

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

jTextArea1.setColumns(20);

jTextArea1.setFont(new java.awt.Font("Bell MT", 0, 14)); // NOI18N

jTextArea1.setRows(5);

jScrollPane1.setViewportView(jTextArea1);

jLabel1.setFont(new java.awt.Font("Bell MT", 1, 18)); // NOI18N

jLabel1.setText("BIGDATA ANALYTICS ON CANCER PATIENTS FOR HEALTH CARE");

jButton5.setFont(new java.awt.Font("Bell MT", 1, 14)); // NOI18N

jButton5.setText("Result");

jButton5.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton5ActionPerformed(evt);

}

});

jButton6.setFont(new java.awt.Font("Bell MT", 1, 14)); // NOI18N

jButton6.setText("Naive bayes Classification Algorithm");

jButton6.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton6ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(109, 109, 109)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jButton2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton3, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton4, javax.swing.GroupLayout.PREFERRED\_SIZE, 285, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(jButton5, javax.swing.GroupLayout.PREFERRED\_SIZE, 285, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton6, javax.swing.GroupLayout.PREFERRED\_SIZE, 285, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

.addComponent(jScrollPane1)

.addGap(19, 19, 19))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(328, 328, 328)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 604, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(248, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 22, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(57, 57, 57)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 58, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(18, 18, 18)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 55, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(20, 20, 20)

.addComponent(jButton3, javax.swing.GroupLayout.PREFERRED\_SIZE, 56, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(34, 34, 34)

.addComponent(jButton4, javax.swing.GroupLayout.PREFERRED\_SIZE, 57, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(32, 32, 32)

.addComponent(jButton6, javax.swing.GroupLayout.PREFERRED\_SIZE, 57, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 34, Short.MAX\_VALUE)

.addComponent(jButton5, javax.swing.GroupLayout.PREFERRED\_SIZE, 57, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(66, 66, 66))

.addGroup(jPanel1Layout.createSequentialGroup()

.addGap(42, 42, 42)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 520, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))))

);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

pack();

}// </editor-fold>

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

try{

Path pt=new Path("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.csv");//Location of file in HDFS

FileSystem fs = FileSystem.get(new Configuration());

BufferedReader br=new BufferedReader(new InputStreamReader(fs.open(pt)));

String line;

line=br.readLine();

String line1;

line1=br.readLine();

jTextArea1.append("Medical Data \n");

while (line != null){

//int c=1;

String[] title=line.split(",");

//System.out.println(title[1]);

String[] ts=title[2].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for(String imp : ts){

String dd=line;

String dd2;

ArrayList<Double> List = new ArrayList<Double>();

for(int i=0;i<21;i++)

if( title[i] == "\*")

{

String stage=title[20].toString();

while (line1 != null){

String[] title1=line.split(",");

//System.out.println(title[1]);

String[] ts1=title1[20].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for(String imp1 : ts1){

if(imp1.equalsIgnoreCase(stage))

{

if(!title[i].isEmpty()||title[i] != "\*")

{

Double ji= Double.parseDouble(title[i]);

List.add(ji);

}

}

line1=br.readLine();

}

}

Double dd1=missingreplace.mode(List);

title[i]=String.valueOf(dd1);

} jTextArea1.append(title[0] +"\t"+title[2] +"\t"+title[3]+"\t"+title[4]+"\t"+title[5]+"\t"+title[6]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

System.out.println(title[0]+"\t"+title[2] +"\t"+title[3]+"\t"+title[4]+"\t"+title[5]+"\t"+title[6]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

//c=0;

//}

}

line=br.readLine();

}

}catch(IOException e){

System.out.println(e);

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String s = "Stage4";

int c = 0;

int count = 0;

int count1 = 0;

String[] key = s.split(" ");

ArrayList<String> alt = new ArrayList<String>();

jTextArea1.append("Patient Id \t Age \t ascites\t hepato\tspiders\tedema\tbili\tchol\talbumin\tcopper\talk.phos\tast\ttrig\tplatelet\tprotime\tstage \n");

for (String s1 : key) {

try {

BufferedReader br = new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.csv"));

String line;

if (c == 0) {

c = 1;

line = br.readLine();

while (line != null) {

//int c=1;

String[] title = line.split(",");

//System.out.println(title[1]);

String[] ts = title[20].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for (String imp : ts) {

//if(c==1){

if (s1.equalsIgnoreCase(imp)) {

jTextArea1.append(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

System.out.println(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

cOunt = count + 1;

alt.add(line);

}

line = br.readLine();

//}

}

}

System.out.println("The Number of Emergency patient in the hospital:" + count +"\n");

jTextArea1.append("The Tolal number of Emergency patient in the hospital:" + count +"\n");

} else {

line = br.readLine();

while (line != null) {

//int c=1;

String[] title = line.split(",");

//System.out.println(title[1]);

String[] ts = title[32].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for (String imp : ts) {

//if(c==1){

if (s1.equalsIgnoreCase(imp)) {

for (String gg : alt) {

if (!gg.equalsIgnoreCase(line)) {

jTextArea1.append(title[0] + "\t" + title[2] + "\t" + title[3] + "\t" + title[4] + "\n");

//System.out.println(title[0]+"\t"+title[1] +"\t"+title[2]);

//c=0;

break;

}

}

}

//}

}

line = br.readLine();

}

}

} catch (Exception e) {

System.out.println(e);

}

}

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

try {

SimpleKMeans kmeans = new SimpleKMeans();

kmeans.setSeed(10);

//important parameter to set: preserver order, number of cluster.

kmeans.setPreserveInstancesOrder(true);

kmeans.setNumClusters(3);

BufferedReader br1 = null;

br1 = new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.arff"));

Instances trainData = new Instances(br1);

kmeans.buildClusterer(trainData);

// This array returns the cluster number (starting with 0) for each instance

// The array has as many elements as the number of instances

int[] assignments = kmeans.getAssignments();

int kl=0;

for(int clusterNum : assignments) {

System.out.printf("Instance %d -> Cluster %d \n", kl, clusterNum);

String dd=String.valueOf(kl);

String gg= String.valueOf(clusterNum);

jTextArea1.append("Instance " +dd+" -> Cluster "+gg+"\n");

kl++;

}

// TODO add your handling code here:

String s = "Stage4";

int c = 0;

int count = 0;

int count1 = 0;

String[] key = s.split(" ");

ArrayList<String> alt = new ArrayList<String>();

jTextArea1.append("Patient Id \t Age \t ascites\t hepato\tspiders\tedema\tbili\tchol\talbumin\tcopper\talk.phos\tast\ttrig\tplatelet\tprotime\tstage \n");

jTextArea1.append("Starting level of stage 4 \n");

for (String s1 : key) {

try {

BufferedReader br = new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.csv"));

String line;

if (c == 0) {

c = 1;

line = br.readLine();

while (line != null) {

//int c=1;

String[] title = line.split(",");

//System.out.println(title[1]);

String[] ts = title[20].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for (String imp : ts) {//if(c==1){

if (s1.equalsIgnoreCase(imp)) {

int i= Integer.valueOf(title[12]);

int j= Integer.valueOf(title[14]);

int k= Integer.valueOf(title[18]);

if(i<500 && j<200 && k<200 )

{

jTextArea1.append(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

System.out.println(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

count = count + 1;

alt.add(line);

}

}

line = br.readLine();

//}

}

}

System.out.println("The Number of Emergency patient(Statring stage of 4th level) in the hospital:" + count +"\n");

jTextArea1.append("The Tolal number of Emergency patient (Statring stage of 4th level) in the hospital:" + count +"\n");

}

}catch (Exception e) {

System.out.println(e);

}

}

count=0;

c = 0;

jTextArea1.append("Middle level of stage 4 \n");

for (String s1 : key) {

try {

BufferedReader br = new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.csv"));

String line;

if (c == 0) {

c = 1;

line = br.readLine();

while (line != null) {

//int c=1;

String[] title = line.split(",");

//System.out.println(title[1]);

String[] ts = title[20].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for (String imp : ts) {

//if(c==1){

if (s1.equalsIgnoreCase(imp)) {

int i= Integer.valueOf(title[12]);

int j= Integer.valueOf(title[14]);

int k= Integer.valueOf(title[18]);

if(i>500 && i<1000 && j>200 && j<400 && k>200 && k<300 )

{

jTextArea1.append(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

System.out.println(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

count = count + 1;

alt.add(line);

}

}

line = br.readLine();

//}

}

}

System.out.println("The Number of Emergency patient(Middle stage of 4th level) in the hospital:" + count +"\n" +"\n");

jTextArea1.append("The Tolal number of Emergency patient(Middle stage of 4th level) in the hospital:" + count +"\n");

}

}catch (Exception e) {

System.out.println(e);

}

}

count=0;

c = 0;jTextArea1.append("High level of stage 4 \n");

for (String s1 : key) {

try {

BufferedReader br = new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.csv"));

String line;

if (c == 0) {

c = 1;

line = br.readLine();

while (line != null) {

//int c=1;

String[] title = line.split(",");

//System.out.println(title[1]);

String[] ts = title[20].split(" ");

//ts=ts.replaceAll("\\s","");

//if(s.matches("(.\*)"+ts+"(.\*)")){

for (String imp : ts) {

//if(c==1){

if (s1.equalsIgnoreCase(imp)) {

int i= Integer.valueOf(title[12]);

int j= Integer.valueOf(title[14]);

int k= Integer.valueOf(title[18]);

if(i>1000 || j>400 || k>300 )

{

jTextArea1.append(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

System.out.println(title[0] +"\t"+title[5]+"\t"+title[7]+"\t"+title[8]+"\t"+title[9]+"\t"+title[10]+"\t"+ title[11]+"\t"+title[12] +"\t"+title[13]+"\t"+title[14]+"\t"+title[15]+"\t"+title[16]+"\t"+title[17]+"\t"+title[18]+"\t"+title[19]+"\t"+title[20]+"\n");

count = count + 1;

alt.add(line);

}

}

line = br.readLine();

//}

}

}

System.out.println("The Number of Emergency patient (End stage of 4th level) in the hospital:" + count +"\n");

jTextArea1.append("The Tolal number of Emergency patient(End stage of 4th level) in the hospital:" + count +"\n");

}

}catch (Exception e) {

System.out.println(e);

}

}

} catch (Exception ex) {

Logger.getLogger(main.class.getName()).log(Level.SEVERE, null, ex);

}

}

public static BufferedReader readDataFile(String filename) {

BufferedReader inputReader = null;

try {

inputReader = new BufferedReader(new FileReader(filename));

} catch (FileNotFoundException ex) {

System.err.println("File not found: " + filename);

}

return inputReader;

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

try{

//Reading training arff or csv file

Instances data = new Instances(new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.arff")));

J48 cls = new J48();

data.setClassIndex(data.numAttributes() - 1);

cls.buildClassifier(data);

// display classifier

Evaluation evaluation = new Evaluation(data);

Random rand = new Random(10);

int folds = 2;

evaluation.crossValidateModel(cls, data, folds, rand);

System.out.println(evaluation.toSummaryString("\nResults\n======\n", true));

System.out.println(evaluation.toClassDetailsString());

System.out.println("Results For Stage 4(Starting stage ");

System.out.println("Precision= " + evaluation.precision(0));

System.out.println("Recall= " + evaluation.recall(0));

System.out.println("F-measure= " + evaluation.fMeasure(0));

System.out.println("Results For Stage 4(Midlde stage)");

System.out.println("Precision= " + evaluation.precision(1));

System.out.println("Recall= " + evaluation.recall(1));

System.out.println("F-measure= " + evaluation.fMeasure(1));

System.out.println("Results For Stage 4(End stage)");

System.out.println("Precision= " + evaluation.precision(2));

System.out.println("Recall= " + evaluation.recall(2));

System.out.println("F-measure= " + evaluation.fMeasure(2));

jTextArea1.append(evaluation.toSummaryString("\nResults\n======\n", true));

jTextArea1.append(evaluation.toClassDetailsString());

jTextArea1.append("Results For Stage 4(Starting stage) \n");

jTextArea1.append("Precision= " + evaluation.precision(0)+"\n");

jTextArea1.append("Recall= " + evaluation.recall(0)+"\n");

jTextArea1.append("F-measure= " + evaluation.fMeasure(0)+"\n");

jTextArea1.append("Results For Stage 4(Midlde stage) \n");

jTextArea1.append("Precision= " + evaluation.precision(1)+"\n");

jTextArea1.append("Recall= " + evaluation.recall(1)+"\n");

jTextArea1.append("F-measure= " + evaluation.fMeasure(1)+"\n");

jTextArea1.append("Results For Stage 4(End stage)\n ");

jTextArea1.append("Precision= " + evaluation.precision(2)+"\n");

jTextArea1.append("Recall= " + evaluation.recall(2)+"\n");

jTextArea1.append("F-measure= " + evaluation.fMeasure(2)+"\n");

jTextArea1.append(evaluation.toMatrixString());

System.out.println(evaluation.toMatrixString());

} catch (Exception ex) {

Logger.getLogger(main.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

try {

// TODO add your handling code here:

String path = "C:/Users/karthikeyan/Desktop/project/datamining/graph.png";

File file = new File(path);

BufferedImage image = ImageIO.read(file);

JLabel label = new JLabel(new ImageIcon(image));

JFrame f = new JFrame();

f.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

f.getContentPane().add(label);

f.pack();

f.setLocation(200,200);

f.setVisible(true);

f.setTitle("Comparision Result For Naviebayes and j48 - Random Forest Algorithm");

} catch (IOException ex) {

Logger.getLogger(main.class.getName()).log(Level.SEVERE, null, ex);

}

}

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {

try {

// TODO add your handling code here:

Instances data = new Instances(new BufferedReader(new FileReader("C:/Users/karthikeyan/Desktop/project/datamining/dataset/pbc.arff")));

NaiveBayesUpdateable cls = new NaiveBayesUpdateable();

data.setClassIndex(data.numAttributes()-1);

cls.buildClassifier(data);

// display classifier

Evaluation evaluation = new Evaluation(data);

Random rand = new Random(10000);

int folds = 200;

evaluation.crossValidateModel(cls, data, folds, rand);

System.out.println(evaluation.toSummaryString("\nResults\n======\n", true));

System.out.println(evaluation.toClassDetailsString());

System.out.println("Results For Stage 4(Starting stage ");

System.out.println("Precision= " + evaluation.precision(0));

System.out.println("Recall= " + evaluation.recall(0));

System.out.println("F-measure= " + evaluation.fMeasure(0));

System.out.println("Results For Stage 4(Midlde stage)");

System.out.println("Precision= " + evaluation.precision(1));

System.out.println("Recall= " + evaluation.recall(1));

System.out.println("F-measure= " + evaluation.fMeasure(1));

System.out.println("Results For Stage 4(End stage)");

System.out.println("Precision= " + evaluation.precision(2));

System.out.println("Recall= " + evaluation.recall(2));

System.out.println("F-measure= " + evaluation.fMeasure(2));

jTextArea1.append(evaluation.toSummaryString("\nResults\n======\n", true));

jTextArea1.append(evaluation.toClassDetailsString());

jTextArea1.append("Results For Stage 4(Starting stage) \n");

jTextArea1.append("Precision= " + evaluation.precision(0)+"\n");

jTextArea1.append("Recall= " + evaluation.recall(0)+"\n");

jTextArea1.append("F-measure= " + evaluation.fMeasure(0)+"\n");

jTextArea1.append("Results For Stage 4(Midlde stage) \n");

jTextArea1.append("Precision= " + evaluation.precision(1)+"\n");

jTextArea1.append("Recall= " + evaluation.recall(1)+"\n");

jTextArea1.append("F-measure= " + evaluation.fMeasure(1)+"\n");

jTextArea1.append("Results For Stage 4(End stage)\n ");

jTextArea1.append("Precision= " + evaluation.precision(2)+"\n");

jTextArea1.append("Recall= " + evaluation.recall(2)+"\n");

jTextArea1.append("F-measure= " + evaluation.fMeasure(2)+"\n");

jTextArea1.append(evaluation.toMatrixString());

System.out.println(evaluation.toMatrixString());

} catch (Exception ex) {

Logger.getLogger(main.class.getName()).log(Level.SEVERE, null, ex);

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\*

\* Set the Nimbus look and feel

\*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\*

\* If Nimbus (introduced in Java SE 6) is not available, stay with the

\* default look and feel. For details see

\* http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger (main.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\*

\* Create and display the form

\*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new main().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton4;

private javax.swing.JButton jButton5;

private javax.swing.JButton jButton6;

private javax.swing.JLabel jLabel1;

private javax.swing.JPanel jPanel1;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTextArea jTextArea1;

// End of variables declaration

}