ISSN (Online) : 2455 - 3662
SJIF Impact Factor :3.967

EPRA International Journal of Multidisciplinary Research

Monthly Peer Reviewed & Indexed
International Online Journal

Volume: 3 Issue: 10 October 2017

Published By :
EPRA Journals

SJIF Impact Factor: 3.967 Volume: 3 | Issue: 10 |October 2017

EPRA International Journal of Multidisciplinary Research (IJMR) ISSN (Online): 2455-3662

COST-SENSITIVE LEARNING: A HYBRID OF TREE INDUCTION PRISM ON MEDICAL DATA

Samuel Adiele–Osmond[1]
[1]School of Systems Engineering,
The University of Reading,
Reading, Berkshire, UK

Dr. Frederic Stahl[2]
[2]School of Systems Engineering,
The University of Reading,
Reading, Berkshire, UK

ABSTRACT

Classification of generated rules from medical datasets is important research problem. However, generating a cost-sensitive rule is a more critical issue that could help in cost-efficient aided diagnosis and treatment. Previous research based on decision tree induc-tion have been carried out to classify cost sensitive rules however, the cost of the root node seems to be the loophole with these algorithms. In this research, a Cost-Sensitive Prism (CS-PRISM) algorithm is proposed to generate classified rules sensitive to cost of test (attributes), and misclassification error. Most importantly, the objective of the CS-PRISM is to builds rules that eliminates of the redundancy of the root node's cost. This is carried out by incorporating the test cost technique of tree induction with the Prism algorithm, initiating a function, Probability Cost Function (PCF) which is the basis for classifying rule.

1 INTRODUCTION

It is arguably certain that prior to current times, biomedical analysis were theoretically anecdotal, though with varying degrees of high accuracy [32]. The inability to find per-manent cures for virulent diseases such as Cancer, Anaemia, Down syndrome, Cystic Fibrosis etc has challenged technological advances in clinical and biological sciences. In an era of vast amount of data, digital growth has encouraged the collection, storage and organisation of diseased patients data. It is believed that trends of knowledge or relationships could be extracted from these data. Nevertheless, if patterns could be identified within medical data, certain chronic conditions could be managed efficiently with early diagnosis and best treatment procedures[25][50].

Limitations to human cognition and perception has propelled the application of De-cision Support Systems (DSS) models to effectively identify useful trends from large datasets. Data mining is the acronym for knowledge extraction applied in this ex-periment. The goal of this research is to introduce a new cost efficient algorithm for generating rules known as Cost Sensitive Prism (CS-PRISM). The proposed model is an evolution of the standard prism [9] algorithm, incorporated with a tree induction. CS-PRISM fitness function is the average cost of classification. This includes the cost of test(attribute features) as well as the cost of misclassification errors. The new algo-rithm when applied on a medical dataset has the ability to analyse medical records and then generate sets of rules with strong considerations to test cost of attributes and error of misclassification.

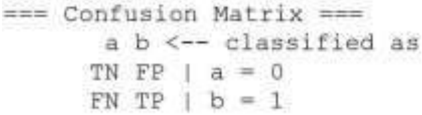
Assuming machine learning techniques is applied in a medical center to induce a diagnostic tool from patients records. It would be logically expected to obtain an extensive model with minimal low test cost and obviously of high accuracy. Taking into consideration that here are many cases associated with predictive errors, the task of the miner is to produce a model with minimum expected test cost and misclassification cost.

1.1Cost Concept Learning

Cost can be measured in different units such as monetary units (dollars), temporal units (seconds), or abstract units of utility (utils)[56]. In medical diagnosis, cost may include things, such as the time or money for medical test to be carried or even the quality of life of the patient. In image recognition, cost might be measured in terms of the CPU time required for certain computations. However, cost could appear uncertain and in some cases, the uncertainty can be represented with a probability distribution over a range of possible costs[55]. This applies to misclassification errors and cost of test.

1.1.1Cost of Misclassification Errors

In a confusion matrix with P classes, we may P x P matrix, where the element in row m and column n specifies the cost of assigning a case to class m , when it actually belongs in class n. Typically undoubtedly, the cost is zero when m equals n. In a minor variation on this approach, we may have a rectangular matrix, where there is an extra row for the cost of assigning a case to the unknown class[55].



```
=== Confusion Matrix ===
      a b <-- classified as
     TN FP | a = 0
     FN TP | b = 1
```

Figure 1: Confusion Matrix structure[21]

Misclassification cost could be constant by having same cost value for all cases.If the cost is zero m equals n and one otherwise, then our cost measure is the familiar error-rate measure. If the cost is one if m equals n and zero otherwise, then our cost measure is the familiar accuracy measure [56]

However, the cost of certain types of misclassification error may be provisional to some circumstances. Error cost may depend on nature of the particular cases. For ex-ample, in detection of fraud, the cost of missing a particular case of fraud will depend on the amount of money involved in that particular case [16][17]. Similarly, the cost of a certain kind of mistaken medical diagnosis may be conditional on the particular patient who is misdiagnosed. For example, the misdiagnosis may be more costly in el-derly patients. It may be possible to represent such scenario with a constant error cost by distinguishing sub-classes. For example, instead of two classes, sick and healthy, there could be three classes, sick-and-young, sick-and elderly, and healthy[56]. This is an imperfect solution when the cost varies continuously, rather than discretely

In medical dialysis, the cost of a classification error may also be dependant on the timing. Consider a medical device intended to signal an alarm during surgery if there are complications or might occur. With the sensor readings classified as either alarm or noalarm, the cost of the classification depends on whether the classification is correct as well as the timeliness of the classification. The alarm is not useful unless there is sufficient time for an adequate response to the alarm [16][17].

In certain circumstances, the cost of making a classification error with one case may depend on the errors made with other cases. The well-known Precision and Re-call measures, widely used in the information retrieval literature, may be seen as cost measures of this type [57]. For example, consider an information retrieval task where we are to search for a document on a certain topic. Collections of the documents are be to classified as relevant or not-relevant for the given topic. The cost of mistakenly assigning a relevant document to the not-relevant class depends on whether there are any other relevant documents that we have correctly classified[56].

1.1.2 Cost of Tests

Each test, also referred to as attribute or measurement or feature, may have an associ-ated cost. For example, in medical diagnosis, a blood test has a cost. Turney [55] points out that we can only rationally determine whether it is worthwhile to pay the cost of a test when we know the cost of misclassification errors. If the cost of misclassification errors is much greater than the cost of tests, then it is rational to purchase all tests that seem to have some predictive value. If the cost of misclassification errors is much less than the cost of tests, then it is not rational to purchase any tests.

Tasks involving both misclassification and attribute costs are abundant in real-world applications. In medical diagnosis, medical tests are referred to as attributes in machine learning whose values may be obtained at a cost, and misdiagnoses are like misclassi-fication which may also bear a cost (misclassification cost). When building a learning model for medical diagnosis from the training data, we must consider both the attribute costs (medical tests such as blood tests) and misclassification costs (errors in the diag-nosis).

Each test has a different cost but the cost of a given test is the same for all cases[42][52]. Contrary, the cost to perform certain test may vary with the circumstances surround-ing the test. Such conditional circumstances could be selection of prior test. A given patient's test cost could be tentative on the previous test that have been selected for the patient. For instance, a group of urine test ordered together may be cheaper than the sum of the costs of each test considered by itself, since the tests share common costs, such as the cost of collecting urine from the patient[55]. Furthermore, the cost of performing certain test may be conditional on the results of previous tests of a pa-tients. For example, the cost of a blood test may be conditional on the patient's age. Thus a blood test must be preceded by a patient-age test, which contributes to the cost of the blood test. Also possible side-effects of a particular medical test could affect the cost of performing certain test on a given patient. A patient who is allergic to dye used for radiological procedures, could have their test cost triggered due allergic complications.[56]. Furthermore, when additional medical tests are ordered, at a cost of a patient or an insurance company, diagnosis or prediction of a disease of the patient is improved, reducing the misclassification cost.

1.2 Motivation

The issue of cost-sensitive learning has been a constant problem not only in med-ical diagnosis[42] but also in robotics[53][52], industrial production processes[58], communication network troubleshooting [1], machinery diagnosis (where main cost is skilled labour), automated testing of electronic equipment (where the main cost is time), and many other areas. Learning models from the past have considered the cost of test, they include EG2 by Nunez [42][43], IDX by Norton[41] and CS-ID3 by Tan & Schlimmer [53][52]. There are also other learners that consider misclassification cost [6][19][27][23][45][47]. However CS-PRISM considers both the cost of test and misclassification cost.

There are favourable reasons to support the coalition of both costs. An expert can-not logically determine the cost of test without knowing the cost of correct or incorrect classification. It is also expected for an expert to balance the cost of each test with the contribution of the test to accurate classification. Experts must also consider when further testing is not economically justified as it often happens that the benefits of further testing are not worth the costs of the tests. This proofs that a cost value must be as-signed to both the tests and the classification errors[55]. Another constraint with many existing cost-sensitive model such as EG2[42][53][52] etc, is that they apply greedy heuristic search to select whichever step contributes more to accuracy and least to cost. A more sophisticated model such as (Inexpensive Classification with Expensive Tests) ICET[55] and Anytime Cost-sensitive Tree learner (ACT) were initiated to combine the greedy heuristic search with other algorithms to evaluate the interaction among sequences of test. A test may appear useful in isolation, using the greedy heuristic or may appear not useful when considered in association with other test. However, one major flaw with this approach and other tree learning models is the cost of the root node in cost effective learning. The root node of any tree model [6][49] is inevitable in rules generation and class prediction. However, the cost of this node poses a threat of being recycled in every rules classified thereby, increasing the average cost of clas-sification. CS-PRISM model eliminates this flaw by producing a more compact sets of rules which are cost-sensitive. However, the objectives of this research are to

1. Develop cost effective model that generate rules sensitive to test cost,
2. Sensitive to cost of misclassification error and .
3. Eliminate the redundancy cost of the root node in tree learning.

Section 2 discusses knowledge discovery from medical dataset, previous cost-sensitive algorithm applied and their application in medicine and healthcare. Section 3 would describe the dataset applied, the CS-PRISM technique, experimental methodologies, evaluation measures and validation design. The final section would highlight the set-backs of the algorithm, future work plans and a proposed work flow.

2. LITERATURE REVIEW

2.1 Average cost of classification

Given a dataset of training and testing set. The typical cost of classification is estimated by the average cost of classification for the testing set. The average cost of classifica-tion is computed by dividing the total cost for the whole testing set by the number of cases in the testing set. The total cost includes both the costs of tests and the costs of classification errors[55].

$$\text{TotalCost} = \text{TestCost} + \text{Misclassi f icationCost}$$

$$\text{AverageCost} = \frac{\text{TotalCost}}{\text{Numbero f TestingSetSample}}$$

Lets consider a simple experiment where we can specify test costs simply by listing each test, paired with its corresponding cost and that we can specify the costs of clas-sification errors using a classification cost matrix p * p. The element P of class mandn is the cost of guessing that a case belongs in class m, when it actually belongs in class n. However, in such scenario, we have restricted our attention to classification cost matrices in which the diagonal elements are zero (we assume that correct classification has no cost) and the off-diagonal elements are positive numbers. For a cost tree model, to calculate the cost of a particular case, the path down the tree is traced and the cost of each test chosen is added up. However, If the same test appears twice, we only charge for the first occurrence of the test[55]. For example, one node in a path may say patient age is less than 20 years and another node may say patient age is more than 10 years, but we only charge once for the cost of determining the patients age.

The decision to do a test must be based on both the cost of tests and the cost of classification errors. If a test costs $10 and the maximum penalty for a classification error is $5, then there is clearly no point in doing the test. Contrary, if the penalty for a classification error is $10,000, the test may be quite worthwhile, even if its information content is relatively low. Previous study with algorithms that are sensitive to test costs [42][43][41] has overlooked the importance of also considering the cost of classifica-tion errors

Although, when test cost are reasonable compared to cost of misclassification, it may be logical to do all test that seems important. In such situation, it is easier to separate the selection of tests from the process of learning. Firstly, choose the set of relevant tests and then focus on the problem of learning a case using the results of these tests. On the other hand, if test costs are expensive when compared to the cost of clas-sification errors, it may be irrational to separate the selection of tests from the process of learning[55]. In this situation, a much lower costs can be attained by interleaving the two. First, choose a test, then examine the test result. The result of the test might give information to influence the choice for the next test. In a more rigid (expensive test) situations, the cost of further tests may not be justified, therefore stopping testing to make a classification.

2.2 Knowledge Discovery of Medical Databases

Knowledge Discovery from medical Databases (KDD) is a Decision Support System (DSS) that encompasses several fields which includes, pattern recognition, statistics, machine learning, database management and visualisation tools to support the analysis and discovery of symmetries that are hidden within data [31].

The diagram below describes the complete KDD processes from converting raw medi-cal data into useful knowledge [26].

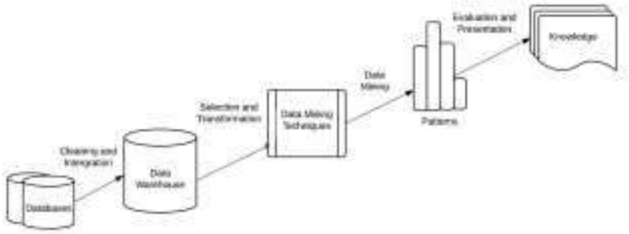


Figure 2: The KDD Processes

1. Data cleaning and Integration: Experiments done by multiple independent components are bound to experience inconsistencies in data representation[61]. Data Cleaning is applied to eliminate inconsistencies and noisy data, while integration com-prises of merging or joining multiple data sources into one data store.
2. Data selection and Transformation: Medical database maintain multiples val-ues recorded at different format which could be inapplicable to standards of machine learning models [28]. Data selection entails retrieving most relevant data from the database for the analysis task. Transformation is done to convert data which most likely appear in different formats into appropriate standard for the mining process
3. Data mining: This involves applying intelligent methods such as Support Vec-tor Machine(SVM), Artificial Neural Network (ANN), Nearest Neighbour (KNN)etc. to the pre-processed data to extract useful patterns or knowledge.
4. Knowledge Evaluation and presentation: Knowledge evaluation involves identifying the interesting facet of the mined pattern and presenting the mined knowl-edge in an understandable and meaningful way.

2.2.1 Data Mining Techniques for Knowledge Discovery

As seen in the previous section, data mining being one of steps of the KDD process has an objective to identify useful information from databases[2] and converting them into understandable form for further use. This task is divided into two major categories:

Predictive Learning and Descriptive Learning This task involves classifying un-seen data based on model or knowledge from a similar dataset[44]. In Medical science, predictive learning is used to generate cost effective rules that could learn certain char-acteristics associated with related diseases[42][29][33][35]. However, the goal of this task is to investigate models that curtail the error of predictions as well as target vari-ables to be predicted [44].

The objective of descriptive learning is to discover hidden trends of knowledge within large datasets. This approach is potentially applicable to identify new patterns of knowledge relating to certain diseases. Models such as ICET[55] have been de-signed to generate sets of medical rules with minimal test cost and classification cost. Additionally, the ACT[14] is aimed at generating rules by trading computational time for lower classification cost. EG2, [42] is known widely for implementing an Informa-tion Cost Function (ICF) to select attributes for rule generation based on their cost and information gain.

2.3 Supervised Learning Methods

Supervised learning is concerned with the construction of learning systems that, when previously trained, can assign the suitable classes among sets of possible variables [20].Supervised classification analysis have two major objectives: To accurately pre-dict class for new variables and to extract patterns from past trainings sessions. There are a number of supervised techniques, but in this section would only focus on the cost sensitive techniques.

2.4 Decision Tree Induction:

Induction of decision trees generates predictive classifier based on a tree structure as shown in Figure 3. The tree consist of a root node, internal nodes and leaf or terminals nodes. The leaf nodes are usually given as the target class attribute also known as the predictive class attributes [39]. C4.5 [49] builds a decision tree using the top down induction of decision tree (TDIDT) approach to partition the data into smaller subsets. From the root node to every leaf node, there is a path which consist of different multiple internal nodes attributes which generates rules for classifying unknown data [62].

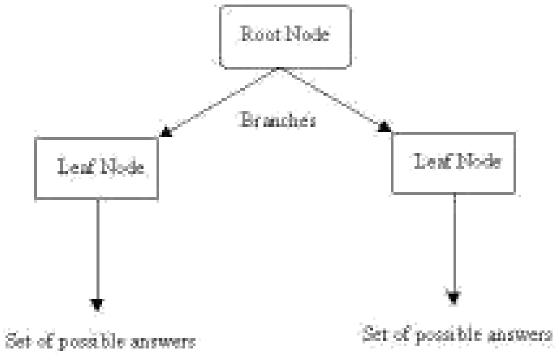


Figure 3: The Decision Tree

At each step of the decision tree construction, C4.5 selects the attributes with high-est information gain ratio and divides the training set into classes of attributes until all records associated with a node is assigned to one class. The Induction of tree is consid-ered to be fast in building and generating decision rules. The rules generated are easy to interpret, which has made it one of most the applied classifiers [13] as well as very pow-erful algorithms in medicine [12]. In fact, lots of the cost sensitive algorithm applied in medical science have been based on decision tree induction[55][14][29][33][35]. This section highlights some of cost-sensitive tree induction.

2.4.1 EG2

This is TDIDT algorithm that implements a mathematical function known as Informa-tion Cost Function (ICF) for selection of attributes[42]. ICF selects attributes based on two factors, their information gain and attributes cost. This technique was achieved by manipulating the C4.5 source code and replacing the information gain ratio with ICF. The ICF of an x-th attribute is represented as

$$2^{DI_x}$$

$$ICF_X = \frac{1}{(C_X + 1)^w}$$

In this formulae, DI_x is the information gain associated with the x-th attribute at a given stage in the construction of the decison tree and C_x is the cost of measuring the x-th attribute. It is also assumed that 0<w<. C4.5 selects attributes that maximises in-formation gain ratio DI_x while the EG2 is a modified version that selects the attributes that maximises ICF_x.

The parameter w adjusts the strength of the bias towards lower cost attributes. If w = 0, cost is ignored and selection ICF_x is equivalent to selection by DI_x. However, when w = 1, ICF_x is fully biased on cost. In the cost sensitive environment, w is set as 1, therefore eliminating w from the equation.

2.4.2 CS-ID3

This is another TDIDT approach that selects attributes that maximises a cost heuristic equation. CS-ID3 technique is implemented by modifying the C4.5 source code so that it selects attribute that maximizes the CS-ID3 function. The cost of measuring x-th attribute is represented as

$$CS \quad ID3_X = \frac{DI_{X2}}{C_X}$$

CS-ID3 uses a lazy evaluation approach by only constructing the part of the deci-sion tree that classifies the current case[53][52].

2.4.3 IDX

This one of the oldest cost sensitive TDIDT technique. IDX [41] selects sets of attribute that maximizes the IDX heuristic function. Likewise the two previous model, the IDX was implementing by manupulating the C4.5 so that attributes that maximises the IDX formulae is selected. The cost of measuring x-th attribute is represented as

$$IDX\ x = \frac{DI_X}{C_x}$$

C4.5 applies a greedy approach at every step that chooses that attributes with the highest information gain ratio while IDX uses a lookahead strategy that looks n test ahead, where n is the parameter that is set by the user[41]. The shortcoming of these algorithms highlighted above is that they are only sensitive to the cost of test overlook-ing the argument of misclassification error. However, one the objectives of the research is to built an efficient algorithm that is sensitive to both cost of test and misclassifica-tion cost. Though, There are some tree induction techniques that meet these criteria and are discussed below.

2.4.4 Inexpensive Classification with Expensive Tests (ICET)

The ICET[55] algorithm is an amalgam of a genetic algorithm known as GENESIS [24] and a TDIDT decision tree algorithm. The decision tree induction applied was C4.5[49] but a modified version of an ICF. In other words, the tree induction used is the EG2, as described in the subsection above.

ICET applies a two-level search approach. On the bottom level, EG2 uses the stan-dard TDIDT strategy to perform a greedy but cost effective search through pathways of decision tree while on the top level, GENESIS performs a genetic search through a space of biases. In the ICET algorithm, EG2 is applied differently. The n costs, C_i used in EG2 attribute selection function are considered as bias parameters, not as costs. In other words ICET manipulates the bias of EG2 by adjusting the cost C_i parameter.

Having GENESIS being initiated with a population of randomly generated individ-ual(bit strings) with measured fitness of each, ICET represents a bit string as a bias for EG2. When an EG2 is applied on a data using the bias of a given bit string, the bit string is evaluated by calculating its fitness which is the average cost of classification of the decision tree that is generated by EG2. This process is repeated on new individuals generated by a mutation and crossover scheme and after a fixed number of generations, ICET halts and its output of the decision tree is decided by the fittest individual. Es-meir & Markovitch [14] realised that this algorithm can use additional time resources to produce more generations and hence to widen its search in the space of costs. They stressed that in building trees, EG2 prefers attributes with high information gain (and low test cost). Therefore, when the concept to learn hides interdependency between attributes, the greedy measure may underestimate the usefulness of highly relevant attributes, resulting in more expensive trees. Secondly, even if ICET may overcome the above problem by re-evaluating the attributes, ACT searches the space of parameters globally, regardless of the context. This imposes a problem if an attribute is impor-tant in one sub-tree but useless in another. To handle these deficiencies Esmeir & Markovitch proposed a model known as Lookahead-by-Stochastic-ID3 (LSID3) com-bined with Anytime Cost-sensitive Tree learner (ACT) [14].

2.4.5 LDID3 & ACT

In this particular work[14], LSID3 was developed on and ACT learner to exploit ad-ditional time to produce trees of lower costs. The LSID3 [15] is cost-insensitive algo-rithm, which can produce more accurate trees when given more time. The algorithm uses stochastic sampling techniques to evaluate candidate splits. However, it is not de-signed to minimize test and misclassification costs. This is when the ACT algorithm comes into play. The primary goal of this combined model is to trade the learning time for reduced test and misclassification costs.

LSID3 adopts the general TDIDT scheme starting from the entire set of training examples, partitions it into subsets by testing the value of an attribute. Subsequently, it recursively builds sub-trees. Unlike greedy inducers, LSID3 invests more time re-sources for making better split decisions. For every candidate split, LSID3 attempts to estimate the size of the resulting sub-tree were the split to take place and by Occams ra-zor it favours the one with the most minimal expected size [4]. The estimation is based on a biased sample of the space of trees rooted at the evaluated attribute. In Stochastic EG2 (SEG2), attributes are split semi-randomly, proportionally to their ICF. Due to this stochastic nature we expect to be able to escape local minima for at least some of the trees in the sample. To obtain a sample of size S, ACT uses EG2 once and SEG2 S1 times. Contrary to ICET, EG2 and SEG2 are given direct access to context-based costs. In other words, if an attribute has already been tested its cost would be zero and if another attribute that belongs to the same group has been tested, a group discount is applied.

The sub-trees generated from this model is evaluated using an estimator. For a leaf with v training examples, of which w are misclassified the expected error is defined as the upper limit on the probability for error, i.e., EE(v; w; c f) = U c f (w; u) where c f is the confidence level and U is the confidence interval for binomial distribution. The expected error of a tree is the sum of the expected errors in its leafs. In ACT model, the expected error is used to approximate the misclassification cost. Assume a problem with jCj classes and a misclassification cost matrix M. Let c be the class label in a leaf l. Let m be the total number of examples in l and mi be the number of examples in l that belong to class i. The expected misclassification cost in l is

$$mc\ \ cost(l) = EE(m; mmc; c\ f\ )\frac{1}{\underset{j\ \ j}{C\quad 1}}\ \underset{1!=c}{\text{å}}\ Mc; i = EE(m; m\ \ mc; c\ f\ )mc$$

As mentioned earlier, one problem with the cost-sensitive tree algorithms discussed above is the cost of the root node in classification of rules. The root node happens to initiates every decision tree learning process thereby, the threat of incurring its cost on every instance classified is unavoidable. This problem could is eliminated by integrating a tree induction technique on the prism [10] algorithm to produce a model that eradicates the recycled cost of the root node. This new model is refereed to as the Cost-Sensitive Prism (CS-PRISM).

2.5 Cost-Sensitive Analysis in Medical and Health Care

With developments of Information Technology, data mining techniques has proven to be successful in assisting Health care practitioners with decision making procedures. Extracted knowledge from a medical database could improve disease diagnosis, treat-ments, prognosis and the overall management of patients. As illustrated in earlier Sec-tions, diverse problem from different aspects of life have been addressed implementing cost-sensitive techniques but this subsection focuses on some various applications of these techniques in medical analysis.

A huge percentage of the population in the United states, approximately 300,000 people suffer from epilepsy. The most challenging aspect of this neurological disease is the unpredictable nature of seizures. Many epileptics live in constant worry that a seizure could strike impromptu resulting in humiliation, social stigma, or injury. This led to an investigative study to develop a patient-specific classification model to cat-egorize between preictal and interictal features extracted from EEG dataset[40]. The classifier built was a Cost-Sensitive Support Vector Machine (CSVM). Support Vector Machine (SVM) was chosen to be modified because of its robustness for estimating predictive models from noisy, sparse and high-dimensional data. The CSVM was op-timized for each patient using the misclassification cost and the relative weights of in-terictal to preictal windows. Five-fold cross validation was performed with the training set. Each classification model is built with the learning set to minimize the following cost function:

$$\frac{1}{2}kwk^2 + C^+ \underset{i2+class}{\mathring{a}} x_i + C \underset{j2\ class}{\mathring{a}} x_j$$

Once optimized through the above process, the classifier was applied on the test set, generating (predicting) the label for the unknown dataset. The proposed algorithm was applied to EEG recordings of 9 patients in the Freiburg EEG database, totalling 45 seizures and 219-hour-long interictal, and it produced sensitivity of 77.8% (35 of 45 seizures) and the zero false positive rate using 5-minute-long window of preictal via double-cross validation. This approach can help an embedded device for seizure prediction, consume less power by real-time analysis.

The cost function above was modified applied an Computer aided detection (CAD) to aid radiologists detect nodule in the early stages of lung cancer diagnosis[7]. Peng et al noted that the radiologist needed a CAD system to eliminate or reduce the false positives while retaining high sensitivity from an unbalanced dataset. Imbalanced data however, initiated unequal misclassification costs, making common classification methods inappropriate. In order to solve this problem, a novel Cost learner (CS-SVM) was designed and Particle Swarm Optimization (PSO) is employed as the optimization strategy due to its fast and effective solution space exploration. This algorithm basically used a wrapper approach to perform the search for the potentially optimal misclassifica-tion cost, intrinsic parameters and feature subset of CS-SVM. When evaluated on a 3D Lung nodule dataset, the technique outperforms many other exiting standard methods, as well as specific imbalanced data learning methods. This indicated the effectiveness of the SC-SVM mdoel on imbalanced and unequal misclassification cost data.

In an event to improve the identification of malignant cases of breast cancer Schae-fer & Nakashima [51] employed a cost-sensitive fuzzy classification approach for breast cancer diagnosis. In particular, the fuzzy classification system incorporates the concept of misclassification costs of training patterns for an improved classification performance. The cost term allows more emphasis on the correct classification for a certain class which is particularly useful for breast cancer diagnosis. The classification system consist of N fuzzy if-then rules each of which has a form as in Equation below

Rule R_j : I f x_1 is A_{j1} and::::and x_n is A_{jn}

T hen Class C_j with CF_j; j = 1; 2; :::; N;

Two steps are involved here specification of antecedent part and determination of consequent class C_j and the grade of certainty CF $_j$. The fuzzy rule is then re-formulated as a cost minimisation function which is introduced for each training pattern in order to handle the cost of its misclassification. The fuzzy rule is equation is then modified to

$$b_{Class\ h}(j) = \underset{x_p 2Class\ h}{\mathring{a}} m_j(x_p)\ w_p$$

where p is the cost associated with training pattern p. When applied on the Wiscon-sin breast cancer dataset, results revealed good classification results, confirming that through appropriate definition of costs, improving of classification sensitivity is ob-tained.

To effectively identify the characterization of gait abnormalities in Parkinson’s Disease (PD), a cost sensitive SVM learner is applied to improve the classification model[54]. Data were collected from 23 subjects with a clinical diagnosis of PD at-tending the UCSF Parkinson’s Disease Clinic and Research Center, San Francisco. Of the subjects diagnosed with PD, 11 had a clinically significant disturbance of gait, and 12 had no such disturbance. Data was collected through wireless inertial sensors that were attached to subjects’ feet and transmitted via blue tooth. After preprocessing, SVM is then used for classification by optimally separating classes and maximizing the margin between classes therefore, minimizing the classification error. A non-linear radial basis function (RBF) kernel used in the SVM process to maps the data into a new space, where a k-dimensional hyperplane is used to separate the classes. A cost-sensitive SVM classification model is built for the binary classification task by solving the optimization equation

$$\min_{w;b;x} \quad \frac{1}{2}k\; w^{2}\; k + C_{PD} \overset{n_{PD}}{\underset{i=1}{\mathring{a}}} x_i + C_{Control} \overset{n_{control}}{\underset{j=1}{\mathring{a}}} x_j$$

$$\text{subject to} \quad y_i(w^{T} x + b) \;\; 1 \;\; x_i;\, x_i \quad 0$$

where w; b; x are optimization parameters; C_{PD} and $C_{Control}$ are the costs for mis-classifying PD and control subjects, respectively; y 2 [1; 1]^{n} is the vector of labels, i.e. 1 for PD and -1 for control, for n_{P}D and n_{c}ontrol PD and control data points, respectively; and x is the feature value for the data point to be classified. Points are classified based on the sign of w^{T} x + b, i.e. on which side of the hyperplane the data point falls on. After K-fold cross-validation results was compared to the common SVM and results revealed the cost sensitive learner reflected a performance of 100% speci-ficity and precision, while maintaining sensitivity of close to 89%.

The cost sensitive approach was focused on SVM to improve the prediction of dif-ferent surgical complications within the American College of Surgeons National Sur-gical Quality Improvement Program registry[11]. The technique is targeted improving the performance relative to both supervised (binary or 2-class SVM) and unsupervised (1-class SVM) methods, as well as the use of cost-sensitive weighting techniques, for cost-efficient predictions. Given the training set, 2-class SVM classification is applied for finding a maximum margin boundary. A transfer learning formulation transfers the 2-class boundary to the 1-class SVM task by an optimization problem. This model reg-ularizes the 1-class SVM solution towards the model parameter Formula obtained from the 2-class SVM classification task. When this model is evaluated on data from over 30,000 patients undergoing inpatient surgical procedures, the transfer SVM algorithm generally achieved better discrimination of patients at high risk of different morbidity outcomes than both 2-class and 1-class SVM models. In addition, this approach con-sistently outperformed 2-class SVM models where cost-sensitive weighting is used to overcome class imbalance, as well as logistic regression.

A different approach was applied to improve classification of unbalanced medical data[59]. to overcome the problem of unbalanced dataset a cost-sensitive extension of the Regularized Least Square) RLS algorithm that penalizes errors of different samples with different weights was implemented. In the experiments, the unbalance levels of the data set was changed by gradually taking out samples from one class. According to the unbalance level of the data set, weights were chosen automatically for each class. Then the weighted RLS classifiers were adjusted by those weights and were compared with the original RLS classifiers and Support Vector Machine classifiers. The experi-mental results showed that the accuracy performance of weighted RLS after balancing was significantly improved.

A cost sensitive tree induction together with Genetic programming (GP) to build decision tree to minimize not only the expected number of errors, but also the expected misclassification costs through a novel constraint fitness function[34]. In this approach, a GP-based classifier with a simple fitness function, the Rate of Correctness (RC) was first developed. This function is to achieve a high classification accuracy (equivalent to a low classification error), as much as possible. To initiated this error-based GP classifier to cost-sensitive classification, the training data was manipulated either the rebalancing or reweighing method. This was easily achieved by taking the weights of instances into the fitness function. Afterwards, a novel constrained fitness function is proposed and incorporated into the error-based GP-based classifier, which is able to guide GP to search for promising solutions. Ideally, the solutions are to trade off be-tween high cost errors and low cost errors so that the overall cost is minimised. After a 9-fold cross validation on a heart disease dataset of 270 samples result showed that the modified GP model achieved a mean cost of 0.472.

To determine the most appropriate medical test during disease diagnosis, a cost-sensitive machine learning algorithms is designed to learning diagnosis process of heart diseased patients[36]. Firstly, a lazy decision tree learning algorithm that minimizes the sum of attribute costs and misclassification costs is proposed. After which, the expected total misclassification cost when selecting attributes for splitting is used to produce trees with a smaller total cost. This step is done to produced more accurate split with total cost for test examples (new patients). A case studied carried out on a heart diseased dataset, revealed that the model is cost-effective and outperforms previ-ous methods

A study by Krawczyk et al [30] presented a Cost-Sensitive Ensemble Classification (ECSE) model to handle to problem imbalance distribution of malignant and benign cases. Basically, a pool of base classifiers were selected. Each of them makes a deci-sion with respect to a class. The combined classifier then makes a decision according to a weighted voting rule. The base classifier chosen is the EG2 algorithm whose deci-sion tree is based on misclassification cost rate. A local sequential search at each node is performed to boost the recognition rate of the minority class and assigning a greater cost to a case when a minority object is misclassified. This method was applied to two medical datasets, the Wisconsin dataset and breast thermogram dataset and com-pared with other ensemble learners. Nevertheless, this approach performed the best, and gives statistically significantly better sensitivity (81.02%) compared to all other tested ensembles and in terms of specificity ECSE gives statistically the best results.

2.6 Rule-Based Learners

These are types of classification technique used to predict group membership for data instances. These are algorithms that are programmed to efficiently learn rules from sample training data and builds a model. The model is applied to new objects to clas-sify new rules. Rule-based algorithm provide mechanism that generate rule by

1. Concentrating on a specific class at a time
2. Maximising the probability of the desired classification

Rule generated suggests that strong relationships exist between items and their clas-sified outcomes. However, Bramer (2013) illustrated that some of the generated associ-ation rules tend to have little or no significant value[5]. Therefore it is of added interest to provide further information to express how reliable the rules are . For example:

IF Age > 50 AND Sex = male AND Alcohol = yes T HEN Disease = Diabetes(Probability = 0:7)

The probability value also considered as the rule confidence level, shows how often male who are 50 years and above, consumes alcohol and tend to have diabetes. How-ever, if the confidence level tends to go lower, the generated rule would be considered unfit. There are several models FOR generating rules of relationship between item sets but for this research experiential, the focus is on the prism modular algorithm

2.6.1 Prism

Prism [9] is a ruled based learner developed by Cendrowska. It is designed to generate rules for each class by looking at the training data and adding rules that completely describe all tuples in that class. Prism generate rules by using the IF to initiate a rule, AND to separate related item sets and T HEN which finally proposes classification class. An example of a prism is represented as

IF a == 1 AND b == 2 AND c == 3 T HENClass == X

Rules generated are considered correct or perfect. That is, the accuracy of the prism generated rules is 100%[48]. successes of the rule is measured by a formulae repre-sented as P=T , where P is the number of positive instance and T is the total number of instance covered by the rule.

Prism takes a training set as input with each attribute and attribute value entered as a file of ordered, each set being terminated by a classification class. Information about the attributes and classifications are input and the individual rules are output for each of the classifications listed in terms of the described attributes[9]. Basically, prism uses the 'take the first rule that fires' conflict resolution approach resulting to the most important rule first[5]. Prism generates the rules by concluding each of the possible classes in turn. Each of these rules are generated term by term represented as 'attribute = value'[5]. However, the attribute-value chosen at each step is that with the highest probability of the target outcome class, and for each new class. A training set with instances of more that one classification class, therefore for each class x:

1. Calculate the probability of occurrence of class = x for each attribute-value pair
2. select the pair with maximum probability and create a subset of the training set comprising all the instances which contain the selected attribute/value combination.
3. Repeat Steps 1 and 2 for this subset until it contains only instances of class = x. The induced rule is a conjunction of all the attribute/value pairs selected in creating the homogeneous subset
4. Remove all instances covered by this rule from the training set,
5. Repeat Steps 1-4 until all instances of class = x have been removed.

When the rules for one classification have been exhausted, the training set is re-stored to its original state and the algorithm process is applied again to induce set of rules covering the next classification class. The Prism pseudo-code is represented as

```
For each class C
Initialize E to the instance set
W hile E contains instances in class C
Create a rule R with an empty le f t hand side that predicts class C U ntil R is per f ect (or there are no more attributes to use) do
For each attribute A not mentioned in R; and each value v; Consider adding the condition A = v to the le f t hand side o f R Select A and v to maximize the accuracy p=t
(break ties by choosing the condition with the largest p)
Add A = v to R
Remove the instances covered by R f rom E
```

For example, consider a dataset of cancer patients in Figure 4, the following steps below is be applied to generate induced rules

Stage One: A class (Malignant) is chosen at random and the probability condition of attribute/value pair for the class is computed

For Class = Malignant Age [young] = 1/2[0.5]
Age [Old] = 2/4[0.5
Blood Group [AB]= 2/3[0.6]
Blood Group [O]= 1/3[0.3]
Organ [Pancreas] = 3/3[1]

Patient ID	Age	Blood Group	Organ	Class
Patient001	Old	AB	Pancreas	Malignant
Patient002	Young	AB	Breast	Benign
Patient003	Old	0	pancreas	Benign
Patient004	Old	0	Breast	Benign
Patient005	Old	0	Pancreas	Malignant
Patient006	Young	AB	Pancreas	Malignant

Figure 4: Cancer Dataset

Organ [Breast] = 0/2[0]

The attribute/value pair with the highest probability function is used to generate the first rule which is

IFOrgan = [Pancreas]T HENClass = Malignant

Stage Two: This Introduces the AND command by selecting all pancreatic organ pa-tients and stage one is repeated excluding the organ attribute

Age [young] = 1/1[1]

Age [Old] = 2/3[0.6
Blood Group [AB]= 2/2[1]

Blood Group [O]= 1/2[0.5]

In such situation either attribute/value pairs with the highest probability function is used to generate the next rule
IF Organ = [Pancreas] AND BloodGroup = [AB] T HEN Class = Malignant

Stage Three:The next course of action is to select all pancreatic cancer patients with blood group AB repeat the stage two process until all the samples of malignant class are exhausted. Afterwards, we return to the original dataset and redo the whole process for the next class (Benign) until the whole dataset is learned.

This approach initiates a divide and conquer techniques that produces compact accurate rules that completely exterminates the redundancy of the root node, which posses a risk of increasing the average cost of classification. The next Section (3) would explain how the Prism modular is manipulated to generate cost-sensitive rules.

3. METHODOLOGY

This section explains the details of the experimental project so far. The datasets ap-plied, preprocessing technique, CS-PRISM rule inducer and evaluation method would all be illustrated. At the end of this section, examiners should be able to interpret the experimental method and its relevance to the research objectives.

WEKA software environment[3] and Netbeans are the platforms applied to run the experimental analysis. The WEKA mining environment is the podium used to test the CS-PRISM algorithm on medical dataset and Netbeans is used to manipulate the algo-rithm code using Java programming language.

3.1 Datasets

The dataset applied for this study is different from the typical format of machine learn-ing datasets of attributes, attributes value and class. As illustration from the previous sections, cost sensitive learning is dependant on cost of test which also represent the cost values of each attributes. However, the dataset used is the Hepatitis Prognosis Dataset granted by Gail Gong et al[22] which has test values assigned to every at-tribute. This dataset deals with the prognosis of known diagnosis with the problem of determining the outcome of the disease represented in the class attribute. The test value assigned a nominal cost generated by either by asking a question to the patient or by performing a basic physical examination on the patient. For a example, a his-tological examination of the liver costs $81.64, asking the patient whether a histology was performed only costs $1.00. Henceforth, the prognosis can exploit the information conveyed by a decision to perform a histological examination made during the diagno-sis.

There are two class variables, to Die and to Live. The dataset originally contains 20 attributes (inclusive of the class attribute) and 155 cases with missing values. However, data prepossessing techniques is done to accommodate the CS-PRISM algorithm. The table in Figure 5 the test cost of Hepatitis patients plied in the experiment.

3.1.1 Data Preprocessing

In order for the Hepatitis dataset to accommodate the CS-PRISM, various preprocess-ing tasks are applied. These procedures include data cleaning, data reduction to elimi-nate irrelevant instances and data transformation to convert instances into formats un-derstandable by the algorithm.

Data Cleaning and Reduction

The is the first step applied to visually detect any discrepancy, since technical errors are inevitable with medical data. The algorithm happens to be sensitive to missing values making it logical to ignore such instances. Also, cases with incomplete and inconsis-tent values were also avoided as well as samples without class values.

Attribute(Test)	Description	Cost
Age	years lived	$1
Sex	gender	$1
Steroid	patient on steroids	$1
Antiviral	patient on antiviral	$1
fatigue	patient reports fatigue	$1
Malaise	patient reports malaise	$1
Anorexia	patient anorexic	$1
Liver big	liver big on physical exam	$1
Liver firm	liver firm on physical exam	$1
Spleen palpable	spleen palpable on physical	$1
Spiders spider	veins visible	$1
Ascites	ascites visible	$1
Varices	varices visible	$1
Bilirubin	levels in blood	$727
Alk phosphate	alkaline phosphotase	$727
Albumin	albumin blood test	$727
Protime	protime blood test	$830
Histology	was histology performed?	$1
class	prognosis of hepatitis	prognostic class: live or die

Figure 5: Hepatitis patients Test Cost

Data Transformation

In the Hepatitis dataset, all the attribute values are represented as numeric values, de-spite the fact that most of the attributes were originally in nominal format. With the sensitive nature of the proposed algorithm on numeric data, the data values are man-ually transformed back to nominal forms. Although, few attributes were originally numeric and a categorical pattern was applied to covert them to nominal standards. Some of such attributes include Age, Bilirubin, Albumin etc. For Age, it was easy to categorise patients between 0 40 as youth and 40 above as adult. With spe-cialised medical terms such as Bilirubin, Albumin etc, medical libraries [46][8] aided the grouping of the attributes as either normal or abnormal. For instance, Case-Lo & Krucik [8] revealed 0:4 1:9mg=dL as normal levels of Bilirubin in the blood and any values otherwise is represented as abnormal. A more specialised medical term (sgot) is left out of the experiment due to insufficient knowledge to correctly categorise its instances, hereby reducing the attribute to 19, including the class.

The entire preprocessing procedure illustrated above was done manually, thereby, 20 instances that meet the required criteria is selected at random to test the algorithm. Figure 5 shows a list of the attribute selected for the experiment and their test cost.

3.1.2 Dataset Preparation

Data preparation methods was applied in the experiment to boost model accuracy. This entails dividing the datasets into two sections; One of the section is used to train the model and the other to test the model. The Hepatitis samples were split into two halves, training and test data by applying the percentage split option in WEKA. This is also known as hold out sampling. 60% of the dataset is used to build the model and 40% for testing the model. Although the algorithm is trained against the trained data but the accuracy is calculated on the whole data set.

3.2 CS-PRISM

As mentioned before, a major purpose in this experiment is to propose a solution that enables the generation of cost-sensitive rules, eliminating the cost of the root node of induce tree learners. This resolution makes CS-PRISM a better cost efficient algorithm than the tree induced algorithms. CS-PRISM is a hybrid of the standard prisms[9] learner and a tree induction algorithm. The decision tree induction algorithm is the modified C4.5[49] of an Information Cost Function (ICF). In other words, the EG2[42].

As illustrated on Section 2.6.2, Prism generate compact rules based on attribute-value probability to a class, but with CS-PRISM, the concept of EG2 tree learner is integrated for cost sensitive rules. The cost of test (attribute) approach from the EG2 is used to select attributes and the prism scheme of generating rule covers the classi-fication error. Therefore, initiating a model sensitive to test cost as well as classifi-cation error. Information Cost Function (ICF) of EG2 used to select attributes based on their information gain DI_x and their test cost C_x is modified to a Probability Cost Function (PCF). PCF eliminates the information gain DI_x function and replaces it with attribute/value probability to class DP_x, though retaining the test cost function. How-ever, the PCF for the AGE attribute with respect to a selected class (DIE) is defined below as:

$$2^{DP_{Age}}$$

$$PCF_{Age} = \frac{1}{(C_{Age}+1)^{w}}$$

In this modified equation, DP_{Age} is the attribute-value probability associated with the Age attribute of the given class DIE. C_{Age} is the cost of measuring the Age attribute. The parameter w adjusts the strength of a lower cost attributes. When w is zero, cost is ignored and selection of PCF_{Age} is equivalent to selection by DP_{Age} . On the other hand, When w = 1 is hugely bias by cost. The bias nature of w is the major reason EG2 was selected for modification instead of other cost functions[53][41]. Nez[42] does not propose an ethical way of setting w. Therefore, with CS-PRISM, w is set to 1 and the selection measure applied is:

$$\frac{2^{DP_{Age}} \quad 1}{C_{Age}+1}$$

EG2 selects attributes that maximizes the ICF. Likewise, the CS-PRISM selects gener-ated rules from an attribute-value that maximizes PCF. When applied on the Hepatitis training sets, CS-PRISM applies a three-phase rule generating strategy

3.2.1 Rule-Base Construction

In the first phase, a first line rule is generated. To achieve this, all the attribute-value probability DP_x associated with a given class (DIE) is calculated. Using the attribute test cost, their respective Probability Cost Functions (PCF) is computed. The pair with the highest PCF value is then selected and used to generate the first rule of the class.

In the next phase, all instances covered by the rule are separated out. Basically, a subset of the training dataset is generated comprising of all instances of the selected pair. Afterwards, stage one and two is repeated until the remaining instances are con-quered with respect to the class. CS-PRISM eliminates all the instances covered by the generated rule until the class (DIE) is exhausted. These mechanism are then repeated on the next class until all instances of the training set are covered.

The rules generated on the training test are tested on the test dataset to boost model accuracy. Subset covered by a rule doesn't need to be explored any further therefore giving no room for an additional cost. The block diagram of the proposed algorithm is represented in Figure below.

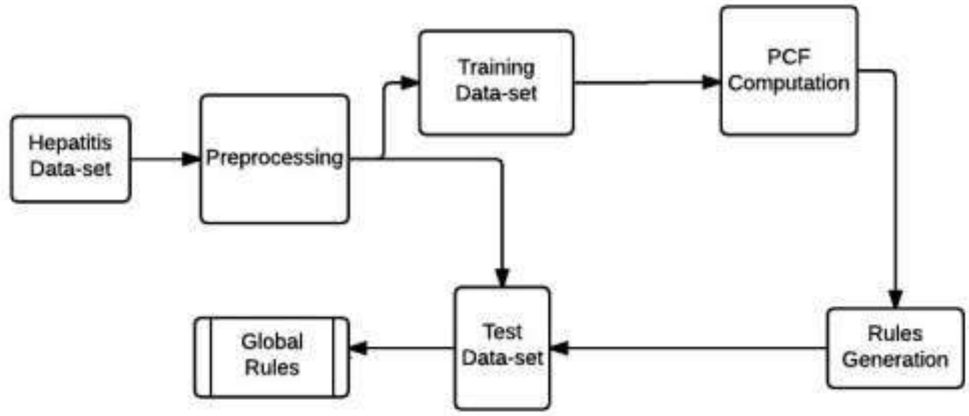


Figure 6: The Block Diagram: CS-PRISM

3.3 Experimental Phase

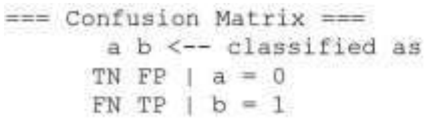
This phase is divided into two stages. The first stage is the predictive analysis, which is concerned with the application of the algorithm on the Hepatitis dataset, classification performance of the generated rules. The tasks of algorithms testing and performance evaluations are illustrated at phase one. The second phase, descriptive analysis, is concerned with the certification of the rules generated. It also discuses techniques on how the rules are verified with medical ontology so as to avoid rules of bad examples.

3.3.1 Predictive Analysis

The proposed cost-sensitive classifier is tested on the Hepatitis data sets from granted by Gail Gong et al[22] of random sampling with 60% training and 40% to test the model. A brief description about the data set tested is presented in Figure 5 and Sec-tion 3.1. Unfortunately, the model's result so far is unconvincing in terms of percentage of correctly classified rule as well as no visual rule. This requires further modification of the CS-PRISM source code.

However, in a situation where CS-PRISM is fully implemented and produces re-sults, assessment methods are applied to reveal how well the algorithm performed in classifying rules with respect to their actual class. However, there are evaluation met-rics to monitored in assessing the classified rules.

Confusion metrics are the building blocks for computing the evaluation measures. They are tables that analysis how well proposed classifiers can recognise rules of differ-ent classes. They consist of four terms, True Positives(TP), True Negatives(TN), False Positives(FP) and False Negatives(FN). TP and TN would acknowledge CS-PRISM is getting rules right while FP and FN would indicate its getting rules wrong.



```
=== Confusion Matrix ===
      a b <-- classified as
     TN FP | a = 0
     FN TP | b = 1
```

Figure 7: Confusion Matrix structure[21]

The evaluation metrics to asses the performance of the algorithm are accuracy level, sensitivity, precision, F-measure and confusion matrix.

The accuracy would measures the percentage of the generated rules that are cor-rectly classified as LIV E or DIE. This is represented as

$$\text{Accuracy} = \frac{T\,P+T\,N}{P+N}$$

P = total LIV E and
N = total DIE

Sensitivity: Also referred to as recall measures the ability of the algorithm to pre-dict rules of a certain class. In other words, Recall measures completeness of classified rules. This is represented as

$$\text{Sensitivity} = \frac{T\ P}{T\ P + FN}$$

Precision This measures the fraction of the rules generated correctly classified out all correctly classified instances. and is also referred to as the true malignant rate or specificity. Precision is computed as

$$\text{Precision} = \frac{T\ P}{T\ P + FP}$$

An alternative to monitor precision and recall is to combine them into a single mea-sure. This approach is the F-measure and it measures the harmonic mean of precision and recall and is computed as

$$F \quad \text{measure} = 2\,\frac{\text{Precision} \quad \text{Recall}}{\text{Precision} + \text{Recall}}$$

The nature of the experiment advocates clinical and therapeutic importance of cor-rectly classified metastasis. For this reason, FP and FN measures need to be critically monitored. FP indicates the rule generated are classified as DIE when their actual classification is LIV E. Cases like this could initiate unnecessary novel treatments. However, it gets worst with False Negatives (FN), which indicates rules classified as positive (LIV E) while their true classification is negative (DIE). Such rules could mislead doctors and medical practitioners from appropriate treatments that could con-sequently lead to death.

An ideal measure for a classification model is a high F-measure of DIE class, and a recall of higher than 0.5 (when FN is less than T P). However, for the proposed research experiment, this would be unsatisfactory since any number of FN measured would result to high cost of patients loosing their lives. Thus, it would be logical to give more weight to the cost FN errors than FP. To achieve this, heuristic search of classification parameters would be adjusted to seek for different weights for prediction errors to voluntarily produce unbalanced rules of minimal FN

3.3.2 Descriptive Analysis

One major problem with CS-PRISM is that classified rules with high precision or other evaluation measure may defile medical ontology. For instance, lets Assume the medical information for 3 patients of the hepatitis dataset shown in Figure 8. For each patient, a decision is taken whether it is medically logical or not.

Applying the PCF to generate a rule with respect to class DIE, The first line of rule would be
IF Bilirubin = Normal T HEN Class = DIE

Bilirubin	Alk phosphate	Class
Normal	abnormal	DIE
Normal	Normal	DIE
Abnormal	Abnormal	LIVE

Figure 8: Subset: Hepatitis dataset

On the other hand, this rule contradicts the medical sense. A physician would ques-tion why patient with normal levels of Bilirubin in the blood still die. This could be as result of an abnormal level of Alk phosphate or any other factors which is not repre-sented in the rule. The easiest and fastest way to evaluate such rules is through visual validation by health or medical practitioner, to detect any inapplicable classified rules.

In lack of medical practitioner, another way to verify rules of relatively high pre-cision and recall would be to apply text mining tools to on-line health and medical databases such Pubmed [18], EMBASE [60] etc, to affirm the generated rules compli-ments medical ontology. However, only known rules would be verified but unknown rules would be considered for more verification process.

3.4 Optimized CS-PRISM

The CS-PRISM is inescapable of improvements. This subsection highlights the dif-ferent setbacks of the algorithm and how to contain it. One major problem with cover rule-based classifiers as well as CS-PRISM are the stopping criterion measure. This is expected to repulse the algorithm from generating bad rules. If the stopping criterion is not met, generated rules might cover negative examples from other classes. In other words, iteratively adding unnecessary conditions to the rules. This condition is met by finding the best condition with comparing candidate conditions for each attribute and evaluating them with respect to a chosen measure[37].

CS-PRISM does not have natural ability to avoid over-fitting. Nevertheless to im-prove computational efficiency of the rules generated, pruning methods are applied to eliminate whole rules or single rule terms from a CS-PRISM rule set. Bramer's [?] introduces a J-pruning measure which has the ability to assess the information content of rule, in order to ignore rules with very low information content. This approach im-proves the algorithm by initiating the criteria that stops generating needless rules of very low support. Another optimization of the model is to accommodate datasets of missing and numeric values. Data transformation from numeric to nominal forms is impossible or tedious to accomplish due to the large nature of datasets. However, it would be worthwhile to medicine and data miners to improve the CS-PRISM model to incorporate numeric instances as well as missing and noisy values. These instances are inevitable in large datasets and may posses some hidden knowledge. To fully explore learning intelligence, CS-PRISM would be upgraded to accommodate these highlighted optimizations.

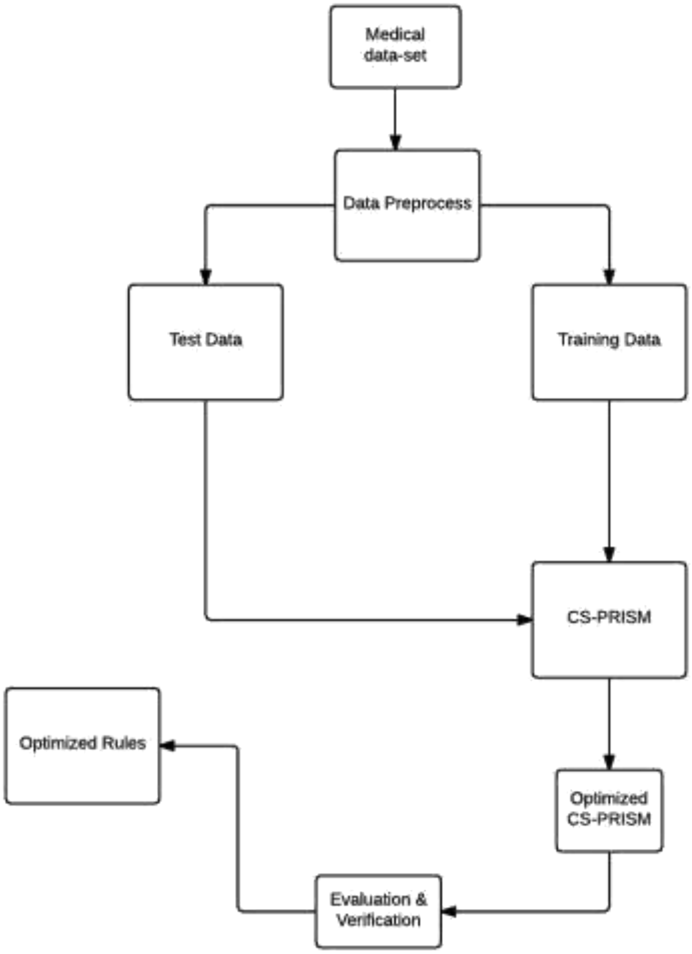


Figure 9: Project Flow Chart

4. CONCLUSION AND FUTURE WORKS

Classification of generated rules from medical datasets is an important research prob-lem. However, generating a cost-sensitive rule is a more critical issue that could help in cost-efficient aided diagnosis and treatment. Although, previous research based on decision tree induction have been carried out to classify cost sensitive rules but the cost of the root node seems to be the loophole with these algorithms. In this research, the CS-PRISM is proposed for generating classification rules from the data set without the redundancy cost of the root node.

Subsequently, modifications to the CS-PRISM source code would be continued to yield convincing results. This would entail, correctly assigning the test costs values to their appropriate attributes. Thereafter, other real medical datasets would be ex-plored to evaluate CS-PRISM performance. A period of one month should be viable for this task. After which, the issue of stopping criteria would be addressed to pre-vent rules of bad examples. This is achieved by implementing an argument technique initiated by [37]. This procedure finds the best condition of a rule by comparing can-didate conditions for each attribute and assuring the coherence with arguments and also, evaluating them with respect to a chosen measure. This practice should feasible in couple of months to implement. In events of over-fitting, pruning approaches to measure information contents of generated rules would be explored to eliminate rule with minimal support. The CS-PRISM would be considered incompetent if it does not accommodate numeric and missing data instances. The model is then compared with other cost-sensitive algorithms such as ICET[55] and other standard rules-based clas-sifiers in terms of classification accuracy. Finally, the classified rules are verified on medical ontology database[18]etc. to asses the relevance of the classified rules. The work-flow plan is represented in Figure 9.

REFERENCES

1. *The kdd process for extracting useful knowledge from volumes of data. Commu-nications of the ACM, 39:11.*
2. *Michael R. Berthold, Nicolas Cebron, Fabian Dill, Thomas R. Gabriel, Tobias Kotter,¨ Thorsten Meinl, Peter Ohl, Christoph Sieb, Kilian Thiel, and Bernd Wiswedel. KNIME: The Konstanz Information Miner. In Studies in Classifica-tion, Data Analysis, and Knowledge Organization (GfKL 2007). Springer, 2007.*
3. *Alselm Blumer, Andrzej Ehrenfeucht, David Haussler, and Manfred K. Warmuth. Occam's razor. Inf. Process. Lett., 24(6):377–380, April 1987.*
4. *Max Bramer. Using j-pruning to reduce overfitting of classification rules in noisy domains. In Abdelkader Hameurlain, Rosine Cicchetti, and Roland Traunm-ller, editors, Database and Expert Systems Applications, volume 2453 of Lecture Notes in Computer Science, pages 433–442. Springer Berlin Heidelberg, 2002.*
5. *Max Bramer. Principles of Data Mining. Springer London, 2nd edition edition, 2013.*
6. *Friedman J. Olshen R. Stone C. Breiman, L. Classification and regression trees. California. 1984.*
7. *Peng Cao, Dazhe Zhao, and O. Zaiane. Measure oriented cost-sensitive svm for 3d nodule detection. In Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE, pages 3981–3984, July 2013.*
8. *Christine Case-Lo and George Krucik. Bilirubin blood test. 2015.*
9. *Jadzia Cendrowska. Prism: an algorithm for inducing modular rules. Int J Man-Machine Studies, pages 349–370, 1987.*
10. *Jadzia Cendrowska. Prism: An algorithm for inducing modular rules. Interna-tional Journal of Man-Machine Studies, 27(4):349 – 370, 1987.*
11. *Chih-Chun Chia, Z. Karam, Gyemin Lee, I. Rubinfeld, and Z. Syed. Improving surgical models through one/two class learning. In Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE, pages 5098–5101, Aug 2012.*
12. *Dursun Delen, Glenn Walker, and Amit Kadam. Predicting breast cancer sur-vivability: a comparison of three data mining methods. Artificial Intelligence in Medicine, 34(2):113 – 127, 2005.*
13. *Stephan Dreiseitl and Lucila Ohno-Machado. Logistic regression and artificial neural network classification models: a methodology review. Journal of Biomed-ical Informatics, 35(56):352 – 359, 2002.*
14. *Saher Esmeir and Shaul Markovitch. Anytime induction of cost-sensitive trees. In J.c. Platt, D. Koller, Y. Singer, and S. Roweis, editors, Advances in Neural Information Processing Systems 20, pages 425–432. MIT Press, Cambridge, MA, 2007.*
15. *Saher Esmeir, Shaul Markovitch, and Claude Sammut. Anytime learning of de-cision trees. Journal of Machine Learning Research, 8:891–933, 2007.*
16. *T. Fawcett. Feature Discovery for Problem Solving Systems. Doctoral disserta-tion, Department of Computer Science, University of Massachusetts, 2nd edition edition, 1993.*
17. *T. Fawcett and F.J. Provost. Combining data mining and machine learning for effective user profiling. In Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 2nd edition edition, 1996.*
18. *Seshadri R Powell EC Freedman SB, Adler M. Oral ondansetron for gastroen-teritis in a pediatric emergency departmen, volume 6. PubMed, 2006.*
19. *Jerome H. Friedman and Werner Stuetzle. Projection pursuit regression. Journal of the American Statistical Association, 76(376):817–823, 1981.*

20. *Francesco Gagliardi. Instance-based classifiers applied to medical databases: Di-agnosis and knowledge extraction. Artificial Intelligence in Medicine, 52(3):123-139, 2011.*
21. *G. Giarratana, M. Pizzera, M. Masseroli, E. Medico, and P.L. Lanzi. Data mining techniques for the identification of genes with expression levels related to breast cancer prognosis. pages 295–300, June 2009.*
22. *Gail Gong. The hepatitis prognosis dataset. 1988.*
23. *Diana Gordon and Donald Perlis. Explicitly biased generalization. Computa-tional Intelligence, 5(2):67–81, 1989.*
24. *J.J. Grefenstette. Optimization of control parameters for genetic algorithms. Sys-tems, Man and Cybernetics, IEEE Transactions on, 16(1):122–128, Jan 1986.*
25. *Yu Hai-yan, Li Jing-song, Han Xiong, Hu Zhen, Zhou Tian-shu, Chi Jie, and Zheng Tao. Data mining analysis of inpatient fees in hospital information system. In IT in Medicine Education, 2009. ITIME '09. IEEE International Symposium on, volume 1, pages 82–85, Aug 2009.*
26. *Jiawei Han and Micheline Kamber. Data Mining Concepts and Techniques. Mor-gan Kaufman Publishers, 2nd edition edition, 2006.*
27. *J. Hermans, J. D. F. Habbema, and A. T. van der Burgt. Cases of doubt in alloca-tion problems, k populations, 1973. mit franz. Zsfassung.*
28. *Mehdi Kaytoue, Sergei O. Kuznetsov, Amedeo Napoli, and Sbastien Duplessis. Mining gene expression data with pattern structures in formal concept analysis. Information Sciences, 181(10):1989 – 2001, 2011. Special Issue on Information Engineering Applications Based on Lattices.*
29. *B. Krawczyk, G. Schaefer, and M. Wozniak. A cost-sensitive ensemble classifier for breast cancer classification. In Applied Computational Intelligence and Infor-matics (SACI), 2013 IEEE 8th International Symposium on, pages 427–430, May 2013.*
30. *B. Krawczyk, G. Schaefer, and M. Wozniak. A cost-sensitive ensemble classifier for breast cancer classification. In Applied Computational Intelligence and Infor-matics (SACI), 2013 IEEE 8th International Symposium on, pages 427–430, May 2013.*
31. *Nada Lavra. Selected techniques for data mining in medicine. Artificial Intelli-gence in Medicine, 16(1):3 – 23, 1999. Data Mining Techniques and Applications in Medicine.*
32. *Arthur M. Lesk. Introduction to Bioinformatics. Oxford University Press, 1st edition edition, 2014.*
33. *Jin Li, Xiaoli Li, and Xin Yao. Cost-sensitive classification with genetic pro-gramming. In Evolutionary Computation, 2005. The 2005 IEEE Congress on, volume 3, pages 2114–2121 Vol. 3, Sept 2005.*
34. *Jin Li, Xiaoli Li, and Xin Yao. Cost-sensitive classification with genetic pro-gramming. In Evolutionary Computation, 2005. The 2005 IEEE Congress on, volume 3, pages 2114–2121 Vol. 3, Sept 2005.*
35. *C.X. Ling, V.S. Sheng, and Qiang Yang. Test strategies for cost-sensitive decision trees. Knowledge and Data Engineering, IEEE Transactions on, 18(8):1055– 1067, Aug 2006.*
36. *C.X. Ling, V.S. Sheng, and Qiang Yang. Test strategies for cost-sensitive decision trees. Knowledge and Data Engineering, IEEE Transactions on, 18(8):1055– 1067, Aug 2006.*
37. *Krystyna Napieraa and Jerzy Stefanowski. Addressing imbalanced data with ar-gument based rule learning. Expert Systems with Applications, 42(24):9468 – 9481, 2015.*
38. *Tim W. Nattkemper, Bert Arnrich, Oliver Lichte, Wiebke Timm, Andreas Degen-hard, Linda Pointon, Carmel Hayes, and Martin O. Leach. Evaluation of radio-logical features for breast tumour classification in clinical screening with machine learning methods. Artificial Intelligence in Medicine, 34(2):129 – 139, 2005.*
39. *T. Netoff, Yun Park, and K. Parhi. Seizure prediction using cost-sensitive sup-port vector machine. In Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE, pages 3322–3325, Sept 2009.*
40. *M. Nez. Economic Induction: A Case Study, in Proceedings del Third European Working Session on Learning. 1988.*
41. *Steven W. Norton. Generating better decision trees. In Proceedings of the 11th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'89, pages 800–805, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.*
42. *MARLON NUNEZ. The use of background knowledge in decision tree induc-tion. Machine Learning, pages 231–250, 1991.*
43. *Vipin Pang-Ning Tan, Michael Steinbach. Introduction to data mining. Addison-Wesley, 1st edition edition, 2005.*
44. *Merz C. Murphy P. Ali K. Hume T. Brunk C. Pazzani, M. Reducing misclassifi-cation costs: Knowledge-intensive approaches to learning from noisy data. Pro-ceedings of the Eleventh International Conference on Machine Learning, 1994.*
45. *Medicine Plus. Alp - blood test. 2015.*
46. *F.J. Provost. Goal-directed inductive learning: Trading off accuracy for reduced error cost. AAAI Spring Symposium on Goal-Driven Learning, 1994.*
47. *Shakhina Pulatova and Zhao Xinyou. Covering (rule based) algorithm. 11:25–30, 2007.*
48. *J. R. Quinlan. Induction of decision trees. MACH. LEARN, 1:81–106, 1986.*
49. *R.S. Santos, S.M.F. Malheiros, S. Cavalheiro, and J.M. Parente de Oliveira. A data mining system for providing analytical information on brain tumors to pub-lic health decision makers. Computer Methods and Programs in Biomedicine, 109(3):269 – 282, 2013.*
50. *G. Schaefer and T. Nakashima. Hybrid cost-sensitive fuzzy classification for breast cancer diagnosis. In Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE, pages 6170–6173, Aug 2010.*
51. *Ming Tan. Cost-sensitive learning of classification knowledge and its applications in robotics. Machine Learning, 13(1):7–33, 1993.*

52. *Ming Tan and Jeffrey C. Schlimmer. Cost-sensitive concept learning of sensor use in approach and recognition. In Proceedings of the Sixth International Workshop on Machine Learning, pages 392–395, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.*
53. *Tien, S.D. Glaser, and M.J. Aminoff. Characterization of gait abnormalities in parkinson's disease using a wireless inertial sensor system. In Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE, pages 3353–3356, Aug 2010.*
54. *Peter D. Turney. Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm. CoRR, cs.AI/9503102, 1995.*
55. *Peter D. Turney. Types of cost in inductive concept learning. CoRR, cs.LG/0212034, 2002.*
56. *C.J. van Rijsbergen. Information Retrieval. 2nd edition edition, 1979.*
57. *Floor Verdenius. A method for inductive cost optimization. In Proceedings of the European Working Session on Learning on Machine Learning, EWSL-91, pages 179–191, New York, NY, USA, 1991. Springer-Verlag New York, Inc.*
58. *Nguyen Ha Vo and Yonggwan Won. Classification of unbalanced medical data with weighted regularized least squares. In Frontiers in the Convergence of Bio-science and Information Technologies, 2007. FBIT 2007, pages 347–352, Oct 2007.*
59. *Davies K. Wilkins T, Gillies RA. EMBASE versus MEDLINE for family medicine searches: can MEDLINE searches find the forest or a tree?, volume 51. Can Fam Physician.*
60. *John C Wooley and Herbert S Lin. Catalyzing inquiry at the interface of comput-ing and biology. National Research Council (US) Committee on Frontiers at the Interface of Computing and Biology, 2005.*
61. *Duen-Yian Yeh, Ching-Hsue Cheng, and Yen-Wen Chen. A predictive model for cerebrovascular disease using data mining. Expert Systems with Applications, 38(7):8970 – 8977, 2011.*