İSTANBUL TECHNICAL UNIVERSITY ★ INSTITUTE OF SCIENCE AND TECHNOLOGY

# COMBINING CLASSIFICATION ALGORITHMS USING DEMPSTER'S RULE OF COMBINATION

Ph.D. Thesis by

Hüseyin AYGÜN, M.Sc.

**Department :** Computer Engineering

**Programme:** Computer Engineering

**Supervisor :** Prof. Dr. Eşref ADALI

JULY 2005

# Acknowledgements

First of all I would like thank to my thesis advisor Prof. Eşref Adalı for his guidance throughout this study. As a person full of positive energy, he always encouraged me all the way along from the beginning of the study until the end. I would like to express my deep appreciation for Prof. Şebnem Baydere and Associate Prof. Coşkun Sönmez for their valuable criticism during the proposal presentation and the study check presentations.

I can not go without thanking to Assistant Prof. Ibrahim Sogukpinar who encouraged me to start the PhD study.

Most of my gratitudes would go to my dear wife Şükran and lovely kids Gülşah and Caner who spend much time without me. They always supported me to complete my PhD study on time. They also let me use the only computer at home almost all of the time, putting aside the rules of equality in the family.

I cannot forget the discussion that I had with Dr. Nafiz Arica while I was writing my first paper. At the same time I appreciate his help in reviewing my thesis. His comments were very helpful.

I would also like to thank to Metin Balcı and Muhammed Altun from the Turkish Naval Research Center for the discussion on the issues Data Fusion and Dempster-Shafer Method.

**Contents**

# ABBREVIATIONS

| | | |
|---|---|---|
| ADABOOST | : | Adaptive Boosting |
| BEL | : | Belief |
| BM | : | Baye’s Method |
| BPA | : | Basic Probability Assignment |
| CART | : | Classification and Regression Trees |
| CV | : | Cross Validation |
| DF | : | Data Fusion |
| DM | : | Data Mining |
| DOC | : | Degree of Confidence |
| DROC | : | Dempster’s Rule of Combination |
| DSM | : | Dempster-Shafer Method |
| MLR | : | Multi-response Linear Regression |
| MDT | : | Meta Decision Trees |
| NNGE | : | Non-Nested Generalized Exemplars |
| OLAP | : | Online Analytical Processing |
| PLA | : | Plausibility |
| REP | : | Reduced Error Pruning |
| RIDOR | : | Ripple-Down Rule Learner |
| RIPPER | : | Repeated Incremental Pruning to Produce Error Reduction |
| WEKA | : | Waikato Environment for Knowledge Analysis |

## List of Tables

## List of Figures

# COMBINING CLASSIFICATION ALGORITHMS USING DEMPSTER'S RULE OF COMBINATION

## SUMMARY

The constantly growing volume of data makes it impossible to analyze and capture the valuable knowledge among large amounts of data using the current statistical methods. Because of the insufficiency of the current analysis tools, new solutions have been found for extracting the valuable but hidden knowledge among huge data. These solutions are data mining and data fusion.

Data mining tries to extract implicit, previously unknown, and potentially useful information from large amounts of data. It is a process that uses a variety of data analysis tools to discover patterns and relationships in data that may be used to make valid predictions.

Data fusion, on the other hand, is the process of combining information coming from different sensors. Data fusion algorithms are mostly used for target tracking and target identification purposes in intelligence, surveillance and reconnaissance operations in the defense sector.

Although data mining and data fusion are two reciprocal processes completing each other, people are generally working on these two areas independently without having any interaction. There are few studies which combine the techniques used in these areas in order to improve the performance of classification.

In this study, we propose a method for improving classification results. The method consists of combining the classification results using Dempster's Rule of Combination, considering the classifier outputs as beliefs. In the combination we utilize some of the existing classification algorithms. We do experiments with different data sets to evaluate our proposed method and we arrive at the conclusion that combining the classifier outputs using Dempster's Rule of Combination gives better classification results than each of the classification algorithms.

Dempster's Rule of Combination does not include degree of confidence presently. We also propose a method for using degree of confidence during the combination in order to improve the accuracy of the classification. The experiments performed on several data sets show that the employment of degree of confidence during combination results in more precise classification results.

## *DEMPSTER-SHAFER ALGORİTMASININ KULLANIMI İLE SINIFLANDIRMA ALGORİTMALARININ BİRLEŞTİRİLMESİ*

### ÖZET

Sürekli olarak büyümekte olan veri, mevcut istatistiksel yöntemlerin kullanılmasıyla, büyük miktardaki veri içindeki değerli bilginin bulunmasını ve analiz edilmesini imkansız hale getirmektedir. Mevcut analiz araçlarının  yetersizliği nedeniyle çok büyük miktardaki veri içindeki değerli fakat saklanmış bilginin bulunup çıkarılması için yeni çözümler bulunmuştur. Bu çözümler veri madenciliği ve veri füzyonudur.

Veri madenciliği önceden bilinmeyen, fakat yararlı bilginin büyük miktardaki veri arasından bulunup çıkarılmasıdır. Veri madenciliği, veri içindeki örüntünün keşfedilmesini ve geleceğe ilişkin tahminler yapılmasında kullanılabilecek ilişkilerin çıkarılması sağlayan analiz araçlarını kullanır.

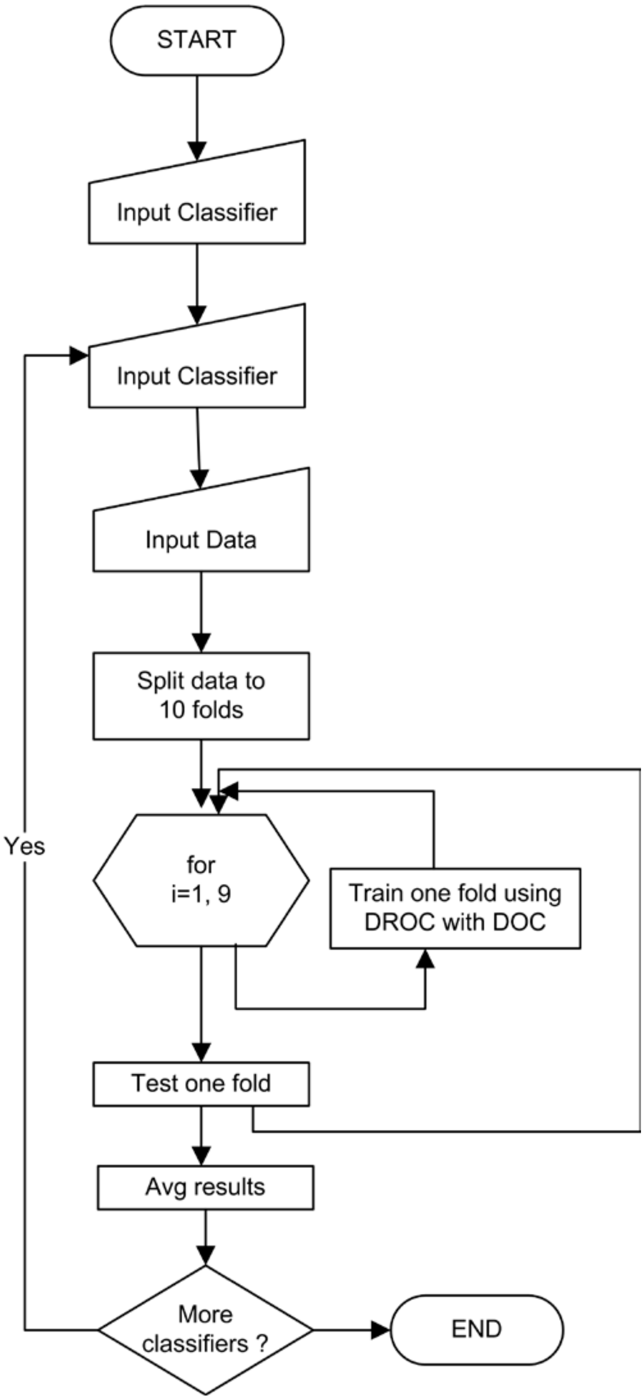
Veri füzyonu ise farklı sensörlerden gelen bilgilerin birleştirilmesi işlemidir.  Veri füzyonu algoritmaları, savunma sektöründe hedef takibi, hedef kimlik tespiti amacıyla istihbarat, keşif ve gözetleme operasyonlarında kullanılmaktadır.

Veri madenciliği ve veri füzyonu birbirini tamamlayan prosesler olmasına rağmen, araştırmacılar bu iki alanda birbirinden bağımsız olarak, herhangi bir ilişkiye girmeden çalışmaktadırlar. Sınıflandırmanın etkinliğini artırmak için bu alanlarda kullanılan teknikleri birleştiren çok az sayıda çalışma mevcuttur.

Bu çalışmada sınıflandırma sonuçlarını iyileştirmek için yeni bir yöntem önerilmektedir. Sözkonusu yöntem Dempster'in Birleştirme Algoritmasını kullanarak farklı sınıflandırma algoritmalarından elde edilen sonuçların birleştirilmesinden oluşmaktadır. Önerilen yöntemi desteklemek amacıyla farklı veri takımlarıyla yapılan deneyler sonucunda Dempster'in Birleştirme Algoritmasının kullanımıyla yapılan birleştirme işleminin, birleşimde kullanılan her bir sınıflandırma algoritmasından daha başarılı sonuçlar verdiği görülmüştür.

Dempster'in Birleştirme Kuralında güven derecesi mevcut değildir. Bu çalışmada aynı zamanda, sınıflandırmanın hassasiyetini artırmak amacıyla, birleştirme işleminde güven derecesi kullanımı için de bir yöntem önerilmektedir.  Çeşitli veri takımlarıyla yapılan deneyler sonucunda önerilen yöntem ile daha hassas sınıflandırma sonuçları elde edildiği görülmektedir.

.


START
Input Classifier
Input Classifier
Input Data
Split data to 10 folds
for i=1, 9
Train one fold using DROC with DOC
Test one fold
Avg results
More classifiers ?
Yes
END

# 1 INTRODUCTION

John Naisbitt, the auther of 1982 bestseller Megatrends *says "We are drowning in information but starved for knowledge. This level of information is clearly impossible to be handled by present means. Uncontrolled and unorganized information is no longer a resource in an information society, instead it becomes the enemy".* This statement has gained more meaning today. We are in need of finding the valuable knowledge among huge amount of unorganized data. There are two solutions to this problem: Data mining and data fusion.

Data mining tries to extract implicit, previously unknown, and potentially useful information from large amount of data (Han and Kamber, 2000). It is a process that uses a variety of data analysis tools to discover patterns and relationships in data that may be used to make valid predictions. Data mining algorithms are widely used in areas such as market analysis, risk analysis, fraud detection, text mining and web analysis. In market analysis, for instance, data mining tries to find clusters of customers who share the same characteristics in target marketing, determines customer purchasing patterns over time, and finds associations between product sales (Han and Kamber, 2000).

Data fusion, on the other hand, is the process of combining information coming from different sensors. Data fusion algorithms are mostly used for target tracking and target identification purposes in intelligence, surveillance and reconnaissance operations in the defense sector. In White (1987) the definition of data fusion is given as: A process dealing with the association, correlation, and combination of data and information from single and multiple sources to achieve refined position and identity estimates, and complete and timely assessments of situations and threats, and their significance. The process is characterized by continuous refinements of its estimates and assessments, and the evaluation of the need for additional sources, or modification of the process itself, to achieve improved results.

Although data mining and data fusion are two reciprocal processes completing each other according to Waltz (1999), people are generally working on these two areas independently without having any interaction. There are a few studies which combine the techniques used in these areas in order to improve the performance. The study in Waltz (1999) explores the integration of data mining and data fusion techniques in order to expand the ability to detect and classify non-literal target signatures, hidden in disparate data sets such as imagery data, spatial data, video imagery, statistical data sets, textual data sets containing key words, phrases or concepts. The method suggested in Waltz (1999) tightly couples the discovery and detection processes using all available source data to provide cues and clues to intelligence and business analysts tasked with challenging investigative problems.

Returning back to data mining, the most important task of data mining is classification. Classification can be examined in two groups: supervised classification and unsupervised classification. In supervised classification, first of all, classifiers are built and then classification is performed according to these classifiers. Unsupervised classification, on the other hand, tries to locate clusters of records having similar characteristics.

Some of the classification algorithms are Decision Trees, OneR, IBK, Naïve Bayes, K-means clustering, Kohonen Vector Quantization, Autoclass and so on.

There is a continuous research for combining outputs of different classification algorithms in order to increase the performance of classification in various fields, including statistics, econometrics, probabilistic forecasting and machine learning.

Various methods have been developed to construct diverse single classifiers. The first method is to generate single classifiers by the same classification technique but on different training samples. The second method is to build single classifiers by the same technique on the same training set, but including different input attributes. The third method for constructing different classifiers is to apply the same classification techniques, but with random configurations. Lastly, the multiple classifiers can be learned by different classification techniques.

The most common  methods are given in the following paragraphs.

The algorithm maintaining a weight distribution over the training observations is boosting (Freund and Schapire 1996;  Friedman et al. 2000; Hastie et al. 2001).  For all the training observations, the weights are assigned equally at the beginning.

Multiple classifiers obtained by using different learning algorithms on a single dataset is combined by stacking. At the beginning, a set of base-level classifiers is generated. In the second phase, in order to combine the outputs of the base-level classifiers a meta-level classifier is learned. More information can be found in classifiers (Dzeroski and Zenko 2004; Gama and Brazdil, 2000).

Seewald and Furnkranz (2001) suggested grading which learns a meta-level classifier for each base-level classifier. The meta-level classifier predicts whether the base-level classifier is to be trusted.

Besides the efforts for classifier combination mentioned above, there is also some study for integrating data mining and data fusion techniques for combining classifiers. Some of these techniques using Dempster-Shafer evidence combination rule for combination outputs of classifiers are given below.

Wachowicz and Carvalho (2002) integrated data fusion and data mining techniques for automating the fundamental reasoning process in environmental information systems. Mahajani and Aslandogan (2003) used Dempster-Shafer's theory of evidence for combining medical data. The concept of a "weighted Dempster-Shafer evidence combining rule" is used in context aware computing in (Wu et al., 2002). Al-Ani and Deriche (2002) use a technique based on Dempster-Shafer theory for combining classification algorithms. Fabiani (1994) uses likelihood vectors with degree of confidence.

Every classifier has a certain level of uncertainty. What is missing in most of the above algorithms is, firstly, uncertainty management. Every classifier has uncertainty to some extent. The second issue regarding the algorithms mentioned is the lack of degree of confidence. Degree of confidence is the success that a certain classifier has displayed on similar data sets in the past. A classification algorithm must be able to use degree of confidence in order to give more precise classification results.

Dempster-Shafer's Method, in other words evidence combination rule, has the capability to handle uncertainty. Dempster-Shafer's Method is widely used for combining evidences obtained from different sensory information in the area of data fusion. Dempster-Shafer Method does not require exact probability values in order to combine evidences. Pieces of information, some being incomplete, obtained from different information sources can be combined using Dempster's Rule of Combination.

## 1.1 Contribution of the Thesis

In this study, we propose a method for improving classification results. The method consists of combining the classification results using Dempster's Rule of Combination, considering the classifier outputs as beliefs, with the employment of degree of confidence of classifiers. In the combination we utilize some of the existing classification algorithms and do experiments with different data sets to check if we get better classification results from our proposed method.

Dempster’s Rule of Combination does not include degree of confidence presently. In our proposed method we use degree of confidence during the combination in order to improve the accuracy of the classification. We perform experiments on several data sets to show that the employment of degree of confidence during combination results in more precise classification results.

In summary, our proposed combination method using Demspter’s Rule of Combination does the following contributions:

- Employment of degree of confidence during the combination.
- Uncertainty management in combining classifiers
- Achievement of better classification results.

We perform experiments by combining several existing classifiers. The results of the experiments show that combining classifiers using Demspter’s Rule of Combination with the employment of degree of confidence not only performs better than each of the classifiers taking place in the combination but also performs better than the current hybrid classifiers.

The results of the experiments also show that the employment of degree of confidence during the combination give more precise classification results which also decrease uncertainty in the combination.

## 1.2 Organization of the Thesis

The thesis is organized as follows:

In Chapter 2 we give information about classifier combination techniques. We mention the superiorities and deficiencies of the hybrid algorithms.

In Chapter 3 we introduce the proposed method of combining classification results using Dempster’s Rule of Combination. We present Baye’s Theorem and Dempster-Shafer Method and compare them to each other after which we give detailed information about our proposed method of combining classifiers using Dempster’s Rule of Combination. We also present the employment of degree of confidence during the combination.

In chapter 4, we give detailed information about the data sets that we use in the experiments. First we experiment with the existing classifiers and we do experiments with the current hybrid classification algorithms on the same data sets. We, later, show the results of the experiments performed on the data sets using the proposed method of combining classifiers using Dempster's Rule of Combination. We, then, present the results of experiments by employing degree of confidence during the classifier combination with the proposed method. In Chapter 5, we finally  present the concluding remarks and future work.

# 2 OVERVIEW OF CURRENT STUDIES ON HYBRID ALGORITHMS

Classification is the one of the most important tasks in data mining. Classification can be examined in two groups: supervised classification and unsupervised classification.

In supervised classification, first of all, classifiers are built and then classification is performed according to these classifiers. Unsupervised classification, on the other hand, tries to locate clusters of records having similar characteristics.

Some of the classification algorithms are listed below:

- Supervised classification algorithms
    - Decision Tree
        - C4.5
        - Rule Learner (PART)
    - OneR
    - IBK
    - Naïve Bayes
- Unsupervised classification algorithms
    - K-means clustering
    - Kohonen Vector Quantization
    - Autoclass (Bayesian Classification System)

Detailed information about supervised and unsupervised classification algorithms can be obtained from Han and Kamber (2000) and Witten and Frank (2000).

There is a continuous research for combining outputs of different classification algorithms in order to increase the performance of classification in various fields, including statistics, econometrics, probabilistic forecasting and machine learning.

The purpose of combining classifiers is to reduce the mean square error of the classification. In linear combinations, the key issue is to determine the optimal linear coefficients for multiple forecasts in the combination. The study on classifier combination started in the 1960s, when Bates and Granger (1960) explored the weighted average of multiple forecasts to reduce the variance of forecast errors. Granger and Ramanathan (1984) and Clemen (1989) provide a detailed and comprehensive review of various linear combination methods seen in the forecasting literature.

Linear combinations of predictors have also been studied extensively in the field of machine learning. Examples include Perrone and Cooper's (1993) ensemble method and Hashem's (1997) optimal linear combination for neural networks, Wolpert's (1992) stacked generalization and Breiman's (1996) stacked regressions for various models including linear regression and CART.

In the literature of combining classifiers for classification problems, the output from a single classifier could be categorical (Hansen and Salomon, 1990; Kang et al., 1997; Lam and Suen, 1995; Xu et al. 1992), continuous (Huang and Suen, 1994; Turner and Ghosh, 1996), or a ranked list of classes (Ho et al., 1994). Schemes for combining single classifiers include weighted or unweighted majority voting based on categorical outputs from classifiers (Kittler et al., 1998), linear combination based on probabilistic outputs (Turner and Ghosh, 1996; Kittler et al., 1998), combination by logistic regression (Ho et al., 1994), various Bayesian methods (Kang et al., 1997; Kittler et al., 1998; Xu et al. 1992), and neural networks with the output of single classifiers as input (Huang and Suen, 1994).

Various methods have been developed to construct diverse single classifiers. The first method is to generate single classifiers by the same classification technique but on different training samples. For example, bootstrapping, cross-validation and reweighing training observations have been used to resample training sets (Breiman, 1996a; Breiman, 1996b; Freund and Schapire, 1996). The second method is to build single classifiers by the same technique on the same training set, but including different input attributes (Breiman, 1996a; Turner, 1996). The third method for constructing different classifiers is to apply the same classification techniques, but with random configurations. For example, different neural networks are generated by varying the initial weights (Hansen and Salomon, 1990), and various decision trees are constructed by randomly selecting an attribute to split (from the 20 top ranked attributes) at each internal node for C4.5 tree classifiers (Dietterich, 2000a). Lastly, the multiple classifiers can be learned by different classification techniques, such as linear and quadratic discriminant analysis, k- nearest neighbor, neural networks, CART and C4.5 (Lam and Suen, 1995; Woods et al. 1997). Several influential ideas on combining classifiers are presented in the following.

The weighted majority vote is a linear combination which is used when single classifiers produce categorical outputs. Most of the majority voting schemes are simple to implement and their simplicity allows for theoretical analysis.

Bagging, which stands for *Bootstrapping Aggregation* is proposed by Breiman in (1996b). It employs a Bootstrapping technique to draw training observations randomly with replacement from the original training set. Single classifiers are thus learned from these replicates of the original training set, and then combined by majority voting. In this sense, the linear coefficients in the combination are the same for all single classifiers.

Boosting is much more complex than bagging. Various versions of boosting have been developed since 1990 by Freund and Schapire (1996) and Friedman et al. (2000) and Hastie et al. (2001). The most used boosting algorithm is AdaBoost, representing Adaptive Boosting (Freund and Schapire, 1996). The central idea of boosting is to maintain a weight distribution over the training observations. Initially the weights are assigned equally for all the training observations.

Bagging and boosting procedures reduce the error rate substantially on both training and test sets (Breiman, 1996b; Friedman et al. 2000). Bagging works especially well for *unstable* classification techniques, where small changes of the training set would result in major changes in classifier outputs (Breiman, 1996b; Breiman, 1996c). Unstable classifiers, such as decision trees, neural networks, and rule based classifiers, are characterized by high variance, but are largely unbiased (Breiman, 1996b). In contrast, linear regression, linear discriminant analysis and nearest neighbor are stable techniques, and their performance is not improved by bagging or boosting (Breiman, 1996b; Breiman, 1996c).

Stacking combines multiple classifiers generated by using different learning algorithms on a single dataset. In the first phase, a set of base-level classifiers is generated. In the second phase, a meta-level classifier is learned that combines the outputs of the base-level classifiers.

To generate a training set for learning the meta-level classifier, a leave-one-out or a cross validation procedure is applied. For leave-one-out, we apply each of the base-level learning algorithms to almost the entire dataset, leaving one example for testing: When performing, say, 10-fold cross validation, instead of leaving out one example at a time, subsets of size one-tenth of the original dataset are left out and the predictions of the learned classifiers obtained on these.

Research in this area investigates what base-learners and meta-learners produce best empirical results (Dzeroski and Zenko 2004; Gama and Brazdil, 2000); how to represent class predictions (Ting and Witten, 1999); and how to define meta-features (Ali and Pazzani, 1996).

In contrast to stacking, no learning takes place at the meta-level when combining classifiers by a voting scheme (such as plurality, probabilistic or weighted voting). The voting scheme remains the same for all different training sets and sets of learning algorithms (or base-level classifiers). The simplest voting scheme is the plurality vote. According to this voting scheme, each base-level classifier casts a vote for its prediction. The example is classified in the class that collects the most votes.

Merz (1999) proposes a stacking method called SCANN that uses correspondence analysis do detect correlations between the predictions of base-level classifiers. The original meta-level feature space (the class-value predictions) is transformed to remove the dependencies, and a nearest neighbor method is used as the meta-level classifier on this new feature space.

Ting and Witten (1999) use base-level classifiers whose predictions are probability distributions over the set of class values, rather than single class values. The meta-level attributes are thus the probabilities of each of the class values returned by each of the base-level classifiers. The authors argue that this allows to use not only the predictions, but also the confidence of the base-level classifiers. Multi-Response Linear regression (MLR) is recommended for meta-level learning, while several learning algorithms are shown not to be suitable for this task.

Seewald and Furnkranz (2001) propose a method for combining classifiers called grading that learns a meta-level classifier for each base-level classifier. The meta-level classifier predicts whether the base-level classifier is to be trusted (i.e., whether its prediction will be correct). The base-level attributes are used also as meta-level attributes, while the meta-level class values are + (meaning correct) and − (meaning incorrect). Only the base-level classifiers that are predicted to be correct are taken and their predictions combined by summing up the probability distributions predicted.

Todorovski and Dzeroski (2000) introduce a new meta-level learning method for combining classifiers with stacking: meta decision trees (MDTs) have base-level classifiers in the leaves, instead of class-value predictions. Properties of the probability distributions predicted by the base-level classifiers (such as entropy and maximum probability) are used as meta-level attributes, rather than the distributions themselves. These properties reflect the confidence of the base-level classifiers and give rise to very small MDTs, which can (at least in principle) be inspected and interpreted.

Todorovski and Dzeroski (2003) report that stacking with MDTs clearly outperforms voting and stacking with decision trees, as well as boosting and bagging of decision trees. On the other hand, MDTs perform only slightly better than SCANN and selecting the best classifier with cross validation. Zenko et al. (2001) report that MDTs perform slightly worse as compared to stacking with MLR. Overall, SCANN, MDTs, stacking with MLR and SelectBest seem to perform at about the same level.

It would seem natural to expect that ensembles of classifiers induced by stacking would perform better than the best individual base-level classifier: otherwise the extra work of learning a meta-level classifier doesn't seem justified. The experimental results mentioned above, however, do not show clear evidence of this.

Another approach to meta-learning consists of learning from base learners. The idea is to make explicit use of information collected from the performance of a set of learning algorithms at the base level; such information is then incorporated into the meta-learning process.

Cascading by Gama and Brazdil (2000) is a related variant to Stacking where the classifiers are applied in sequence and there is no dedicated meta classifier. Each base classifier, when applied to the data, adds its class probability distribution to the data and returns an augmented dataset, which is to be used by the next base classifier. Thus, the order in which the classifiers are executed becomes important. Cascading does not use an internal cross validation like most other ensemble learning schemes and is therefore claimed to be at least three times faster than Stacking. On the other hand in Stacking the classifier order is not important, thereby reducing the degrees of freedom and minimizing chances for overfitting. Furthermore, cascading increases the dimensionality of the dataset with each step whereas Stacking's meta dataset has a dimensionality which is independent of the dimensionality of the dataset - i.e. the number of base classifiers multiplied with the number of classes.

Besides the efforts for classifier combination mentioned above, there is also some study for integrating data mining and data fusion techniques for combining classifiers. Some of these techniques using Dempster-Shafer evidence combination rule are presented below.

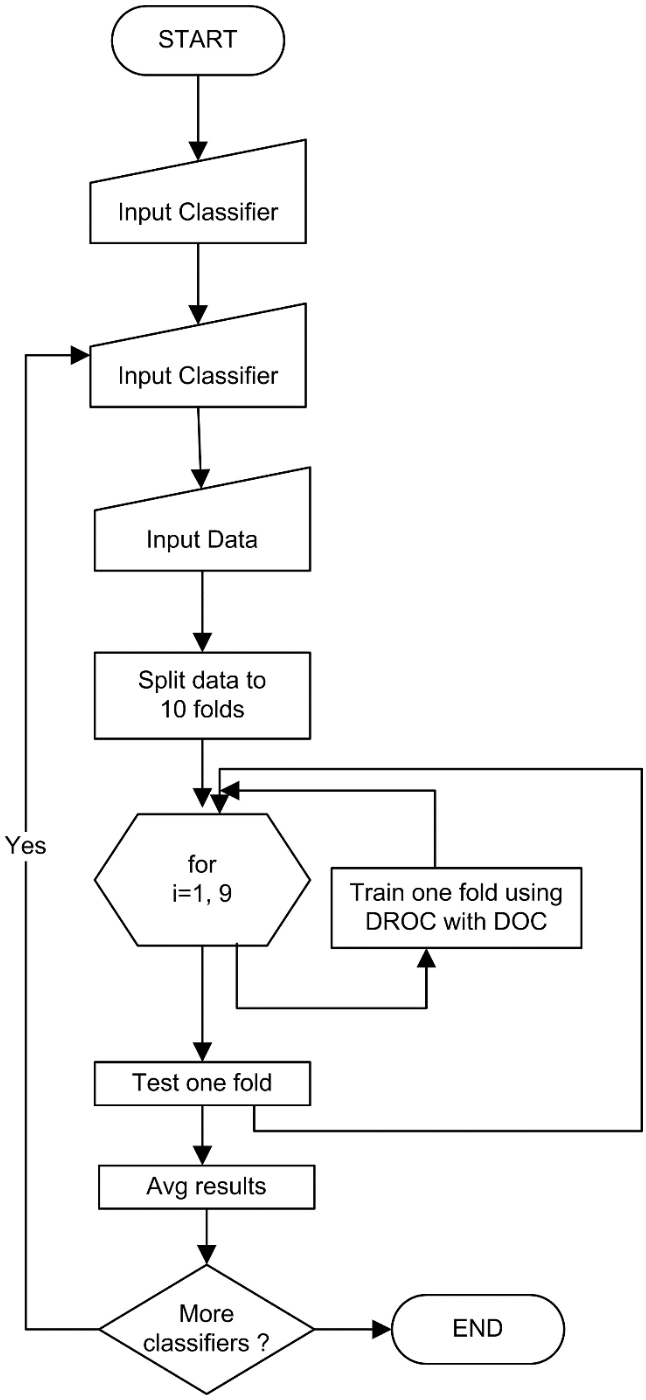
Wachowicz and Carvalho (2002) integrated data fusion and data mining techniques for automating the fundamental reasoning process in environmental information systems. The information gathered from the class level fusion/mining is used as the input for decision tree classification task. Their study shows how the knowledge generated can be used to produce maps of forest land cover and deforestation process.

Mahajani and Aslandogan (2003) used Dempster-Shafer's theory of evidence for combining medical data. Classifier outputs are used as a basis for computing beliefs. Dynamic uncertainty assessment is performed on class differentiation. The beliefs of three classifiers K-Nearest Neighbor (kNN), Naïve Bayesian and Decision Tree are combined using Dempster's Rule of Combination. The experiments with k-fold cross validation show that the nature of the data set has a bigger impact on some classifiers than others and the classification based on combined belief shows better overall accuracy than any individual classifier. The performance of Dempster's combination is compared with those of performance-based linear and majority vote combination models. The improvement achieved by Mahajani and Aslandogan (2003) in combining the results of the classifiers over single classifiers is between 0.1% and 1.1%. In their study no degree of confidence for the classifiers is employed during combination.

The concept of a "weighted Dempster-Shafer evidence combining rule" is used in context aware computing in (Wu et al., 2002). In their approach the weights of the sensors are used at the time of combination. But when tested, the combination of mass functions does not sum to 1, which is a contradiction to the essense of Dempster's Rule of Combination.

Al-Ani and Deriche (2002) use a technique based on Dempster-Shafer theory for combining classification algorithms. They report that they achieve an increase of 2-7 % in classification accuray when compared to other algorithms. Although they are successful in reducing the error rate and have good results in overall performance, their algorithm has computationally expensive training time. The authors indicate that training can be performed off-line without affecting the overall performance of the system. Although this may seem feasible for static data mining, it is not satisfactory for real time implementations where time is of crucial importance.

Fabiani (1994) uses likelihood vectors with degree of confidence. In his study degree of confidence is used for showing belief in the likelihood vector or for non-belief. No combination is performed in this work. For future study, the author indicates that degree of confidence should decreases against time.


START
Input Classifier
Input Classifier
Input Data
Split data to 10 folds
for i=1, 9
Train one fold using DROC with DOC
Test one fold
Avg results
More classifiers ?
Yes
END

# 3 COMBINING CLASSIFIERS USING DEMPSTER’S RULE OF COMBINATION

The motivation for this dissertation is to improve data mining algorithms using data fusion techniques. More specifically, we improve the result of classification algorithms by combining classification algorithms using Dempster’s Rule of Combination, the evidence combination method used in field of data fusion.

In data fusion, in order to represent and combine information obtained from different sensors, different approaches are available in the literature. Baye’s Theorem and Dempster-Shafer Method are the most famous among these approaches. In the following subsections we examine these two approaches and then compare them to each other.

## 3.1 Baye’s Theorem

The purpose of Baye’s Method is to modify the probability of a hypothesis according to prior probabilities and new data. In this context the probability of E when x is given is:

Prior probability of E: The probability  of an event before the evidence is seen.

Posterior probability of E: The probability  of an event after the evidence is seen.

$$\Pr[E \mid x] = \frac{\Pr[x \mid E]\Pr[E]}{\Pr[x]} \tag{3.1}$$

Degrees of belief are [illegible] Let A and B be two events, and X be the set of all possible events. The axioms of Probability theory are:

$$P(A) \geq 0 \tag{3.2}$$

$$\sum_{A \in X} P(A) = 1 \tag{3.3}$$

$$P(A+B) = P(A) + P(B) - P(A) \cdot P(B) \tag{3.4}$$

Let E be an event to be calculated, and x1 and x2 be the information obtained from two different data sources. In this case:

$$p(E|x_i)=\frac{p(x_i|E)\cdot p(E)}{p(x_i)}$$

$$p(E|x_1,x_2)=\frac{p(x_2|E,x_1)\cdot p(E|x_1)}{p(x_2|x_1)}$$

$$p(x_i)=\sum_{i=1}^{2}p(x_i|E)\cdot p(E) \tag{3.5}$$

$$p(E|x_1,x_2)=p(x_2|E,x_1)\cdot p(x_1|E)\cdot\frac{p(E)}{p(x_1,x_2)}$$

## 3.2 Dempster’s Shafer Method

The Dempster-Shafer theory was developed by Canadian statistician Arthur Dempster in the 1960's and extended by Glenn Shafer in the 1970's. The idea behind the Dempster-Shafer theory according to Shafer himself (Shafer, 1976) is:

> The theory of belief functions provides two basic tools to help us make probability judgments: a metaphor that can help us organize probability judgments based on a single item of evidence, and a formal rule for combining probability judgments based on distinct and independent items of evidence.

Dempster-Shafer Method assigns probability intervals to hypothesis . The input to Dempster-Shafer Method are the basic probability assignment (bpa) functions obtained from different sensor reports. It is possible to obtain new bpa functions by combining the bpa functions of the sensors. After combining the bpa’s, upper and lower probability values Belief and Plausibility ([bel,pla]) are calculated for the hypothesis.

Mathematically speaking, let m1 and m2 be two independent sensors, and Θ be the set of observed states. The information given by the sensors is defined on the power set $2^\Theta$ of Θ. Each element of $2^\Theta$ matches to a value between the interval [0,1] such that the sum of these values is 1. Mathematical notation is as follows:

$$m:2^\Theta\rightarrow[0,1] \tag{3.6}$$

$$m(\phi)=0 \tag{3.7}$$

$$\sum_{A \subset \Theta} m(A) = 1 \tag{3.8}$$

m : bpa  function or mass function. The positive values of m1 and m2 are called the 'focal elements'.

Dempster's Rule of Combination is given as:

$$m(A) = \frac{\sum_{B \cap C = A} m_1(B) \cdot m_2(C)}{1 - \sum_{B \cap C = \phi} m_1(B) \cdot m_2(C)} \tag{3.9}$$

The rule can be stated as: When two sources of information contradicts, i.e. when B∩C=Φ, the multiplication of the supporting bpa's is divided to the ones which are B∩C≠Φ.

Given Belief (Bel),  Plausibility (Pla) and A⊂Θ then Bel(A) and Pla(A) are defined as follows:

$$Bel\,(A) = \sum_{B \subset A} m(B) \tag{3.10}$$

$$Pla\,(A) = \sum_{A \cap B \neq \phi} m(B) \tag{3.11}$$

The relation between the Bel and Pla functions are;

$$Bel\ (A) \leq Pla\ (A) \tag{3.12}$$

Dempster's Rule of Combination combines bpa functions and obtains a new bpa function representing the combined bpa functions.

We need to point out that Dempster's Rule of Combination can be used in the case that the sensors are independent.

## 3.3 Comparison of Baye’s Theorem and Dempster-Shafer Method

As stated in section 4.1.1 Baye’s Method tries to modify the probability of a hypothesis according to prior probabilities and new data. On the other hand Dempster-Shafer Method assigns probability intervals to hypothesis. The main differences between the two methods are as follows:

- One needs to have prior and posterior probabilities in order to be able to use Baye’s Method. However Dempster-Shafer method does not need this information.
- The result is a probability value whereas Dempster-Shafer Method gives the result in terms of lower and upper probability values.
- Baye’s Method is more efficient in terms of computational performance. In the mean time Dempster-Shafer is equally or more efficient in some special cases.
- Dempster-Shafer can be applied to all situations where Baye’s Method can be applied.
- Baye’s Method can not be applied in circumstances without making some assumptions where uncertainty is high.

Dempster-Shafer can be used in combining information of different types whereas Baye’s Method can be used for calculating probabilities for hypothesis of the same type.

## 3.4 Combining Classifiers Using Dempster’s Rule of Combination

In this study, we propose a method for combining classification algorithms using Dempster’s Rule of Combination, assuming the results of the classifiers as beliefs, in order to improve the success of classification algorithms.

We assume that the basic probability assignments we use in our experiments are independent, which is a necessary condition for the use of the Dempster’s Rule of Combination. Suppose that we have some information and would like to measure its belief, then we can think of this process as a mapping from the "original information level" to the "belief level". Liu and Bundy (1992) explained that independence in the original information level would lead to independence in the belief level. But, if two independent belief functions are rooted to the original information level, then their original information may or may not be independent. For the problem of combining multiple classifiers, the original information level consists of outputs of the classifiers to be combined, while the belief level consists of the evidence of these classifiers (or their BPAs). The assumption that these BPAs are independent, whether obtained from independent or dependent original information, can hence justify the use of D-S theory.

Before we proceed any further with our proposed method, we would like give an example to show the use of Dempster's Rule of Combination in combining evidences coming from different data sources.

Let us suppose that there are three kinds of geese are swimming in a pool and we have two sensors reporting to us what they see. Let the first swimming goose be a goose with green head, the second one be a goose with red head, and the third one be a goose with yellow head.

State Space = (a,b,c)

Let a be the Goose with green head, b be the Goose with red head, and finally c be the Goose with yellow head .

Likelihood Vector of Sensor$_1$:

Likelihood1(a) = 0.6
Likelihood1(b) = 0.4
Likelihood1(c) = 0.2

Likelihood Vector of Sensor$_2$:

Likelihood2(a) = 0.3
Likelihood2(b) = 0.4
Likelihood2(c) = 0.4

Mass function (m), belief (bel) and plausability (pla) values will be as follows:

Mass

| | |
|---|---|
| m({a}) | = 0.37975015 |
| m({b}) | = 0.32951865 |
| m({a,b}) | = 0.07233336 |
| m({c}) | = 0.15043243 |
| m({a,c}) | = 0.03302175 |
| m({b,c}) | = 0.02865380 |
| m({a,b,c}) | = 0.00628986 |

bel

| | | |
|---|---|---|
| bel({a}) | = 0.37975015 | (Decision= Goose with green head) |
| bel({b}) | = 0.32951865 | |
| bel({a,b}) | = 0.78160217 | |
| bel({c}) | = 0.15043243 | |
| bel({a,c}) | = 0.56320433 | |
| bel({b,c}) | = 0.50860487 | |
| bel({a,b,c}) | = 1.00000000 | |

pla

| | |
|---|---|
| pla({a}) | = 0.49139513 |
| pla({b}) | = 0.43679567 |
| pla({a,b}) | = 0.84956757 |
| pla({c}) | = 0.21839783 |
| pla({a,c}) | = 0.67048135 |
| pla({b,c}) | = 0.62024985 |
| pla({a,b,c}) | = 1.00000000 |

The important point in this problem is the identification of what is swimming in the pool. In other words we would like to know if it is a goose with green head, a goose with red head, or a goose with yellow head. We do not want to make a decision such as it is “a goose with green head or a goose with red head” or “a goose with green head or a goose with yellow head” or “a goose with red head or a goose with yellow head”. So we need to check the value of belief in order to make the decision. Since the greatest of bel({a}), bel({b}) and bel({c}) is bel({a}), we make the decision as “goose with green head”. If the Belief values were the same or close to each other, then we would check the Plausibility values and choose the one with greater Plausibility value.

Returning to our proposed method, we first perform classification with different algorithms. Assuming the results of the classifiers as beliefs, we calculate mass functions for each classifier. We then combine the mass values in a pairwise fashion using Dempster’s Rule of Combination. Finally we calculate belief and plausibility values. The flow diagram of the proposed method is shown in Figure 3.1.

The proposed method consists of the following steps:

- Converting classifier outputs to mass functions.
- Performing pairwise combination using Dempster’s Rule of Combination.
- Making Belief and Plausibility Calculations
- Bel/Pla Calculation
- Presenting Classification results

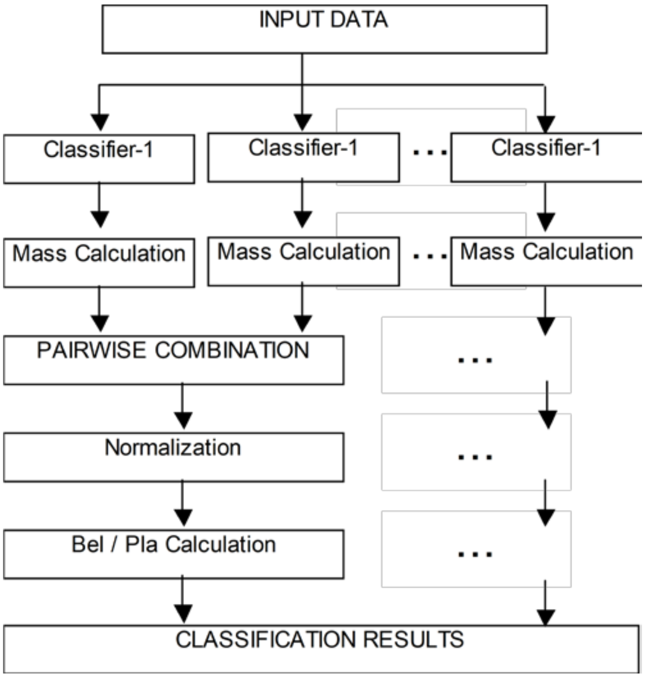


Figure 3.1 Combining Classifiers using Dempster's Rule of Combination

We use 10 fold cross validation in our proposed method. In 10 fold cross validation the dataset is split into 10 equal-sized folds. 9 folds are used for training and the remaining fold (i.e. the 10th fold) for testing. This process is repeated 10 times so that each fold is used for testing exactly once, thus generating one prediction for every example. One classifier's output is therefore a class probability distribution for every example.

Our algorithm can be expressed as in the following pseudo code (Figure 3.2) and

```
1. Select A Classification Algorithm
2. Select Another Classification Algorithm
3. Input The Data
4. Split Data To 10 Folds Randomly
5. Train The Data using the proposed algorithm
6. Use 9 Folds For Training
7. Keep The  10 Th Fold For Testing
8. Testing Phase
9. Do Testing Using The Tenth Fold
10. Repeat The Training and Testing Until All Folds are used
11. Take the average of results obtained in each iteration
12. Repeat the process from Step 2 if there are more algorithms to combine
```

Figure 3.2 Training and Testing in classifier combination using DROC

the same algorithm can be expressed  with the flow chart in Figure 3.3.

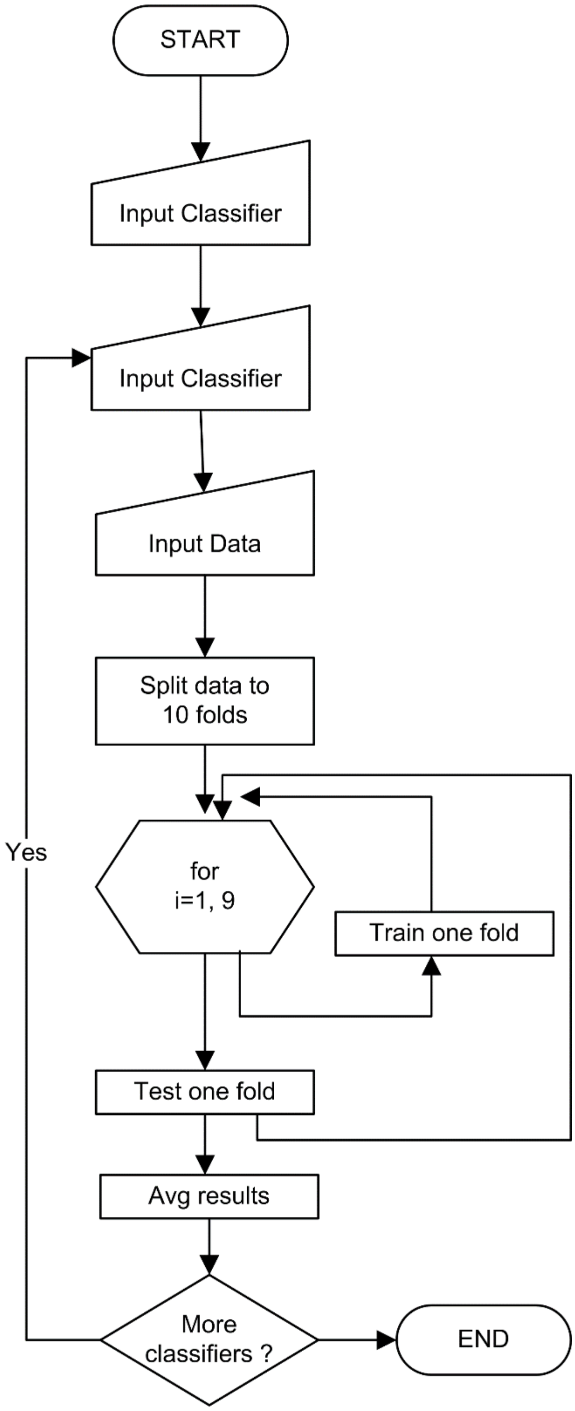


Figure 3.3 Flow chart for training and testing in classifier combination using DROC

Dempster's Rule of Combination performs combination in a pairwise fashion. In other words if one needs to combine more than two algorithms using Dempster's Rule of Combination, first he needs to combine two algorithms and then using the result of this combination he combines the third algorithm and so on.

Combining classifiers in a pairwise fashion is achieved using the algorithm in Figure 3.4.

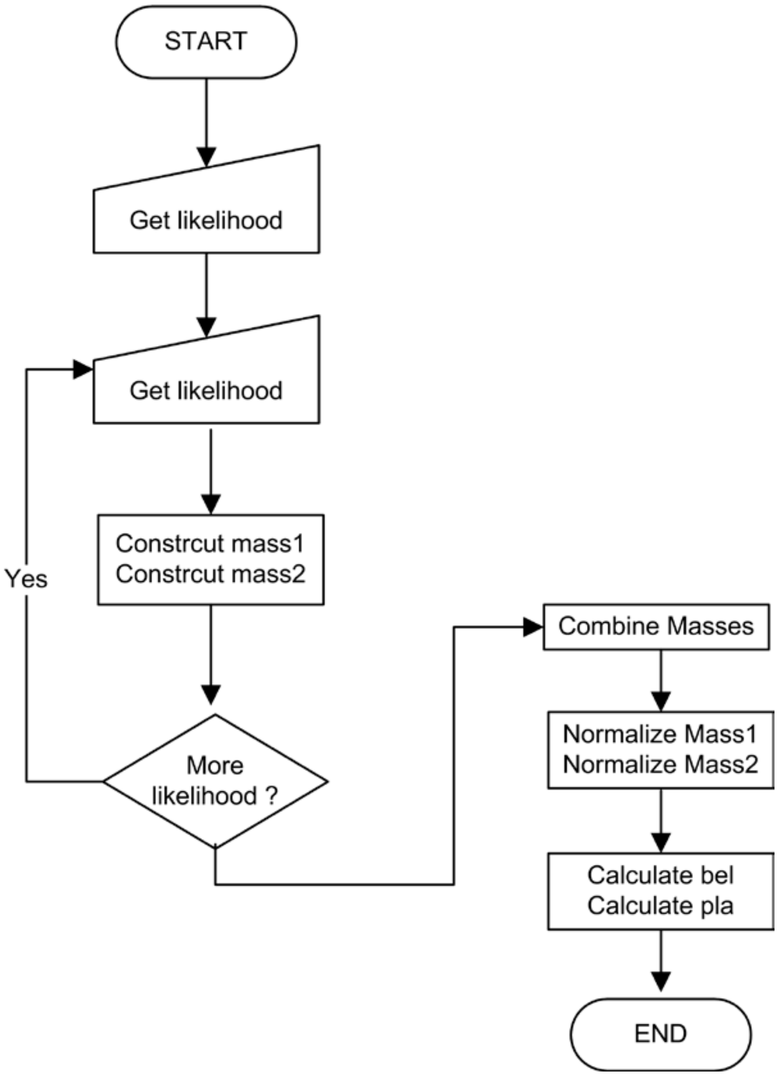


**Figure 3.4 Flow chart for for general framework for classifier combination using Dempster's Rule of Combination**

The algorithm in Figure 3.5 shows how the mass functions are obtained from likelihood values. Combining masses is given in Figure 3.6. Normalization process is presented in Figure 3.7. Belief and Plausability Calculation in DROC is displayed in Figure 3.8.

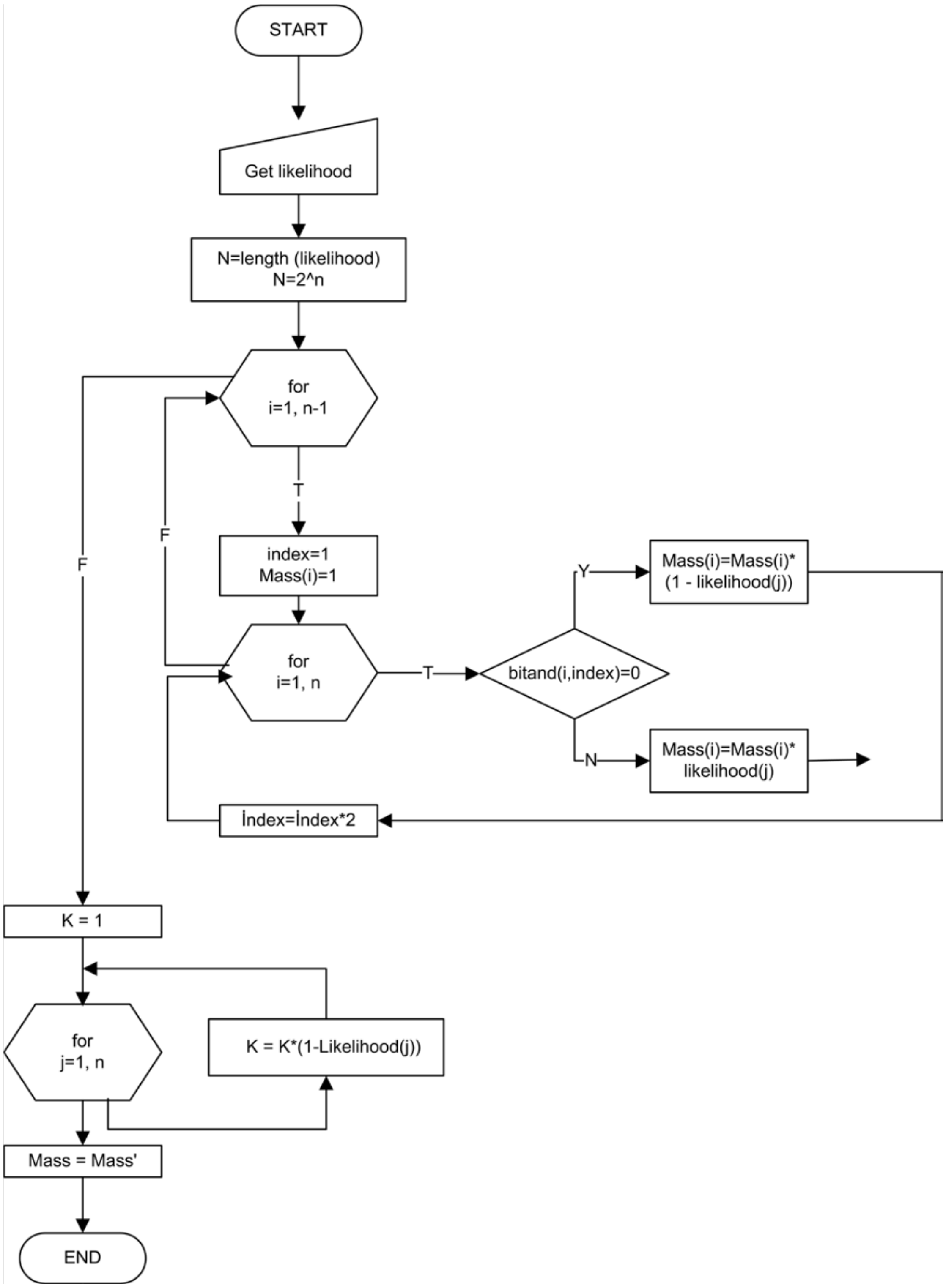


Figure 3.5 Flow Chart for Obtaining mass function from likelihood

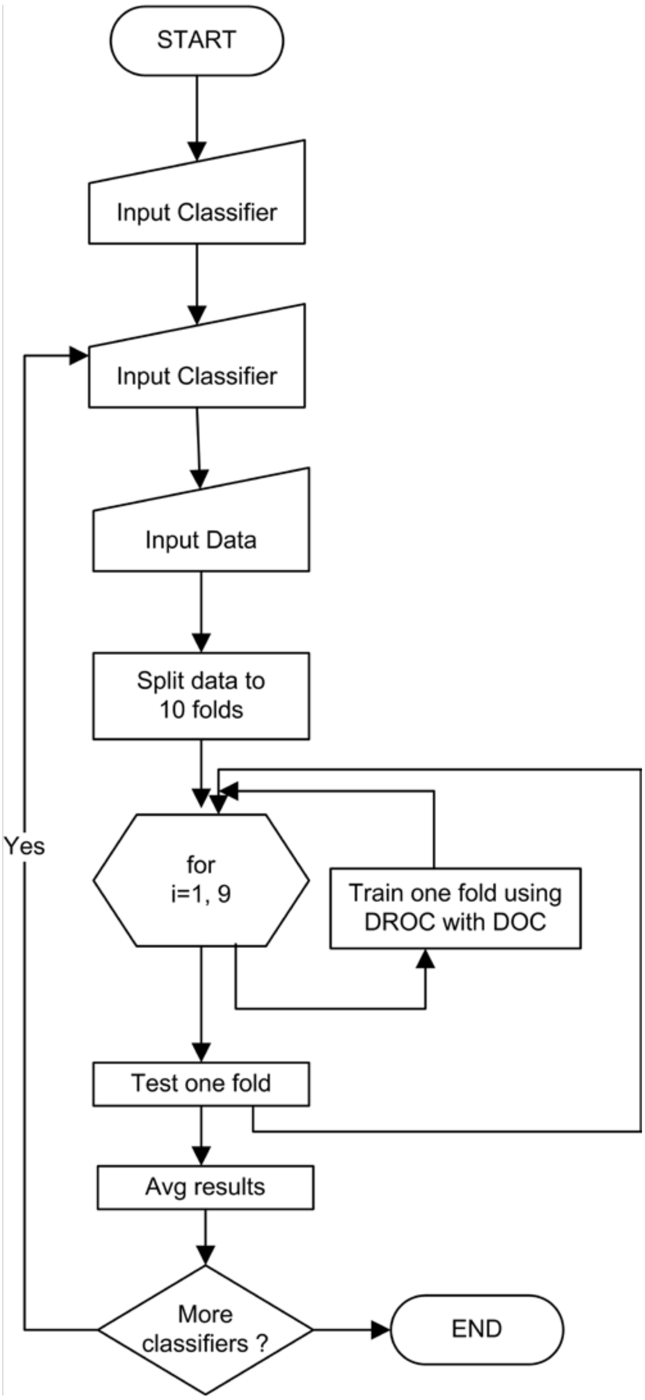


**Figure 3.11 Flow chart for training and testing in classifier combination using Demspter's Rule of Combination  with Degree of Confidence**

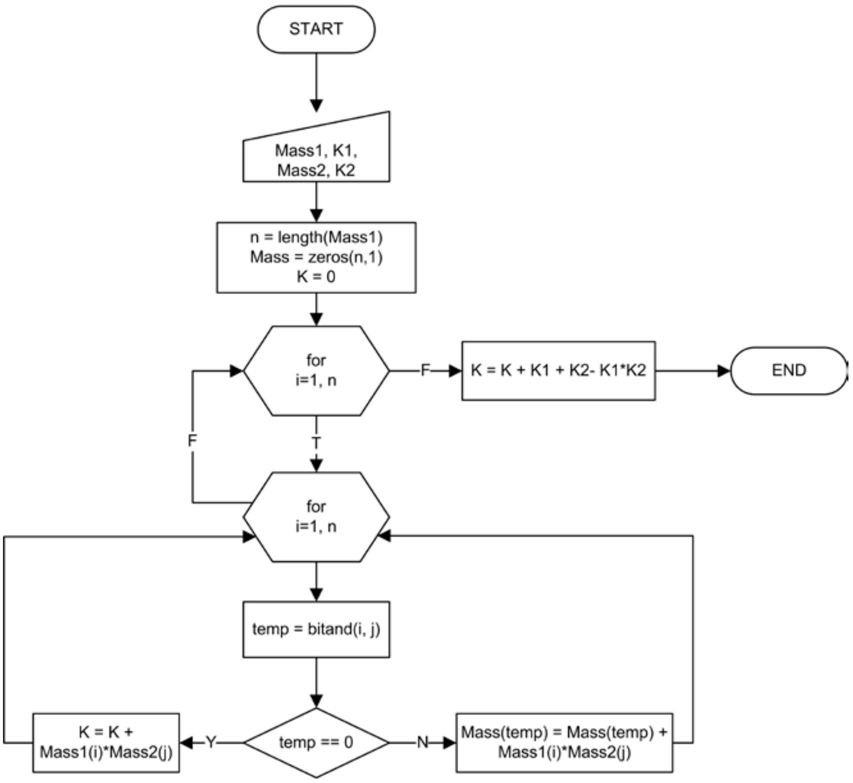


**Figure 3.6 Flow Chart for Combining Masses in DROC**

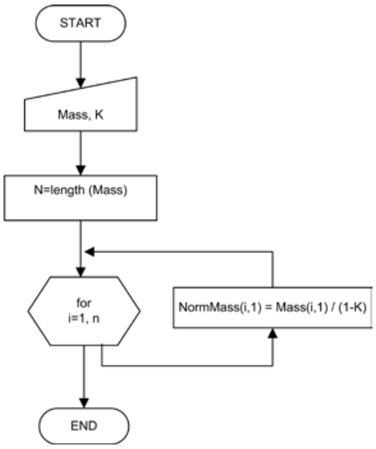


**Figure 3.7 Flow Chart for Normalization in DROC**

## 3. 5 Employing Degree of Confidence in the Combination

Every classification algorithms includes uncertainty to some extent. Uncertainty is the difference between the belief and plausibility values. As the difference gets smaller the uncertainty decreases and as the difference gets larger the uncertainty increases. Little amount of uncertainty yields to a more precise decision.

In this study we also employ degree of confidence of the classifiers to achieve more concrete classification results. First of all, let us explain uncertainty with an example and then present our proposed method for uncertainty management.

**Example:**

Let us consider the swimming pool example that we have given in section 3.4. Another important issue in this problem is the uncertainty of the decision. In our problem the interval [bel, pla] is [0.379, 0.491] and the absolute value which shows the uncertainty is

$$|bel - pla| = 0.112.$$

Let us solve the same problem from the beginning taking into account the reliability of the sensors. Let us suppose that DOC$_1$, the degree of confidence of Sensor$_1$, is 0.70 and that DOC$_2$, the degree of confidence of Sensor$_2$, is 0.80. In this case Mass, Belief and Plausibility values will be as follows:

Mass

| | |
|---|---|
| m({a}) | = 0.40201745 |
| m({b}) | = 0.35295264 |
| m({a,b}) | = 0.03956587 |
| m({c}) | = 0.16819937 |
| m({a,c} ) | = 0.01885509 |
| m({b,c}) | = 0.01655389 |
| m({a,b,c}) | = 0.00185569 |

bel

| | |
|---|---|
| bel({a}) | = 0.40201745 (Decision= Goose) |
| bel({b}) | = 0.35295264 |
| bel({a,b}) | = 0.79453596 |
| bel({c}) | = 0.16819937 |
| bel({a,c}) | = 0.58907192 |
| bel({b,c}) | = 0.53770591 |
| bel({a,b,c}) | = 1.00000000 |

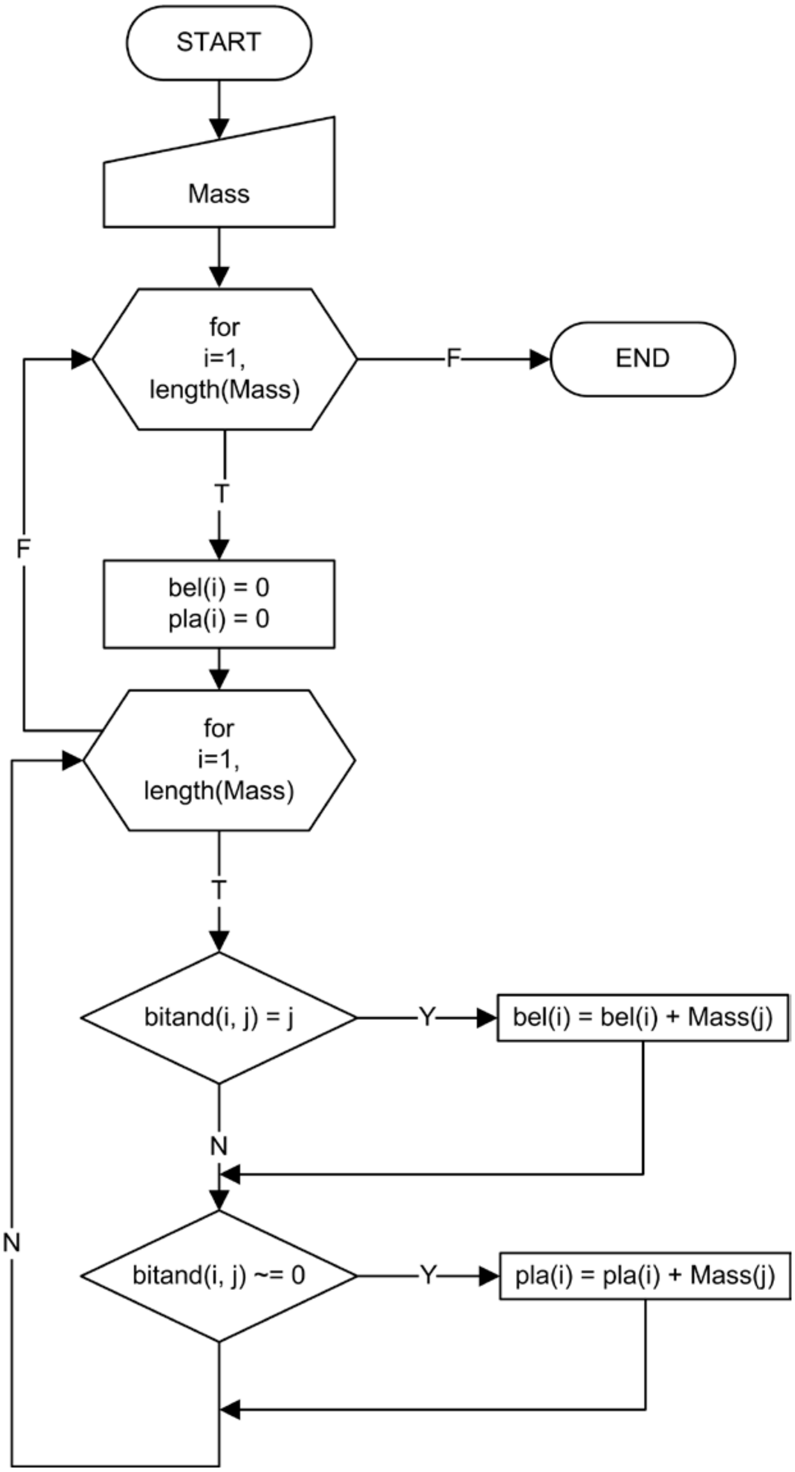


Figure 3.8 Flow Chart for Belief and Plausability Calculation in DROC

pla

| | |
|---|---|
| pla({a}) | = 0.46229409 |
| pla({b}) | = 0.41092808 |
| pla({a,b}) | = 0.83180063 |
| pla({c}) | = 0.20546404 |
| pla({a,c}) | = 0.64704736 |
| pla({b,c}) | = 0.59798255 |
| pla({a,b,c}) | = 1.00000000 |

The decision will be Goose with green head again. The interval [bel, pla] will be [0.402, 0.462] and the uncertainty |bel-pla| = 0.060. As seen here, when the reliability of the sensors is taken into account the uncertainty decreases. The uncertainty decreases by

0.112 - 0.060 = 0.052.

When evidences are combined using Dempster's Rule of Combination generally values which are over a predefined value are taken into consideration. In our example if the Belief step value is chosen as 0.40 then bel({a})=0.402 will be taken into consideration. If the Belief value were 0.37975015, the value calculated without using the degree of confidence of the sensors, then it would be below the predefined value and so it would not be taken into consideration for judgment.

The degree of confidence or the confidence factor is in fact the average success rate of the classification algorithm that it has displayed in the past. Considering the results of each classifier as beliefs we calculate new beliefs by employing the confidence factor of the classifiers. Mathematically speaking; let likelihood vector be $\mathcal{L}$, and degree of confidence be $\mathcal{C}$ then the new likelihood vector $\mathcal{L}_{new}$ including the results of classifiers is given by;

$$\mathcal{L}_{new}[a_{1,j}] = \mathcal{L}[a_{1,j}] \star \mathcal{C} \qquad (3.13)$$

In the proposed method, we first perform classification with different classification algorithms. We then calculate the degree of confidence for each classifier. Afterwards, assuming the results of the classifiers as beliefs, we calculate mass functions for each classifier. We then combine the mass values in a pairwise fashion using the Dempster's

Rule of Combination. Finally we calculate belief and plausibility values. The flow diagram of the proposed method is shown in Figure 3.9.

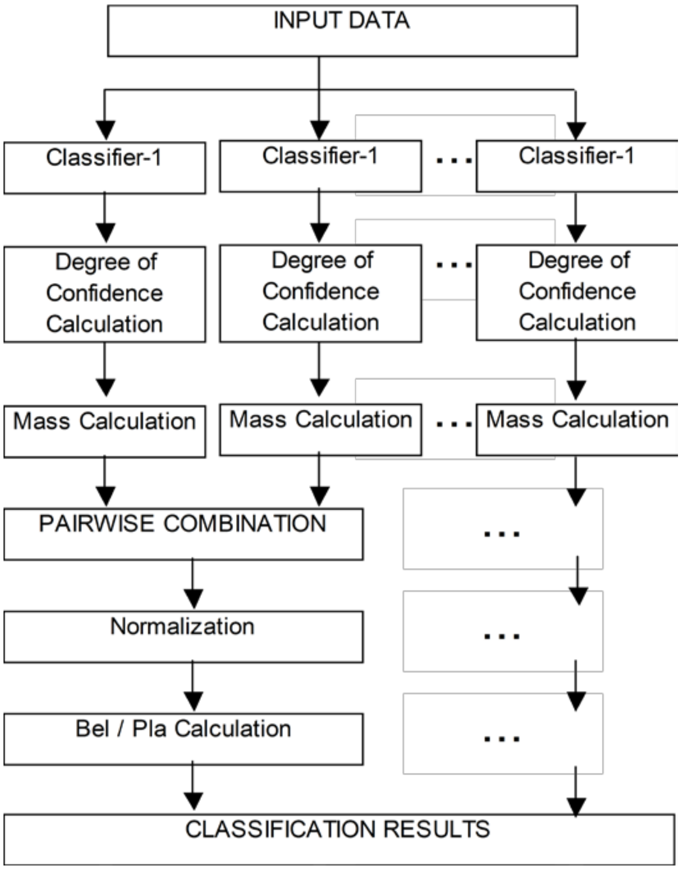


Figure 3.9 Combining Classifiers using Dempster's Rule of Combination with Degree of Confidence Added

As we do in Se ed method. The ps ie al er combination using DROC with degree of confidence is shown in Figure 3.12

1. Select A Classification Algorithm
2. Select Another Classification Algorithm
3. Input The Data
4. Split Data To 10 Folds Randomly
5. Train The Data using Dempster's Rule of Combination with degree of confidence
6. Use 9 Folds For Training
7. Keep The 10 Th Fold For Testing
8. Testing Phase
9. Do Testing Using The Tenth Fold
10. Repeat The Training and Testing Until All Folds are used
11. Take the average of results obtained in each iteration
12. Repeat the prosess from Step 2 if there are more algorithms to combine

Figure 3.10 Training and Testing in classifier combination using DSROC with degree of confidence

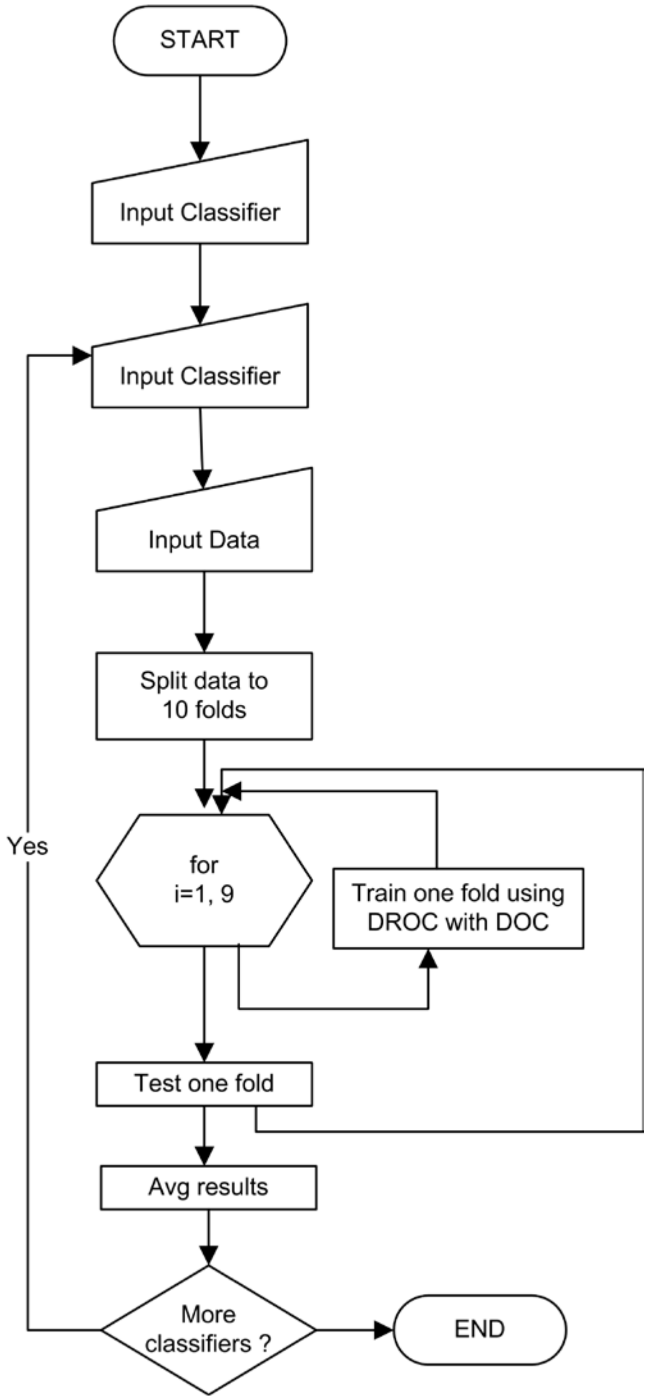


**Figure 3.11 Flow chart for training and testing in classifier combination using Demspter's Rule of Combination with Degree of Confidence**

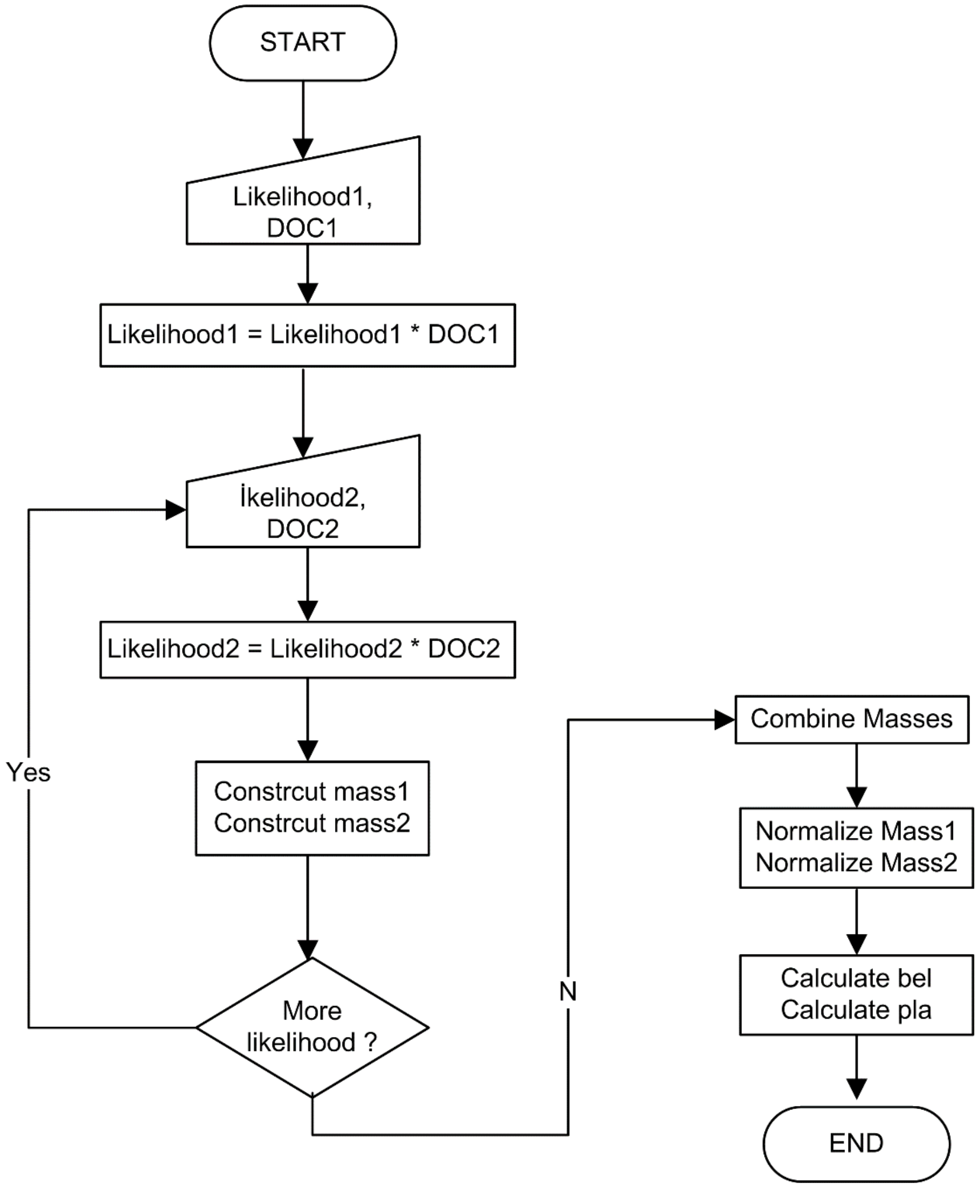


**Figure 3.12 Flow chart for General framework for classifier combination using DROC with degree of confidence**

# 4 EXPERIMENTS

We perform classification on 10 different data sets taken from UCI Machine Learning Repository (Murphy and Alia, 1994). The data sets used in our experiments are shown in Table 4.1.

## 4.1 Data Sets Used in the Experiments

**Table 4.1** Data sets used in the experiments

| Dataset | Examples | Attributes |
|---|---|---|
| Autos | 205 | 26 |
| Breast-Cancer-Wisconsin | 699 | 11 |
| Heart-Disease-Cleveland | 303 | 14 |
| Heart-Disease-Hungary | 294 | 14 |
| Hepatitis | 155 | 20 |
| Iris | 150 | 5 |
| Labor | 57 | 17 |
| Soybean | 683 | 36 |
| Thyroid | 215 | 6 |
| Wine | 178 | 14 |

The characteristics of the data sets used in our experiments are given in Appendix A.

## 4.2 Success of WEKA classifiers on UCI data sets

In our experiments we use WEKA classification algorithms from Witten and Frank (2000) and Murphy and Alia (1994). In the experiments we use the algorithms with their default values.

The success of the WEKA classification methods on the data sets we use are given in the following subsections:

### 4.2.1 Classifiers of Baye’s Group

Classifiers of Bayes Group consists of the classifiers Bayes Net, Naïve Bayes , and Naïve Bayes Updateable. All these three algorithms are based on Baye’s theorem. The success of Baye’s Group algoritms are shown in Table 4.2. The performance of each of the algorithm is explained in the the following paragraphs.

**Table 4.2** Success of WEKA classifiers of Bayes group on UCI data sets (%)

| Dataset | Bayes Net | Naïve Bayes | Naïve Bayes Updateable |
|---|---|---|---|
| Autos | 81.47 | 79.51 | 83.90 |
| Breast-Cancer-Wisconsin | 97.28 | 97.42 | 97.56 |
| Heart-Disease-Cleveland | 54.78 | 56.43 | 56.43 |
| Heart-Disease-Hungary | 82.65 | 85.35 | 85.71 |
| Hepatitis | 71.61 | 70.32 | 69.67 |
| Iris | 92.66 | 96 | 95.33 |
| Labor | 87.71 | 89.47 | 94.73 |
| Soybean | 93.26 | 92.97 | 92.67 |
| Thyroid | 94.41 | 96.74 | 96.27 |
| Wine | 98.87 | 96.62 | 97.19 |
| AVERAGE | 85.47 | 86.08 | 86.94 |

Bayes Net has a success rate of 98.87 on Wine data while it has the worst performance on Heart-Disease-Cleveland (54.78%). Bayes Net is also good on Breast-Cancer-Wisconsin with a success rate of 97.28. It is not good on Hepatitis data set (71.61%).

Naïve Bayes is most succesful on Breast-Cancer-Wisconsin data (97.42%) while it is worst on Heart-Disease-Cleveland data (56.43%). Naïve Bayes has also got a success rate of over 95% at the data sets Thyroid, Wine, and Iris. It is not good on Hepatitis data set (70.32%).

Naïve Bayes Updateable is best on Breast-Cancer-Wisconsin data (97.42%) while it is worst on Heart-Disease-Cleveland data (56.43%). It is also good at Iris, Thyroid and Wine data sets as Naïve Bayes Simple.

The most successful algorithm among the classification algorithms of Baye’s group is Bayes Net with a rate of 98.87% on Wine data.

### 4.2.2 Classifiers of Lazy Group

The algorithms of Lazy group are IB1, IBK, Kstar, and LWL. Short explanation about the classifiers are given in the following paragraphs.

IB1 is a nearest-neighbour classifier. It uses normalized Euclidean distance to find the training instance closest to the given test instance, and predicts the same class as this training instance. If multiple instances have the same (smallest) distance to the test instance, the first one found is used. More information can be obtained from Aha and Kibler (1991) about instance based classifiers.

IBk is K-nearest neighbours classifier. This classifier Normalizes attributes by default. It can select appropriate value of K based on cross-validation. It can also do distance weighting. For more information, see Aha and Kibler (1991).

Kstar is an instance-based classifier, that is the class of a test instance is based upon the class of those training instances similar to it, as determined by some similarity function. It differs from other instance-based learners in that it uses an entropy-based distance function. For more information on Kstar, see John et al. (1995).

LWL is the locally weighted version of Naïve Bayes that relaxes the independence assumption by learning local models at prediction time (Frank et al. 2003). The main advantage of this method compared to other techniques for enhancing Naïve Bayes is its conceptual and computational simplicity.

The success of these algorithms are given in Table 4.3. and the explanation regarding the success of each of the algorithm are presented in the following paragraphs.

IB1 has a success rate of 97.20 on Thyroid data (the most successful) while it has a success rate of 54.45 (the worst) on Heart-Disease-Cleveland data. The algorithm is also good on Iris, Breast-Cancer-Wisconsin , Wine, and Autos data sets. Its success on Heart-Disease-Hungary and Hepatitis are not good.

The classification algorithm IBK is the general form of the algorithm IB1 so it has the same performance on all the data sets.

**Table 4.3** Success of WEKA classifiers of Lazy group on UCI data sets (%)

| Dataset | IB1 | IBK | Kstar | LWL |
|---|---|---|---|---|
| Autos | 94.15 | 94.15 | 94.63 | 85.37 |
| Breast-Cancer-Wisconsin | 95 | 94.70 | 75.67 | 94.13 |
| Heart-Disease-Cleveland | 54.45 | 54.45 | 51.48 | 58.08 |
| Heart-Disease-Hungary | 59.18 | 59.18 | 78.57 | 80.95 |
| Hepatitis | 66.45 | 66.45 | 61.93 | 54.19 |
| Iris | 95.33 | 95.33 | 94.66 | 93.33 |
| Labor | 82.45 | 82.45 | 89.47 | 85.96 |
| Soybean | 89.89 | 89.89 | 87.99 | 57.97 |
| Thyroid | 97.20 | 97.20 | 95.34 | 90.23 |
| Wine | 94.94 | 94.94 | 98.87 | 89.32 |
| AVERAGE | 82.90 | 82.87 | 82.86 | 78.95 |

Kstar is best on Wine data set with a success rate of 98.87. It ıs worst on Heart-Disease-Cleveland with a ratio of 51.48. Its performance on Thyroid, Hepatitis and Autos data sets are over 94%. It is not good on Hepatitis data (61.93%).

The classifier LWL is the most successful with a rate of 94.13 on Breast-Cancer-Wisconsin data while it is worst with a rate of 58.08 on Heart-Disease-Cleveland data. It is also good on Iris data (93.33%) and Thyroid data (90.23%).

The most successful algorithm among the four algortihms of Lazy group is Kstar with a success rate of  98.87% on Wine data.

### 4.2.3 Classifiers of Tree Group

Tree group algorithms is WEKA are Decision Stump, J48, LMT, NB Tree, and REPTree. These algorithms are shown in Table 4.4.

Decision Stump is a very simple learner which learns a decision stump. A decision stump is a decision tree with only one split.

J48 is the class for generating a pruned or unpruned C4.5 decision tree. C4.5 is the extension of the basic ID3 algorithm designed by Quinlan to address the following issues which is not dealt with by ID3:

- Avoiding overfitting the data
- Determining how deeply to grow a decision tree.
- Reduced error pruning.
- Rule post-pruning.
- Handling continuous attributes. e.g., temperature
- Choosing an appropriate attribute selection measure.
- Handling training data with missing attribute values.
- Handling attributes with differing costs.
- Improving computational efficiency.

More information about C4.5 decision tree can be found at Quinlan (1993).

LMT is the classifier for building 'logistic model trees', which are classification trees with logistic regression functions at the leaves. More information can be found at Landwehr et al. (2003).

NBTree is a classifier for generating a decision tree with naive Bayes classifiers at the leaves. More information can be found in Kohavi (1996).

Reduced Error Pruning Tree (REPTree) is a fast decision tree learner. It builds a decision/regression tree using information gain/variance and prunes it using reduced-error pruning (with backfitting). The algorithm only sorts values for numeric attributes once. Missing values are dealt with by splitting the corresponding instances into pieces as in C4.5.

The success of Tree group algorithms are shown in Table 4.4. The success of each of the algorithms is given in the following paragraphs.

Decision Stump is the most successful on Breast-Cancer-Wisconsin data with a success rate of 88.26%. It is the worst on Soybean with a rate of 27.96%. It is not very good on the other data sets.

**Table 4.4** Success of WEKA classifiers of Tree group on UCI data sets (%)

| Dataset | Decision Stump | J48 | LMT | NB Tree | REPTree |
|---|---|---|---|---|---|
| Autos | 80 | 89.75 | 95.12 | 90.24 | 89.75 |
| Breast-Cancer-Wisconsin | 88.26 | 94.85 | 95.27 | 96.56 | 94.13 |
| Heart-Disease-Cleveland | 51.48 | 52.47 | 55.77 | 55.44 | 56.76 |
| Heart-Disease-Hungary | 80.27 | 78.57 | 86.73 | 83.67 | 78.23 |
| Hepatitis | 61.29 | 58.70 | 63.87 | 67.74 | 60 |
| Iris | 66.66 | 96 | 94 | 92.66 | 94 |
| Labor | 80.70 | 73.68 | 89.47 | 87.71 | 77.19 |
| Soybean | 27.96 | 91.50 | 93.55 | 91.50 | 84.33 |
| Thyroid | 77.20 | 92.09 | 97.67 | 93.02 | 92.09 |
| Wine | 57.86 | 93.82 | 97.19 | 96.06 | 94.38 |
| AVERAGE | 67.16 | 82.14 | 86.86 | 85.46 | 82.08 |

J48 is best on Breast-Cancer-Wisconsin with a rate of 94.85 while it is worst on Heart-Disease-Cleveland with a rate of 52.47.

LMT is the most successful on Thyroid data with a success rate of 97.67%. It is the worst on Heart-Disease-Cleveland data with a rate of 55.77%. It is also very succesful on Wine, Autos, Breast-Cancer-Wisconsin and Soybean data sets.

NB Tree is best on Breast-Cancer-Wisconsin with a rate of 96.56% while it is worst on Heart-Disease-Cleveland with a rate of 55.44%. It is also good on Autos, Iris, Soybean, Thyroid, and Wine data sets.

REPTree is the most successful on Wine data with a success rate of 94.38%. It is the worst on Heart-Disease-Cleveland data with a rate of 56.76%. It is also very succesful on Breast-Cancer-Wisconsin, Iris and Thyroid data sets.

The most successful algorithm among the classification algorithms of Tree group is LMT with a rate of 97.67% on Thyroid data.

### 4.2.4 Classifiers of Rules Group

RULES group algorithms is WEKA are Conjunctive Rule, Decision Table, Nnge, OneR, PART, Ridor, Jrip and ZeroR.

Conjunctive Rule classifier implements a single conjunctive rule learner that can predict for numeric and nominal class labels. A rule consists of antecedents "AND"ed together and the consequent (class value) for the classification/regression. In this case, the consequent is the distribution of the available classes (or mean for a numeric value) in the dataset. If the test instance is not covered by this rule, then it's predicted using the default class distributions/value of the data not covered by the rule in the training data.This learner selects an antecedent by computing the Information Gain of each antecendent and prunes the generated rule using Reduced Error Prunning (REP) or simple pre-pruning based on the number of antecedents.

For classification, the Information of one antecedent is the weighted average of the entropies of both the data covered and not covered by the rule.

For regression, the Information is the weighted average of the mean-squared errors of both the data covered and not covered by the rule.

In pruning, weighted average of the accuracy rates on the pruning data is used for classification while the weighted average of the mean-squared errors on the pruning data is used for regression.

Decision Table is a classifier for building and using a simple decision table. More information can be found in Kohavi (1995).

Nnge is nearest-neighbor-like algorithm using non-nested generalized exemplars which are hyperrectangles that can be viewed as if-then rules. More information can be obtained from Martin (1995).

OneR is a classifier for building and using a 1R classifier. In other words, it uses the minimum-error attribute for prediction and discretizes numeric attributes. Holte (1993) can be checked for more information.

PART generates a PART decision list. It uses separate-and-conquer method. It builds a partial C4.5 decision tree in each iteration and makes the "best" leaf into a rule. More information can be received from Frank and Witten (1998).

Ridor is the implementation of a RIpple-DOwn Rule learner. It generates a default rule first and then the exceptions for the default rule with the least (weighted) error rate. Then it generates the "best" exceptions for each exception and iterates until pure. Thus it performs a tree-like expansion of exceptions.The exceptions are a set of rules that predict classes other than the default.

Jrip implements a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER), which was proposed as an optimized version of IREP. More information can be found in Cohen (1995).

ZeroR builds and uses a 0-R classifier. Predicts the mean for a numeric class or the mode for a nominal class.

The success of these algorithms is shown in Table 4.5 and the success of each algorithm is explained in the following paragraphs.

Conjunctive Rule is the most successful on Breast-Cancer-Wisconsin data with a success rate of 87.83%. It is the worst on Soybean with a rate of 26.20%. It is not very good on the other data sets.

Decision Table is best on Iris data set (92.66%) and it worst on Heart-Disease-Cleveland (56.43%). It has a success rate of over 90% on Breast-Cancer-Wisconsin, Thyroid and Wine data sets.

NNge is the most successful on Breast-Cancer-Wisconsin data with a success rate of 95.85%. It is the worst on Heart-Disease-Cleveland with a rate of 51.48%. It is also very good on Soybean, Thyroid and Wine data sets.

OneR is best on Iris data set (94.0%) and it worst on Soybean (39.97%). It has a good success rate on Breast-Cancer-Wisconsin, and Thyroid data sets.

PART is the most successful on Breast-Cancer-Wisconsin data with a success rate of 94.27%. It is the worst on Heart-Disease-Cleveland with a rate of 55.44%. It is also very good on Iris, Soybean, Thyroid and Wine data sets.

Ridor is best on Iris data set (94%) and it worst on Heart-Disease-Cleveland (54.78%). It has a success rate of over 91% on Breast-Cancer-Wisconsin, Thyroid and Wine data sets.

**Table 4.5** Success of WEKA classifiers of RULES group on UCI data sets (%)

| Dataset | Conjunctive Rule | Decision Table | NNge | OneR | PART | Ridor | Jrip | ZeroR |
|---|---|---|---|---|---|---|---|---|
| Autos | 76.58 | 94.63 | 91.21 | 83.41 | 89.75 | 87.31 | 89.26 | 72.19 |
| Breast-Cancer-Wisconsin | 87.83 | 90.27 | 95.85 | 92.70 | 94.27 | 92.70 | 93.70 | 65.52 |
| Heart-Disease-Cleveland | 55.11 | 56.43 | 51.48 | 52.47 | 55.44 | 54.78 | 54.45 | 54.12 |
| Heart-Disease-Hungary | 76.87 | 80.27 | 78.23 | 78.91 | 79.59 | 79.25 | 78.91 | 63.94 |
| Hepatitis | 60 | 61.93 | 62.58 | 61.93 | 64.51 | 60.64 | 63.22 | 54.83 |
| Iris | 66.66 | 92.66 | 96 | 94 | 94 | 94 | 94.66 | 33.33 |
| Labor | 77.19 | 77.19 | 77.19 | 75.43 | 78.94 | 80.70 | 77.19 | 64.91 |
| Soybean | 26.20 | 86.96 | 91.80 | 39.97 | 91.94 | 89.31 | 92.53 | 13.47 |
| Thyroid | 77.67 | 92.09 | 95.81 | 91.16 | 93.95 | 92.55 | 93.95 | 69.76 |
| Wine | 63.48 | 91.01 | 97.75 | 76.40 | 93.25 | 91.01 | 92.13 | 93.13 |
| AVERAGE | 66.75 | 82.34 | 83.79 | 74.63 | 83.56 | 82.22 | 83 | 58.52 |

Similarly Jrip is best on Iris data set (94.66%) and it worst on Heart-Disease-Cleveland (54.45%). It has a success rate of over 92% on Breast-Cancer-Wisconsin, Soybean, Thyroid and Wine data sets.

ZeroR is the most successful on Wine data with a success rate of 93.13%. It is the worst on Soebean with a rate of 13.47%. It is not very good on the other data sets.

The most successful algorithm among the classification algorithms of Rules group is NNge with a rate of 95.85% on Breast-Cancer-Wisconsin data.

### 4.2.5 Classifiers of Functions Group

WEKA classifiers of function group are Logistic, Multilayer Perceptron, RBF Network, SMO, and Simple Logistic. Classifiers of function group are shown in Table 4.6. A short explanation about each of the algorithms is given in the following paragraphs.

Logistic is the classifier for building and using a multinomial logistic regression model with a ridge estimator. If there are k classes for n instances with m attributes, the parameter matrix B to be calculated will be an m*(k-1) matrix.

Although original Logistic Regression does not deal with instance weights the algorithm is modified a little bit to handle the instance weights. More information can be obtained from Cessie and Houwelingen (1992).

Multilayer Perceptron is a neural network using backpropagation.

RBF Network is a classifier which implements a radial basis function network. It uses the K-Means clustering algorithm to provide the basis functions and learns either a logistic regression (discrete class problems) or linear regression (numeric class problems) on top of that.

SMO implements the sequential minimal optimization algorithm in Platt (1998) for training a support vector classifier. This implementation globally replaces all missing values and transforms nominal attributes into binary ones. It also normalizes all attributes by default Multi-class problems are solved using pairwise classification.More information on the SMO algorithm can be found in Platt (1998) and Keerthi et al. (2001).

. **Table 4.6** Success of WEKA classifiers of FUNCTIONS group on UCI data sets (%)

| Dataset | Logistic | Multilayer Perceptron | RBF Network | SMO | Simple Logistic |
|---|---|---|---|---|---|
| Autos | 95.60 | 95.12 | 71.70 | 95.12 | 95.12 |
| Breast-Cancer-Wisconsin | 93.13 | 95.85 | 91.41 | 95.70 | 95.27 |
| Heart-Disease-Cleveland | 59.73 | 53.46 | 55.44 | 59.07 | 55.77 |
| Heart-Disease-Hungary | 85.03 | 78.91 | 64.96 | 81.63 | 86.73 |
| Hepatitis | 82.58 | 81.93 | 54.19 | 67.74 | 63.22 |
| Iris | 96 | 97.33 | 66.66 | 96 | 94 |
| Labor | 92.98 | 85.96 | 66.66 | 89.47 | 89.47 |
| Soybean | 93.22 | 93.41 | 34.84 | 93.85 | 93.55 |
| Thyroid | 96.74 | 96.74 | 91.62 | 89.76 | 97.67 |
| Wine | 97.19 | 97.19 | 74.71 | 98.31 | 97.19 |
| AVERAGE | 89.22 | 87.59 | 67.21 | 86.66 | 86.79 |

Simple Logistic is the classifier for building linear logistic regression models. LogitBoost with simple regression functions as base learners is used for fitting the logistic models. The optimal number of LogitBoost iterations to perform is cross-validated, which leads to automatic attribute selection. More information can be found in Landwehr et al. (2003).

As seen in Table 4.6, Logistic is the most successful on Wine data with a success rate of 97.19%. It is the worst on Heart-Disease-Cleveland with a rate of 59.73%. It has a success rate of over 92% on Autos, Breast-Cancer-Wisconsin, Iris, Labor, Soybean and Thyroid data sets.

Multilayer Perceptron is best on Iris data set (97.33%) and it worst on Heart-Disease-Cleveland (53.46%). It has a success rate of over 93% on Autos, Breast-Cancer-Wisconsin, Soybean, Thyroid and Wine data sets.

RBF Network is the most successful on Thyroid data with a success rate of 91.62%. It is the worst on Soybean with a rate of 34.84%. It is also good at a Breast-Cancer-Wisconsin data.

SMO is best on Wine data set (98.31%) and it worst on Heart-Disease-Cleveland (59.07%). It has a success rate of over 93% on Autos, Breast-Cancer-Wisconsin, Iris, Soybean and wine data sets.

Simple Logistic is the most successful on Thyroid data with a success rate of 97.67%. It is the worst on Heart-Disease-Cleveland with a rate of 55.77 It has a success rate of over 92% on Autos, Breast-Cancer-Wisconsin, Iris and Soybean data sets.

The most successful algorithm among the classification algorithms of functions group is SMO with a rate of 98.31% on Wine data.

## 4.3 Success of WEKA Hybrid Classifiers on UCI Data Sets

WEKA hybrid classifiers are given below. These classifiers are proposed and built into WEKA by different researches

- AdaBoostM1
- AttributeSelectedClassifier
- Bagging
- Classification Via Regression
- CV Parameter Selection
- Decorate
- Grading
- LogitBoost

- MultiBoostAB
- MulticlassClassifier
- MultiScheme
- OrdinalClassClassifier
- RacedIncrementalLogitBoost
- Stacking
- StackingC
- Vote

The details of the above algorithms are given in Chapter 2. The modified versions of some of the hybrid algorithms implemented in WEKA or the ones not mentioned in Chapter 2 are briefly explained in the following paragraphs.

In AttributeSelectedClassifier dimensionality of training and test data is reduced by attribute selection before being passed on to a classifier.

Classification Via Regression does classification using regression methods. Class is binarized and one regression model is built for each class value. More information can be obtained from Frank et al. (1998).

CV Parameter Selection performs parameter selection by cross-validation for any classifier. More information can be obtained from Kohavi (1995).

Decorate is a meta-learner for building diverse ensembles of classifiers by using specially constructed artificial training examples. Comprehensive experiments have demonstrated that this technique is consistently more accurate than the base classifiers, Bagging and Random Forests. Decorate also obtains higher accuracy than Boosting on small training sets, and achieves comparable performance on larger training sets. More details can be found in Melville and Mooney (2003).

LogitBoost performs additive logistic regression. It performs classification using a regression scheme as the base learner, and can handle multi-class problems. It can do efficient internal cross-validation to determine appropriate number of iterations. More details can be found in Friedman et al. (1998).

MultiBoostAB boosts a classifier using the MultiBoosting method. MultiBoosting is an extension to the highly successful AdaBoost technique for forming decision committees. MultiBoosting can be viewed as combining AdaBoost with wagging. It is able to harness both AdaBoost's high bias and variance reduction with wagging's superior variance reduction. Using C4.5 as the base learning algorithm, Multi-boosting is demonstrated to produce decision committees with lower error than either AdaBoost or wagging significantly more often than the reverse over a large representative cross-section of UCI data sets. It offers the further advantage over AdaBoost of suiting parallel execution. More information can be obtained from Geoffrey (2000).

MulticlassClassifier is  metaclassifier for handling multi-class datasets with 2-class classifiers. This classifier is also capable of applying error correcting output codes for increased accuracy.

MultiScheme selects a classifier from among several using cross validation on the training data or the performance on the training data. Performance is measured based on percent correct (classification) or mean-squared error (regression).

OrdinalClassClassifier is a meta classifier that allows standard classification algorithms to be applied to ordinal class problems. More information can be obtained from Frank and Hall (2001).

RacedIncrementalLogitBoost is a classifier for incremental learning of large datasets by way of racing logit-boosted committees.

StackingC implements more efficient version of stacking. More information can be obtained from Seewald (2002).

Vote is a classifier for combining classifiers using unweighted average of probability estimates (classification) or numeric predictions (regression).

In the experiments performed using the current hybrid algorithms, the default values of the hybrid classifiers in WEKA are used. The success of hybrid classifiers AdaBoostM1, AttributeSelectedClassifier, Bagging,  Classification Via Regression, CV Parameter Selection and Decorate is shown in Table 4.7. The success of each of these algorithms is explained in the following paragraphs.

As seen in Table 4.7, AdaBoostM1 is the most successful on Iris data with a success rate of 95.33%.  It is the worst Soybean with a rate of 27.96%.  It has a success rate of over 91% on  Breast-Cancer-Wisconsin, Thyroid and Wine data sets.

AttributeSelectedClassifier is best on Autos data set (72.19%) and it worst on Soybean (13.47%).  It is not very good on the other data sets.

Bagging is the most successful on Wine data with a success rate of 94.94%.  It is the worst Heart-Disease-Cleveland with a rate of 56.43%.  It has a success rate of over 90% on  Autos, Iris, and Thyroid data sets.

Classification Via Regression is best on Wine data set (97.75%) and it worst on Heart-Disease-Cleveland (58.08%).  It has a success rate of over 92% on  Autos, Breast-Cancer-Wisconsin , Iris, Soybean and Thyroid data sets.

CVParameter Selection, like AttributeSelectedClassifier, is the most successful on Autos data with a success rate of 72.19%. It is the worst Soybean with a rate of 13.47%. It is not very good on the other data sets. The reason why the classifiers CVParameter Selection and AttributeSelectedClassifier have the same success rates is because they use the same default base classifier, namely ZeroR.

Decorate is best on Wine data set (98.31%) and it worst on Heart-Disease-Cleveland (53.13%). It has a success rate of over 91% on Autos, Breast-Cancer-Wisconsin, Iris, Labor, Soybean and Thyroid data sets.

The success of hybrid classifiers Grading, LogitBoost, MultiBoostAB, MulticlassClassifier, MultiScheme, and OrdinalClassClassifier is shown in Table 4.8. The success of each of these algorithms is explained in the following paragraphs.

The hybrid classifiers Grading, MulticlassClassifier, MultiScheme, and OrdinalClassClassifier have the same success rates on the data sets since they all use the same default base classifier ZeroR. They are the most successful on Autos data with a success rate of 72.19%. They are the least successful on Soybean with a rate of 13.47%. They are not very successful on the other data sets.

**Table 4.7** Success of WEKA Hybrid classifiers on UCI data sets –1 (%)

| Dataset | AdaBoostM1 | AttributeSelectedClassifier (with ZeroR) | Bagging | Classification ViaRegression | CVParameter Selection (with ZeroR) | Decorate |
|---|---|---|---|---|---|---|
| Autos | 80 | 72.19 | 90.73 | 93.65 | 72.19 | 95.60 |
| Breast-Cancer-Wisconsin | 94.99 | 65.52 | 95.70 | 95.99 | 65.52 | 96.70 |
| Heart-Disease-Cleveland | 51.48 | 54.12 | 56.43 | 58.08 | 54.12 | 53.13 |
| Heart-Disease-Hungary | 80.27 | 63.94 | 79.93 | 79.93 | 63.94 | 80.27 |
| Hepatitis | 61.29 | 54.83 | 60.64 | 61.29 | 54.83 | 65.80 |
| Iris | 95.33 | 33.33 | 94.00 | 94.00 | 33.33 | 96.00 |
| Labor | 87.71 | 64.91 | 85.96 | 82.45 | 64.91 | 91.22 |
| Soybean | 27.96 | 13.47 | 86.82 | 92.97 | 13.47 | 94.28 |
| Thyroid | 93.48 | 69.76 | 93.48 | 93.95 | 69.76 | 97.20 |
| Wine | 91.57 | 39.88 | 94.94 | 97.75 | 39.88 | 98.31 |
| AVERAGE | 76.40 | 53.19 | 83.86 | 85.00 | 53.19 | 86.85 |

MultiBoostAB is best on Iris data set (95.33%) and it worst on Soybean (27.96%). It has a success rate of over 91% on Breast-Cancer-Wisconsin, Thyroid and Wine data sets.

The success of hybrid classifiers RacedIncrementalLogitBoost, Stacking, StackingC and Vote is shown in Table 4.9. They all have the same success rates on the data sets since they all use the same default base classifier ZeroR. They are the most successful on Autos data with a success rate of 72.19%. They are the least successful on Soybean with a rate of 13.47%. They are not very successful on the other data sets.

### 4.4 Results of Combining Classifiers Using Dempster's Rule of Combination

Assuming the classifier outputs as beliefs, we combine the results obtained from the existing classification algorithms using Dempster's Rule of Combination. For our experiments we have chosen four representative algorithms from the WEKA group of algorithms that we have explained in Section 4.2. These algorithms are Naïve Bayes, IB1, j48, and OneR.

Naïve Bayes is the representative for the Baye's group algorithms; IB1 is the representative for the Lazy group algorithms; j48 for the Tree group and OneR for the Rules group. We have not chosen any algorithm from the WEKA functions group due to the time complexity problems.

We combine the four algorithms in the following manner:

- Naïve Bayes and IB1
- Naïve Bayes and J48
- Naïve Bayes and OneR
- IB1 and J48
- IB1 and OneR
- J48 and OneR
- Naïve Bayes, IB1 and J48

**Table 4.8** Success of WEKA Hybrid classifiers on UCI data sets-2 (%)

| Dataset | Grading (with ZeroR) | LogitBoost | MultiBoostAB | MulticlassClassifier (with ZeroR) | MultiScheme (with ZeroR) | OrdinalClassClassifier (with ZeroR) |
|---|---|---|---|---|---|---|
| Autos | 72.19 | 96.58 | 80 | 72.19 | 72.19 | 72.19 |
| Breast-Cancer-Wisconsin | 65.52 | 95.99 | 94.99 | 65.52 | 65.52 | 65.52 |
| Heart-Disease-Cleveland | 54.12 | 59.07 | 51.48 | 54.12 | 54.12 | 54.12 |
| Heart-Disease-Hungary | 63.94 | 80.61 | 80.27 | 63.94 | 63.94 | 63.94 |
| Hepatitis | 54.83 | 61.29 | 61.29 | 54.83 | 54.83 | 54.83 |
| Iris | 33.33 | 94.00 | 95.33 | 33.33 | 33.33 | 33.33 |
| Labor | 64.91 | 89.47 | 87.71 | 64.91 | 64.91 | 64.91 |
| Soybean | 13.47 | 92.97 | 27.96 | 13.47 | 13.47 | 13.47 |
| Thyroid | 69.76 | 95.81 | 93.48 | 69.76 | 69.76 | 69.76 |
| Wine | 39.88 | 98.31 | 91.57 | 39.88 | 39.88 | 39.88 |
| AVERAGE | 53.19 | 86.41 | 76.40 | 53.19 | 53.19 | 53.19 |

**Table 4.9** Success of WEKA Hybrid classifiers on UCI data sets-3 (%)

| Dataset | RacedIncrementalLogitBoost (with ZeroR) | Stacking (with ZeroR) | StackingC (with ZeroR) | Vote (with ZeroR) |
|---|---|---|---|---|
| Autos | 72.19 | 72.19 | 72.19 | 72.19 |
| Breast-Cancer-Wisconsin | 65.52 | 65.52 | 65.52 | 65.52 |
| Heart-Disease-Cleveland | 54.12 | 54.12 | 54.12 | 54.12 |
| Heart-Disease-Hungary | 63.94 | 63.94 | 63.94 | 63.94 |
| Hepatitis | 54.83 | 54.83 | 54.83 | 54.83 |
| Iris | 33.33 | 33.33 | 33.33 | 33.33 |
| Labor | 64.91 | 64.91 | 64.91 | 64.91 |
| Soybean | 13.47 | 13.47 | 13.47 | 13.47 |
| Thyroid | 69.76 | 69.76 | 69.76 | 69.76 |
| Wine | 39.88 | 39.88 | 39.88 | 39.88 |
| AVERAGE | 53.19 | 53.19 | 53.19 | 53.19 |

- Naïve Bayes, IB1 and OneR
- Naïve Bayes, J48 and OneR
- IB1, J48 and OneR

The result of combining the two algorithms Naïve Bayes and IB1 using Demspter's Rule of Combination is shown in Table 4.10. The table shows the success of each of the algorithms used in the combination, the combination result of Naïve Bayes and IB1 algorithms and the amount of uncertainty in the combination process. As seen from Table 4.10, the combined result of Naïve Bayes and IB1 has a an average success rate of success of 91.68 on all the data sets. The success of the combination is over 99% on Breast-Cancer-Wisconsin, Iris, Thyroid and Wine data sets. The combination has a success rate of 54.37 on Heart-Disease-Cleveland data set.

**Table 4.10** Combination Result of Naïve Bayes and IB1 using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | IB1 | Naïve Bayes + IB1 | Uncertainty |
|---|---|---|---|---|
| Autos | 79.51 | 94.15 | 98.31 | 1.25 |
| Breast-Cancer-Wisconsin | 97.42 | 95 | 99.83 | 0.15 |
| Heart-Disease-Cleveland | 56.43 | 54.45 | 54.37 | 13.80 |
| Heart-Disease-Hungary | 85.35 | 59.18 | 88.44 | 5.79 |
| Hepatitis | 70.32 | 66.45 | 80.26 | 9.12 |
| Iris | 96 | 95.33 | 99.78 | 0.2 |
| Labor | 89.47 | 82.45 | 97.30 | 1.97 |
| Soybean | 92.97 | 89.89 | 98.92 | 0.88 |
| Thyroid | 96.74 | 97.20 | 99.87 | 0.12 |
| Wine | 96.62 | 94.94 | 99.73 | 0.24 |
| AVERAGE | 86.08 | 82.90 | 91.68 | 3.35 |

Examining Table 4.10 shows that uncertainty is highest (13.80%) for Heart-Disease-Cleveland data, and lowest (0.12) for Thyroid data. Uncertainty is over 1% for Hepatitis, Heart-Disease-Hungary, Labor and Autos data sets.

On the average, combining the two algorithms Naïve Bayes and IB1 using Demspter's Rule of Combination results in an increase of 5.6% against Naïve Bayes and an increase of 8.78% against IB1 algorithm.

The result of combining the two algorithms Naïve Bayes and J48 using Demspter's Rule of Combination is given in Table 4.11. As in Table 4.11, the table shows the success of each of the algorithms used in the combination, the success of the combined result of Naïve Bayes and IB1 algorithms and the amount of uncertainty in the combination process. The combined result of Naïve Bayes and J48 has an average success rate of 91.12% on all the data sets. The success of the combination

**Table 4.11** Combination Result of Naïve Bayes and J48 using Demspter's Rule of Combination

| Dataset | Naïve Bayes | J48 | Naïve Bayes + J48 | Uncertainty |
|---|---|---|---|---|
| Autos | 79.51 | 89.75 | 96.74 | 2.29 |
| Breast-Cancer-Wisconsin | 97.42 | 94.85 | 99.80 | 0.18 |
| Heart-Disease-Cleveland | 56.43 | 52.47 | 52.09 | 13.95 |
| Heart-Disease-Hungary | 85.35 | 78.57 | 95.10 | 3.25 |
| Hepatitis | 70.32 | 58.70 | 73.79 | 10.64 |
| Iris | 96 | 96 | 99.82 | 0.16 |
| Labor | 89.47 | 73.68 | 95.50 | 2.92 |
| Soybean | 92.97 | 91.50 | 99.14 | 0.72 |
| Thyroid | 96.74 | 92.09 | 99.63 | 0.32 |
| Wine | 96.62 | 93.82 | 99.68 | 0.28 |
| AVERAGE | 86.08 | 82.14 | 91.12 | 3.47 |

is over 99% on Breast-Cancer-Wisconsin, Iris, Soybean, Thyroid and Wine data sets while it has a success rate of 52.09 on Heart-Disease-Cleveland data set.

Examining Table 4.11 shows that uncertainty is highest (13.95%) for Heart-Disease-Cleveland data, and lowest (0.16) for Iris data. Uncertainty is over 1% for Hepatitis, Heart-Disease-Hungary, Labor and Autos data sets. The average uncertainty for the combination on all the data sets is 3.47%.

The average success rate of the classifier Naïve Bayes is 86.08% and it is 82.14% for the algorithm J48. The average success rate of the combined result is 91.12%. Combining the two algorithms Naïve Bayes and J48 using Demspter's Rule of Combination adds, on the average, a increase of 5.04% against Naïve Bayes and an increase of 8.9% against J48 algorithm.

The result of combining the two algorithms Naïve Bayes and OneR using Demspter's Rule of Combination is given in Table 4.12. The table shows the success of each of the algorithms used in the combination, the success of the combined result of Naïve Bayes and OneR algorithms and the amount of uncertainty in the combination process. The combined result of Naïve Bayes and OneR has an average success rate of 90.14% on all the data sets. The success of the combination is over 99% on Breast-Cancer-Wisconsin, Iris, and Thyroid data sets while it has a success rate of 52.09 on Heart-Disease-Cleveland data set.

Examining Table 4.12 shows that uncertainty is highest (13.95%) for Heart-Disease-Cleveland data, and lowest (0.16) for Breast-Cancer-Wisconsin and Iris data sets. Uncertainty is over 1% for Hepatitis, Heart-Disease-Hungary, Labor, Soybean and Autos data sets. The average uncertainty for the combination on all the data sets is 3.83%.

The average success rate of the classifier Naïve Bayes is 86.08% and it is 74.63% for the algorithm OneR. The average success rate of the combined result is 90.14%. Combining the two algorithms Naïve Bayes and OneR using Demspter's Rule of Combination adds, on the average, a increase of 4.06% against Naïve Bayes and an increase of 15.51% against OneR algorithm.

The result of combining the two algorithms IB1 and J48 using Demspter's Rule of Combination is given in Table 4.13. The table shows the success of each of the algorithms used in the combination, the success of the combined result of IB1 and J48 algorithms and the amount of uncertainty in the combination process. The combined result of IB1 and J48 has an average success rate of 89.19% on all the

**Table 4.12** Combination Result of Naïve Bayes and OneR using Demspter's Rule of Combination

| Dataset | Naïve Bayes | OneR | Naïve Bayes + OneR | Uncertainty |
|---|---|---|---|---|
| Autos | 79.51 | 83.41 | 94.65 | 3.51 |
| Breast-Cancer-Wisconsin | 97.42 | 92.70 | 99.73 | 0.24 |
| Heart-Disease-Cleveland | 56.43 | 52.47 | 52.09 | 13.95 |
| Heart-Disease-Hungary | 85.35 | 78.91 | 97.30 | 1.79 |
| Hepatitis | 70.32 | 61.93 | 76.31 | 10.12 |
| Iris | 96 | 94 | 99.73 | 0.24 |
| Labor | 89.47 | 75.43 | 95.93 | 2.71 |
| Soybean | 92.97 | 39.97 | 87.49 | 4.48 |
| Thyroid | 96.74 | 91.16 | 99.58 | 0.36 |
| Wine | 96.62 | 76.40 | 98.68 | 0.96 |
| AVERAGE | 86.08 | 74.63 | 90.14 | 3.83 |

data sets. The success of the combination is over 99% on Autos, Breast-Cancer-Wisconsin, Iris, Thyroid and Wine data sets while it has a success rate of 51.18 on Heart-Disease-Cleveland data set.

Table 4.13 shows that uncertainty is highest (13.71%) for Heart-Disease-Cleveland data, and lowest (0.20) for Iris data. Uncertainty is over 1% for Heart-Disease-Hungary, Hepatitis, and Labor data sets. The average uncertainty for the combination on all the data sets is 4.08%.

The average success rate of the classifier IB1 is 82.90% and it is 82.14% for the algorithm J48. The average success rate of the combined result is 89.19%. Combining the two algorithms IB1 and J48 using Demspter's Rule of Combination adds, on the average, a increase of 6.29% against IB1 and an increase of 7.05% against J48 algorithm.

**Table 4.13** Combination Result of IB1 and J48 using Demspter's Rule of Combination

| Dataset | IB1 | J48 | IB1 + J48 | Uncertainty |
|---|---|---|---|---|
| Autos | 94.15 | 89.75 | 99.21 | 0.49 |
| Breast-Cancer-Wisconsin | 95 | 94.85 | 99.66 | 0.30 |
| Heart-Disease-Cleveland | 54.45 | 52.47 | 51.18 | 13.71 |
| Heart-Disease-Hungary | 59.18 | 78.57 | 82.27 | 8.16 |
| Hepatitis | 66.45 | 58.70 | 69.67 | 11.61 |
| Iris | 95.33 | 96 | 99.78 | 0.20 |
| Labor | 82.45 | 73.68 | 92.13 | 4.71 |
| Soybean | 89.89 | 91.50 | 98.78 | 0.98 |
| Thyroid | 97.20 | 92.09 | 99.73 | 0.24 |
| Wine | 94.94 | 93.82 | 99.51 | 0.42 |
| AVERAGE | 82.90 | 82.14 | 89.19 | 4.08 |

The result of combining the two algorithms IB1 and OneR using Demspter's Rule of Combination is given in Table 4.14. The combined result of IB1 and OneR has an average success rate of 88.43% on all the data sets. The success of the combination is over 99% on Breast-Cancer-Wisconsin, Iris, and Thyroid data sets while it has a success rate of 49.77 on Heart-Disease-Cleveland data set.

Table 4.14 shows that uncertainty is highest (14.1%) for Heart-Disease-Cleveland data, and lowest (0.27) for Thyroid data. Uncertainty is over 1% for Autos, Heart-Disease-Hungary, Hepatitis, Labor, Soybean and Wine data sets. The average uncertainty for the combination on all the data sets is 4.48%.

The average success rate of the classifier IB1 is 82.90% and it is 74.63% for the algorithm OneR. The average success rate of the combined result is 88.43%. Combining the two algorithms IB1 and OneR using Demspter's Rule of Combination adds, on the average, a increase of 5.53% against IB1 and an increase of 13.8% against OneR algorithm.

**Table 4.14** Combination Result of IB1 and OneR using Demspter's Rule of Combination

| Dataset | IB1 | OneR | IB1 + OneR | Uncertainty |
|---|---|---|---|---|
| Autos | 94.15 | 83.41 | 98.69 | 1.02 |
| Breast-Cancer-Wisconsin | 95 | 92.70 | 99.54 | 0.40 |
| Heart-Disease-Cleveland | 54.45 | 52.47 | 49.77 | 14.1 |
| Heart-Disease-Hungary | 59.18 | 78.91 | 82.27 | 8.16 |
| Hepatitis | 66.45 | 61.93 | 81.07 | 8.87 |
| Iris | 95.33 | 94 | 99.66 | 0.30 |
| Labor | 82.45 | 75.43 | 92.88 | 4.38 |
| Soybean | 89.89 | 39.97 | 82.83 | 5.96 |
| Thyroid | 97.20 | 91.16 | 99.69 | 0.27 |
| Wine | 94.94 | 76.40 | 97.99 | 1.43 |
| AVERAGE | 82.90 | 74.63 | 88.43 | 4.48 |

The result of combining the two algorithms J48 and OneR using Demspter's Rule of Combination is given in Table 4.15. The combined result of J48 and OneR has an average success rate of 87.17% on all the data sets. The success of the combination is over 99% on Breast-Cancer-Wisconsin, Iris, and Thyroid data sets while it has a success rate of 47.45 on Heart-Disease-Cleveland data set.

Table 4.15 shows that uncertainty is highest (14.21%) for Heart-Disease-Cleveland data, and lowest (0.24) for Iris data. Uncertainty is over 1% for Autos, Heart-Disease-Hungary, Hepatitis, Labor, Soybean and Wine data sets. The average uncertainty for the combination on all the data sets is 4.78%.

The average success rate of the classifier J48 is 82.14% and it is 74.63% for the algorithm OneR. The average success rate of the combined result is 87.17%. Combining the two algorithms J48 and OneR using Demspter's Rule of Combination adds, on the average, a increase of 5% against J48 and an increase of 12.5% against OneR algorithm.

**Table 4.15** Combination Result of J48 and OneR using Demspter's Rule of Combination

| Dataset | J48 | OneR | J48 + OneR | Uncertainty |
|---|---|---|---|---|
| Autos | 89.75 | 83.41 | 97.48 | 1.86 |
| Breast-Cancer-Wisconsin | 94.85 | 92.70 | 99.44 | 0.48 |
| Heart-Disease-Cleveland | 52.47 | 52.47 | 47.45 | 14.21 |
| Heart-Disease-Hungary | 78.57 | 78.91 | 92.28 | 4.69 |
| Hepatitis | 58.70 | 61.93 | 64.36 | 12.61 |
| Iris | 96 | 94 | 99.73 | 0.24 |
| Labor | 73.68 | 75.43 | 88.32 | 6.39 |
| Soybean | 91.50 | 39.97 | 85.93 | 4.99 |
| Thyroid | 92.09 | 91.16 | 99.14 | 0.72 |
| Wine | 93.82 | 76.40 | 97.63 | 1.67 |
| AVERAGE | 82.14 | 74.63 | 87.17 | 4.78 |

The result of combining the three algorithms Naïve Bayes, IB1and J48 using Demspter's Rule of Combination is given in Table 4.16. As discussed in Chapter 3 Demspter's Rule of Combination performs combination in a pairwise fashion. The triple combination is performed in the following way: First the classifiers Naïve Bayes and IB1 are combined and afterwards the combined result is combined with the classifier J48.

Examining Table 4.16 shows that Naïve Bayes has an average success rate of 86.08

**Table 4.16** Combination Result of Naïve Bayes, IB1 and J48 using Demspter's Rule of Combination

| Dataset | Naïve Bayes | IB1 | Naïve Bayes + IB1 | J48 | Naïve Bayes + IB1 + J48 | Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 79.51 | 94.15 | 98.31 | 89.75 | 99.74 | 0.24 |
| Breast-Cancer-Wisconsin | 97.42 | 95 | 99.83 | 94.85 | 99.93 | 0.07 |
| Heart-Disease-Cleveland | 56.43 | 54.45 | 54.37 | 52.47 | 49.77 | 14.1 |
| Heart-Disease-Hungary | 85.35 | 59.18 | 88.44 | 78.57 | 96.19 | 2.61 |
| Hepatitis | 70.32 | 66.45 | 80.26 | 58.70 | 83.49 | 7.66 |
| Iris | 96 | 95.33 | 99.78 | 96 | 99.95 | 0.04 |
| Labor | 89.47 | 82.45 | 97.30 | 73.68 | 98.85 | 0.81 |
| Soybean | 92.97 | 89.89 | 98.92 | 91.50 | 99.79 | 0.18 |
| Thyroid | 96.74 | 97.20 | 99.87 | 92.09 | 99.91 | 0.08 |
| Wine | 96.62 | 94.94 | 99.73 | 93.82 | 99.92 | 0.07 |
| AVERAGE | 86.08 | 82.90 | 91.68 | 82.14 | 92.75 | 4.36 |

on all the data sets. In the meantime the average rate for IB1 is 82.90% and 82.14% for J48. The rate for the combination Naïve Bayes and IB1 is 91.68 on the average and finally the average success rate for the triple combination Naïve Bayes, IB1 and J48 is 92.75.

The triple combination Naïve Bayes, IB1 and J48 has a success rate of over 99% on Autos, Breast-Cancer-Wisconsin, Iris, Soybean, Thyroid and Wine data sets while it has a success rate of 49.77 on Heart-Disease-Cleveland data set.

Table 4.16 shows that, for the triple combination, uncertainty is highest (14.1%) for Heart-Disease-Cleveland data, and lowest (0.04%) for Iris data. Uncertainty is over 1% for Heart-Disease-Hungary, and Hepatitis data sets.

From the above explanations we see that combining the three algorithms Naïve Bayes, IB1 and J48 in a pairwise fashion using Demspter's Rule of Combination adds, on the average, a increase of 6.6% against Naïve Bayes, an increase of 9.8% against IB1 algorithm and an increase of 10.6% against J48 algorithm. The triple combination is also 1% more successful, on the average, than the combination Naïve Bayes and IB1.

The result of combining the three algorithms Naïve Bayes, IB1and OneR using Demspter's Rule of Combination is given in Table 4.17. As discussed in Chapter 3 Demspter's Rule of Combination performs combination in a pairwise fashion. The triple combination is performed in the following way: First the classifiers Naïve Bayes and IB1 are combined and afterwards the combined result is combined with the classifier OneR.

Examining Table 4.17 shows that Naïve Bayes has an average success rate of 86.08 on all the data sets. In the meantime the average rate for IB1 is 82.90% and 74.63% for J48. The rate for the combination Naïve Bayes and IB1 is 91.68 on the average and finally the average success rate for the triple combination Naïve Bayes, IB1 and OneR is 92.60.

The triple combination Naïve Bayes, IB1 and OneR has a success rate of over 99% on Autos, Breast-Cancer-Wisconsin, Iris, Thyroid and Wine data sets while it has a success rate of 49.77 on Heart-Disease-Cleveland data set.

Table 4.17 shows that, for the triple combination, uncertainty is highest (14.1%) for Heart-Disease-Cleveland data, and lowest (0.06%) for Iris data. Uncertainty is over 1% for Heart-Disease-Hungary, Soybean and Hepatitis data sets.

**Table 4.17** Combination Result of Naïve Bayes, IB1 and OneR using Demspter's Rule of Combination

| Dataset | Naïve Bayes | IB1 | Naïve Bayes + IB1 | OneR | Naïve Bayes + IB1 + OneR | Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 79.51 | 94.15 | 98.31 | 83.41 | 99.58 | 0.41 |
| Breast-Cancer-Wisconsin | 97.42 | 95 | 99.83 | 92.70 | 99.91 | 0.08 |
| Heart-Disease-Cleveland | 56.43 | 54.45 | 54.37 | 52.47 | 49.77 | 14.1 |
| Heart-Disease-Hungary | 85.35 | 59.18 | 88.44 | 78.91 | 96.19 | 2.61 |
| Hepatitis | 70.32 | 66.45 | 80.26 | 61.93 | 85.22 | 7.21 |
| Iris | 96 | 95.33 | 99.78 | 94 | 99.93 | 0.06 |
| Labor | 89.47 | 82.45 | 97.30 | 75.43 | 98.97 | 0.74 |
| Soybean | 92.97 | 89.89 | 98.92 | 39.97 | 96.86 | 1.2 |
| Thyroid | 96.74 | 97.20 | 99.87 | 91.16 | 99.90 | 0.09 |
| Wine | 96.62 | 94.94 | 99.73 | 76.40 | 99.68 | 0.24 |
| AVERAGE | 86.08 | 82.90 | 91.68 | 74.63 | 92.60 | 2.67 |

From the above explanations we see that combining the three algorithms Naïve Bayes, IB1 and OneR in a pairwise fashion using Demspter's Rule of Combination adds, on the average, a increase of 6.5% against Naïve Bayes, an increase of  9.7% against IB1 algorithm and an increase of  17.9% against OneR algorithm. The triple combination is also about 1% more successful, on the average, than the combination  Naïve Bayes and IB1.

The result of combining the three algorithms Naïve Bayes, J48 and OneR using Demspter's Rule of Combination is given in Table 4.18.  As discussed in Chapter 3 Demspter's Rule of Combination performs combination in a pairwise fashion. The triple combination is performed in the following way: First the classifiers Naïve Bayes and J48 are combined and afterwards the combined result is combined with the classifier OneR.

Examining Table 4.18 shows that Naïve Bayes has an average success rate of 86.08 on all the data sets. In the meantime the average rate for J48 is 82.14% and 74.64% for OneR. The rate for the combination Naïve Bayes and J48 is 91.13 on the average and finally the average success rate for the triple combination Naïve Bayes, J48 and OneR is 92.45.

The triple combination Naïve Bayes, J48 and OneR has a success rate of over 99% on Autos, Breast-Cancer-Wisconsin, Iris, Thyroid and Wine data sets while it has a success rate of  47.45 on Heart-Disease-Cleveland data set.

Table 4.18 shows that, for the triple combination, uncertainty is highest (14.21%) for Heart-Disease-Cleveland data, and lowest (0.06%) for  Iris data. Uncertainty is over 1% for Heart-Disease-Hungary, Hepatitis, and Labor data sets. From the above explanations we see that combining the three algorithms Naïve Bayes, J48 and OneR in a pairwise fashion using Demspter's Rule of Combination adds, on the average, a increase of 6.37% against Naïve Bayes, an increase of  10.31% against J48 algorithm and an increase of  17.81% against OneR algorithm. The triple combination is also about 1.3% more successful, on the average, than the combination  Naïve Bayes and J48.

The result of combining the three algorithms IB1, J48 and OneR using Demspter's Rule of Combination is given in Table 4.19.  As discussed in Chapter 3 Demspter's Rule of Combination performs combination in a pairwise fashion. The triple combination is performed in the following way: First the classifiers IB1 and J48 are combined and afterwards the combined result is combined with the classifier OneR.

**Table 4.18** Combination Result of Naïve Bayes, J48 and OneR using Demspter's Rule of Combination

| Dataset | Naïve Bayes | J48 | Naïve Bayes + J48 | OneR | Naïve Bayes + J48 + OneR | Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 79.51 | 89.75 | 96.74 | 83.41 | 99.14 | 0.68 |
| Breast-Cancer-Wisconsin | 97.42 | 94.85 | 99.80 | 92.70 | 99.91 | 0.08 |
| Heart-Disease-Cleveland | 56.43 | 52.47 | 52.09 | 52.47 | 47.45 | 14.21 |
| Heart-Disease-Hungary | 85.35 | 78.57 | 95.10 | 78.91 | 98.52 | 1.09 |
| Hepatitis | 70.32 | 58.70 | 73.79 | 61.93 | 83.33 | 8.15 |
| Iris | 96 | 96 | 99.82 | 94 | 99.93 | 0.06 |
| Labor | 89.47 | 73.68 | 95.50 | 75.43 | 98.25 | 1.24 |
| Soybean | 92.97 | 91.50 | 99.14 | 39.97 | 98.43 | 0.60 |
| Thyroid | 96.74 | 92.09 | 99.63 | 91.16 | 99.90 | 0.09 |
| Wine | 96.62 | 93.82 | 99.68 | 76.40 | 99.68 | 0.24 |
| AVERAGE | 86.08 | 82.14 | 91.13 | 74.64 | 92.45 | 2.64 |

**Table 4.19** Combination Result of IB1, J48 and OneR using Demspter's Rule of Combination

| Dataset | IB1 | J48 | IB1 + J48 | OneR | IB1 + J48 + OneR | Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 94.15 | 89.75 | 99.21 | 83.41 | 99.79 | 0.17 |
| Breast-Cancer-Wisconsin | 95 | 94.85 | 99.66 | 92.70 | 99.91 | 0.08 |
| Heart-Disease-Cleveland | 54.45 | 52.47 | 51.18 | 52.47 | 46.30 | 14.24 |
| Heart-Disease-Hungary | 59.18 | 78.57 | 82.27 | 78.91 | 93.94 | 3.87 |
| Hepatitis | 66.45 | 58.70 | 69.67 | 61.93 | 75.37 | 10.36 |
| Iris | 95.33 | 96 | 99.78 | 94 | 99.93 | 0.06 |
| Labor | 82.45 | 73.68 | 92.13 | 75.43 | 97.12 | 1.98 |
| Soybean | 89.89 | 91.50 | 98.78 | 39.97 | 96.86 | 1.2 |
| Thyroid | 97.20 | 92.09 | 99.73 | 91.16 | 99.90 | 0.09 |
| Wine | 94.94 | 93.82 | 99.51 | 76.40 | 99.68 | 0.24 |
| AVERAGE | 82.90 | 82.14 | 89.19 | 74.64 | 90.88 | 3.23 |

Examining Table 4.19 shows that IB1 has an average success rate of 82.90 on all the data sets. In the meantime the average rate for J48 is 82.14% and 74.64% for OneR. The rate for the combination IB1 and J48 is 89.19 on the average and finally the average success rate for the triple combination IB1, J48 and OneR is 90.88.

The triple combination IB1, J48 and OneR has a success rate of over 99% on Autos, Iris, Thyroid and Wine data sets while it has a success rate of 46.30 on Heart-Disease-Cleveland data set.

Table 4.19 shows that, for the triple combination, uncertainty is highest (14.24%) for Heart-Disease-Cleveland data, and lowest (0.06%) for Iris data. Uncertainty is over 1% for Heart-Disease-Hungary, Hepatitis, Labor and Soybean data sets. From the above explanations we see that combining the three algorithms IB1, J48 and OneR in a pairwise fashion using Demspter's Rule of Combination adds, on the average, a increase of 7.9% against IB1, an increase of 8.7% against J48 algorithm and an increase of 16.2% against OneR algorithm. The triple combination is also about 1.6% more successful, on the average, than the combination IB1 and J48.

So far we have tried different combinations of classifiers and compared the combined results to the classifiers taking place in the combination. Now we will compare different combinations to each other. The results of all the combinations performed by using Dempster’s Rule of Combination is shown in Table 4.20. On the average the most successful combination is Naïve Bayes, IB1and J48 with a success rate of 92.75 and then comes Naïve Bayes, IB1 and OneR with a rate of 92.60 and then Naïve Bayes, J48 and OneR with the rate 92.45.

The least successful combinations are J48 and OneR (87.17%), IB1 and J48 (89.19) and IB1 and OneR (88.43%).

Combining classifiers using Dempster’s Rule of Combination not only performs better than each of the classifiers used in the combination but also the current hybrid classification algorithms. If we check again the information related to the WEKA hybrid classifiers in Tables 4.7 though 4.9, we see that, on the average, the most successful WEKA hybrid classifier is Decorate with the success rate of 86.85%. If we compare the combination results in Table 4.20 with the WEKA hybrid classifiers in Tables 4.7 though 4.9, we see that the most successful combination achieved using Demspter’s Rule of Combination is 5.9% more successful than the most successful WEKA hybrid classifier, namely Decorate.

Moreover, the WEKA hybrid classifier Decorate has a success rate of 98.31 on Wine data, which is the maximum of all the hybrid classifiers in WEKA, while the triple combination Naive Bayes, IB1 and J48 in our approach has a success rate of 99.95 on Iris data. This means that the most successful combination in our approach is 1.64% better than the most successful WEKA hybrid classifier.

So far, we have used the default values of hybrid classisifiers implemented in WEKA. In order to make one-to-one comparison of the proposed method of combining classifiers using Dempster’s Rule of Combination with the current hybrid algorithms, we do experiments again with the hybrid classifiers which has the capability of combining multiple classifiers. The hybrid algorithms that we will compare with the proposed method are;

**Table 4.20** Comparison of Results of Classifier Combination Using Dempster's Rule Combination

| Dataset | Naïve Bayes + IB1 | Naïve Bayes + J48 | Naïve Bayes + OneR | IB1 + J48 | IB1 + OneR | J48 + OneR | Naïve Bayes + IB1 + J48 | Naïve Bayes + IB1 + OneR | Naïve Bayes + J48 + OneR | IB1 + J48 + OneR |
|---|---|---|---|---|---|---|---|---|---|---|
| Autos | 98.31 | 96.74 | 94.65 | 99.21 | 98.69 | 97.48 | 99.74 | 99.58 | 99.14 | 99.79 |
| Breast-Cancer-Wisconsin | 99.83 | 99.80 | 99.73 | 99.66 | 99.54 | 99.44 | 99.93 | 99.91 | 99.91 | 99.91 |
| Heart-Disease-Cleveland | 54.37 | 52.09 | 52.09 | 51.18 | 49.77 | 47.45 | 49.77 | 49.77 | 47.45 | 46.30 |
| Heart-Disease-Hungary | 88.44 | 95.10 | 97.30 | 82.27 | 82.27 | 92.28 | 96.19 | 96.19 | 98.52 | 93.94 |
| Hepatitis | 80.26 | 73.79 | 76.31 | 69.67 | 81.07 | 64.36 | 83.49 | 85.22 | 83.33 | 75.37 |
| Iris | 99.78 | 99.82 | 99.73 | 99.78 | 99.66 | 99.73 | 99.95 | 99.93 | 99.93 | 99.93 |
| Labor | 97.30 | 95.50 | 95.93 | 92.13 | 92.88 | 88.32 | 98.85 | 98.97 | 98.25 | 97.12 |
| Soybean | 98.92 | 99.14 | 87.49 | 98.78 | 82.83 | 85.93 | 99.79 | 96.86 | 98.43 | 96.86 |
| Thyroid | 99.87 | 99.63 | 99.58 | 99.73 | 99.69 | 99.14 | 99.91 | 99.90 | 99.90 | 99.90 |
| Wine | 99.73 | 99.68 | 98.68 | 99.51 | 97.99 | 97.63 | 99.92 | 99.68 | 99.68 | 99.68 |
| AVERAGE | 91.68 | 91.12 | 90.14 | 89.19 | 88.43 | 87.17 | 92.75 | 92.60 | 92.45 | 90.88 |

- Grading
- Multischeme
- Stacking
- Vote

We perform test using six UCI data sets, namely Autos, Breast Cancer Wisconsin, Hepatitis, Iris, Labor and Soybean, that we have utilized so far. Table 4.21 shows Comparison of Proposed Method of Combining Classifiers Using Demspter’s Rule of Combination with the current hybrid algorithms. For each hybrid algorithm, the first classifier is used as the base classifier for the combination.

The tests are performed as follows. For each combination (say Naive Bayes + IB1) every hybrid algorithm is tested on each of the six data sets and then the results of the success rates are averaged and written in Table 4.21. As we have done so far, we use 10 fold cross validation.

The analysis of Table 4.21 shows that the proposed method outperforms the existing hybrid algorithms on the given data sets. However the proposed method is not the best all the time, though not obvious from the table, some of the existing hybrid algorithms are better then the proposed algorithm for some data sets.

## 4.5 Results of Employing Degree of Confidence in the Combination Using Dempster’s Rule of Combination

In the previous subsection we have combined four different classifiers in several ways using Dempster’s Rule of Combination. We have performed experiments on UCI data sets to test the performance of the combined classifiers. We have compared the results of the combined algorithms to the current hybrid algorithms. We have not employed degree of confidence during the combination.

The degree of confidence or the confidence factor is in fact the average success rate of the classification algorithm that it has displayed in the past. For our purpose we take the average success rate of classifiers that they have displayed on similar data sets in the past and use them as the degree of confidence during the classification process.

**Table 4.21** Comparison of the Proposed Method with the Current Hybrid Algorithms (%)

| Dataset | Naïve Bayes + IB1 | Naïve Bayes + J48 | Naïve Bayes + OneR | IB1 + J48 | IB1 + OneR | J48 + OneR |
|---|---|---|---|---|---|---|

| | | | | | | |
|---|---|---|---|---|---|---|
| Demspter-Shafer | 95.73 | 94.13 | 92.30 | 93.20 | 92.44 | 89.21 |
| Grading | 89.73 | 91.39 | 90.17 | 88.38 | 87.58 | 89.05 |
| Multischeme | 89.86 | 91.04 | 90.02 | 89.57 | 89.57 | 87.91 |
| Stacking | 90.33 | 91.61 | 89.78 | 83.14 | 78.38 | 87.66 |
| Vote | 89.54 | 91.39 | 77.90 | 89.60 | 84.96 | 81.86 |

In this part of the thesis we employ degree confidence during the combination and later check the amount of uncertainty of the combination. We expect the amount of uncertainty to decrease with the use of degree of confidence.

We combine the four algorithms Naïve Bayes, IB1, J48, and OneR by employing degree of confidence in the following manner:

- Naïve Bayes and IB1
- Naïve Bayes and J48
- Naïve Bayes and OneR
- IB1 and J48
- IB1 and OneR
- J48 and OneR
- Naïve Bayes, IB1 and J48
- Naïve Bayes, IB1 and OneR
- Naïve Bayes, J48 and OneR
- IB1, J48 and OneR

First of all we employ degree of confidence in the combination of Naïve Bayes and IB1 using Demspter's Rule of Combination. For this combination we use the average success rate of 86.08% for the Naïve Bayes classifier and the average success rate of 82.90% for the classifier IB1. This means that we can trust Naïve Bayes classifier by 82.90% since it has displayed this average success rate in the past. Table 4.22 shows the combination results of the classifiers Naïve Bayes and IB1, uncertainty of the combination before the use of degree of confidence (i.e., Uncertainty w/o confidence) and uncertainty of the combination after the use of degree of confidence (i.e., Uncertainty with confidence) and the improvement in the uncertainty of the combination.

Examining Table 4.22 shows that uncertainty before the use of degree of confidence was 13.80% (highest) for Heart-Disease-Cleveland data, 9.12% for the Hepatitis data and 5.79% for the Heart-Disease-Hungary data. In the mean time the average uncertainty was 3.35% for all the data sets. After using degree of confidence during the combination, uncertainty drops down to 9.47% (still highest) for Heart-Disease-Cleveland data, 6.34% for the Hepatitis data and 4.07% for the Heart-Disease-Hungary data. This corresponds to an improvement in the uncertainty 4.33%, 2.78% and 1.72% for the data sets Heart-Disease-Cleveland, Hepatitis, and Heart-Disease-Hungary consecutively. At the same time the average uncertainty goes down to 1.1% which is an improvement of 2.2% on the average.

Table 4.22 shows the combination results of the classifiers Naïve Bayes and J48, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.23 shows that uncertainty before the use of degree of confidence was 13.95% (highest) for Heart-Disease-Cleveland data, 10.64% for the Hepatitis data and 3.25% for the Heart-Disease-Hungary data. In the mean time the average uncertainty was 3.47% for all the data sets. After using degree of confidence during the combination, uncertainty drops down to 9.57% (still highest) for Heart-Disease-Cleveland data, 7.36% for the Hepatitis data and 2.3% for the Heart-Disease-Hungary data. This corresponds to an improvement in the uncertainty 4.38%, 3.28% and 0.95% for the data sets Heart-Disease-Cleveland, Hepatitis, and Heart-Disease-Hungary consecutively. At the same time the average uncertainty goes down to 1.77% which is an improvement of 1.7% on the average.

Table 4.24 shows the combination results of the classifiers Naïve Bayes and OneR, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.24 shows that uncertainty before the use of degree of confidence was 13.95% (highest) for Heart-Disease-Cleveland data, 10.12% for the Hepatitis data and 4.48% for the Soybean data. In the mean time the average uncertainty was 3.83% for all the data sets.

**Table 4.22** Results of Employing Degree of Confidence in the Combination of Naïve Bayes and IB1 using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | IB1 | Naïve Bayes + IB1 | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 79.51 | 94.15 | 98.31 | 1.25 | 0.89 | 0.36 |
| Breast-Cancer-Wisconsin | 97.42 | 95 | 99.83 | 0.15 | 0.11 | 0.04 |
| Heart-Disease-Cleveland | 56.43 | 54.45 | 56.10 | 13.80 | 9.47 | 4.33 |
| Heart-Disease-Hungary | 85.35 | 59.18 | 88.63 | 5.79 | 4.07 | 1.72 |
| Hepatitis | 70.32 | 66.45 | 80.76 | 9.12 | 6.34 | 2.78 |
| Iris | 96 | 95.33 | 99.78 | 0.2 | 0.14 | 0.06 |
| Labor | 89.47 | 82.45 | 97.32 | 1.97 | 1.39 | 0.58 |
| Soybean | 92.97 | 89.89 | 98.93 | 0.88 | 0.62 | 0.26 |
| Thyroid | 96.74 | 97.20 | 99.87 | 0.12 | 0.08 | 0.04 |
| Wine | 96.62 | 94.94 | 99.73 | 0.24 | 0.17 | 0.07 |
| AVERAGE | 86.08 | 82.90 | 91.92 | 3.35 | 2.49 | 1.10 |

**Table 4.23** Results of Employing Degree of Confidence in the Combination of Naïve Bayes and J48 using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | J48 | Naïve Bayes + J48 | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 79.51 | 89.75 | 96.76 | 2.29 | 1.62 | 0.67 |
| Breast-Cancer-Wisconsin | 97.42 | 94.85 | 99.80 | 0.18 | 0.13 | 0.05 |
| Heart-Disease-Cleveland | 56.43 | 52.47 | 53.93 | 13.95 | 9.57 | 4.38 |
| Heart-Disease-Hungary | 85.35 | 78.57 | 95.14 | 3.25 | 2.3 | 0.95 |
| Hepatitis | 70.32 | 58.70 | 74.57 | 10.64 | 7.36 | 3.28 |
| Iris | 96 | 96 | 99.82 | 0.16 | 0.12 | 0.04 |
| Labor | 89.47 | 73.68 | 95.53 | 2.92 | 2.07 | 0.85 |
| Soybean | 92.97 | 91.50 | 99.14 | 0.72 | 0.51 | 0.21 |
| Thyroid | 96.74 | 92.09 | 99.63 | 0.32 | 0.23 | 0.09 |
| Wine | 96.62 | 93.82 | 99.68 | 0.28 | 0.20 | 0.08 |
| AVERAGE | 86.08 | 82.14 | 91.40 | 3.47 | 1.77 | 1.7 |

After using degree of confidence during the combination, uncertainty drops down to 9.57% (still highest) for Heart-Disease-Cleveland data, 7.01% for the Hepatitis data and 3.17% for the Soybean data. This corresponds to an improvement in the uncertainty 4.38%, 3.11% and 1.31% for the data sets Heart-Disease-Cleveland, Hepatitis, and Soybean consecutively. At the same time the average uncertainty goes down to 2.52 % which is an improvement of 1.3% on the average.

Table 4.25 shows the combination results of the classifiers IB1 and J48 , uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.25 shows that uncertainty before the use of degree of confidence was 13.71% (highest) for Heart-Disease-Cleveland data, 11.61% for the Hepatitis data, 8.16% for Heart-Disease-Hungary and 4.71% for the Labor data. In the mean time the average uncertainty was 4.08% for all the data sets. After using degree of confidence during the combination, uncertainty drops down to 9.67% (still highest) for Heart-Disease-Cleveland data, 8.01% for the Hepatitis data, 5.69% for the Heart-Disease-Hungary and 3.31% for the Labor data. This corresponds to an improvement in the uncertainty 4.04%, 3.6%, 2.47% and 1.4% for the data sets Heart-Disease-Cleveland, Hepatitis, Heart-Disease-Hungary and Labor consecutively. At the same time the average uncertainty goes down to 2.26% which is an improvement of 1.82% on the average.

Table 4.26 shows the combination results of the classifiers IB1 and OneR, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.26 shows that uncertainty before the use of degree of confidence was 14.1% (highest) for Heart-Disease-Cleveland data, 8.87% for the Hepatitis data, 8.16% for Heart-Disease-Hungary, 8.16% for the Soybean and 4.38% for the Labor data. In the mean time the average uncertainty was 4.48% for all the data sets.

After using degree of confidence during the combination, uncertainty drops down to 9.68% (still highest) for Heart-Disease-Cleveland data, 7.66% for the Hepatitis data, 5.69% for the Heart-Disease-Hungary, 4.18% for the Soybean and 3.08% for the Labor data. This corresponds to an improvement in the uncertainty 4.42%, 1.21%,

**Table 4.24** Results of Employing Degree of Confidence in the Combination of Naïve Bayes and OneR using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | OneR | Naïve Bayes + OneR | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 79.51 | 83.41 | 94.7 | 3.51 | 2.48 | 1.03 |
| Breast-Cancer-Wisconsin | 97.42 | 92.70 | 99.73 | 0.24 | 0.17 | 0.07 |
| Heart-Disease-Cleveland | 56.43 | 52.47 | 53.93 | 13.95 | 9.57 | 4.38 |
| Heart-Disease-Hungary | 85.35 | 78.91 | 95.14 | 1.79 | 2.3 | 0.51 |
| Hepatitis | 70.32 | 61.93 | 76.98 | 10.12 | 7.01 | 3.11 |
| Iris | 96 | 94 | 99.73 | 0.24 | 0.17 | 0.07 |
| Labor | 89.47 | 75.43 | 95.96 | 2.71 | 1.92 | 0.79 |
| Soybean | 92.97 | 39.97 | 87.64 | 4.48 | 3.17 | 1.31 |
| Thyroid | 96.74 | 91.16 | 99.58 | 0.36 | 0.26 | 0.10 |
| Wine | 96.62 | 76.40 | 98.69 | 0.96 | 0.68 | 0.28 |
| AVERAGE | 86.08 | 74.63 | 90.20 | 3.83 | 2.52 | 1.31 |

**Table 4.25** Results of Employing Degree of Confidence in the Combination of IB1 and J48 using Demspter's Rule of Combination (%)

| Dataset | IB1 | J48 | IB1 + J48 | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 94.15 | 89.75 | 99.21 | 0.49 | 0.47 | 0.02 |
| Breast-Cancer-Wisconsin | 95 | 94.85 | 99.66 | 0.30 | 0.21 | 0.09 |
| Heart-Disease-Cleveland | 54.45 | 52.47 | 51.72 | 13.71 | 9.67 | 4.04 |
| Heart-Disease-Hungary | 59.18 | 78.57 | 82.67 | 8.16 | 5.69 | 2.47 |
| Hepatitis | 66.45 | 58.70 | 70.65 | 11.61 | 8.01 | 3.6 |
| Iris | 95.33 | 96 | 99.78 | 0.20 | 0.14 | 0.06 |
| Labor | 82.45 | 73.68 | 92.24 | 4.71 | 3.31 | 1.4 |
| Soybean | 89.89 | 91.50 | 98.78 | 0.98 | 0.7 | 0.28 |
| Thyroid | 97.20 | 92.09 | 99.73 | 0.24 | 0.17 | 0.07 |
| Wine | 94.94 | 93.82 | 99.52 | 0.42 | 0.29 | 0.13 |
| AVERAGE | 82.90 | 82.14 | 89.39 | 4.08 | 2.26 | 1.82 |

After using degree of confidence during the combination, uncertainty drops down to 9.57% (still highest) for Heart-Disease-Cleveland data, 7.01% for the Hepatitis data and 3.17% for the Soybean data. This corresponds to an improvement in the uncertainty 4.38%, 3.11% and 1.31% for the data sets Heart-Disease-Cleveland, Hepatitis, and Soybean consecutively. At the same time the average uncertainty goes down to 2.52 % which is an improvement of 1.3% on the average.

Table 4.25 shows the combination results of the classifiers IB1 and J48 , uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.25 shows that  uncertainty before the use of degree of confidence was 13.71% (highest) for Heart-Disease-Cleveland data, 11.61% for the Hepatitis data, 8.16% for Heart-Disease-Hungary and 4.71% for the Labor data. In the mean time the average uncertainty was 4.08% for all the data sets. After using degree of confidence during the combination, uncertainty drops down to 9.67% (still highest) for Heart-Disease-Cleveland data, 8.01% for the Hepatitis data, 5.69% for the Heart-Disease-Hungary and 3.31% for the Labor data. This corresponds to an improvement in the uncertainty 4.04%, 3.6%, 2.47% and 1.4% for the data sets Heart-Disease-Cleveland, Hepatitis, Heart-Disease-Hungary and Labor consecutively. At the same time the average uncertainty goes down to 2.26% which is an improvement of 1.82% on the average.

Table 4.26 shows the combination results of the classifiers IB1 and OneR, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.26  shows that  uncertainty before the use of degree of confidence was 14.1% (highest) for Heart-Disease-Cleveland data, 8.87% for the Hepatitis data, 8.16% for Heart-Disease-Hungary, 8.16% for the Soybean and 4.38% for the Labor data. In the mean time the average uncertainty was 4.48% for all the data sets.

After using degree of confidence during the combination, uncertainty drops down to 9.68% (still highest) for Heart-Disease-Cleveland data, 7.66% for the Hepatitis data, 5.69% for the Heart-Disease-Hungary, 4.18% for the Soybean  and 3.08% for the Labor data. This corresponds to an improvement in the uncertainty 4.42%, 1.21%,

**Table 4.26** Results of Employing Degree of Confidence in the Combination of IB1 and OneR using Demspter's Rule of Combination (%)

| Dataset | IB1 | OneR | IB1 + OneR | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 94.15 | 83.41 | 98.70 | 1.02 | 0.72 | 0.3 |
| Breast-Cancer-Wisconsin | 95 | 92.70 | 99.54 | 0.40 | 0.28 | 0.12 |
| Heart-Disease-Cleveland | 54.45 | 52.47 | 51.71 | 14.1 | 9.68 | 4.42 |
| Heart-Disease-Hungary | 59.18 | 78.91 | 82.67 | 8.16 | 5.69 | 2.47 |
| Hepatitis | 66.45 | 61.93 | 73.32 | 8.87 | 7.66 | 1.21 |
| Iris | 95.33 | 94 | 99.66 | 0.30 | 0.21 | 0.09 |
| Labor | 82.45 | 75.43 | 92.97 | 4.38 | 3.08 | 1.3 |
| Soybean | 89.89 | 39.97 | 83.12 | 5.96 | 4.18 | 1.78 |
| Thyroid | 97.20 | 91.16 | 99.69 | 0.27 | 0.19 | 0.08 |
| Wine | 94.94 | 76.40 | 98.00 | 1.43 | 1.02 | 0.41 |
| AVERAGE | 82.90 | 74.63 | 87.93 | 4.48 | 3.2 | 1.28 |

**Table 4.27** Results of Employing Degree of Confidence in the Combination of J48 and OneR using Demspter's Rule of Combination (%)

| Dataset | J48 | OneR | J48 + OneR | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|
| Autos | 89.75 | 83.41 | 97.49 | 1.86 | 1.32 | 0.54 |
| Breast-Cancer-Wisconsin | 94.85 | 92.70 | 99.44 | 0.48 | 0.34 | 0.14 |
| Heart-Disease-Cleveland | 52.47 | 52.47 | 49.51 | 14.21 | 9.74 | 4.47 |
| Heart-Disease-Hungary | 78.57 | 78.91 | 92.38 | 4.69 | 3.31 | 1.38 |
| Hepatitis | 58.70 | 61.93 | 65.60 | 12.61 | 8.68 | 3.93 |
| Iris | 96 | 94 | 99.73 | 0.24 | 0.17 | 0.07 |
| Labor | 73.68 | 75.43 | 88.53 | 6.39 | 4.48 | 1.91 |
| Soybean | 91.50 | 39.97 | 86.13 | 4.99 | 3.51 | 1.48 |
| Thyroid | 92.09 | 91.16 | 99.14 | 0.72 | 0.51 | 0.21 |
| Wine | 93.82 | 76.40 | 97.65 | 1.67 | 1.18 | 0.49 |
| AVERAGE | 82.14 | 74.63 | 87.56 | 4.78 | 3.32 | 1.46 |

2.47%, 1.78% and 1.3% for the data sets Heart-Disease-Cleveland, Hepatitis, Heart-Disease-Hungary, Soybean and Labor consecutively.At the same time the average uncertainty goes down to 3.2% which is an improvement of 1.28% on the average.

Table 4.27 shows the combination results of the classifiers J48 and OneR, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.27 shows that uncertainty before the use of degree of confidence was 14.21% (highest) for Heart-Disease-Cleveland data, 12.61% for the Hepatitis data, 4.69% for Heart-Disease-Hungary, 4.99% for the Soybean and 6.39% for the Labor data. In the mean time the average uncertainty was 4.78% for all the data sets.

After using degree of confidence during the combination, uncertainty drops down to 9.74% (still highest) for Heart-Disease-Cleveland data, 8.68% for the Hepatitis data, 3.31% for the Heart-Disease-Hungary, 3.51% for the Soybean and 4.48% for the Labor data. This corresponds to an improvement in the uncertainty 4.47%, 3.93%, 1.38%, 1.48% and 1.91% for the data sets Heart-Disease-Cleveland, Hepatitis, Heart-Disease-Hungary, Soybean and Labor consecutively. At the same time the average uncertainty goes down to 3.32% which is an improvement of 1.46% on the average.

Table 4.28 shows the combination results of the classifiers Naïve Bayes, IB1 and J48, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.28 shows that uncertainty before the use of degree of confidence was 14.1% (highest) for Heart-Disease-Cleveland data, and 7.66% for the Hepatitis data. The average uncertainty was 4.36% for all the data sets.

After employing degree of confidence at the combination, uncertainty drops down to 9.57% (still highest) for Heart-Disease-Cleveland data, and 5.35% for the Hepatitis data. This corresponds to an improvement in the uncertainty 4.53% and 2.32% for the data sets Heart-Disease-Cleveland and Hepatitis consecutively. At the same time the average uncertainty goes down to 1.78% which is an improvement of 0.81% on the average.

Table 4.29 shows the combination results of the classifiers Naïve Bayes, IB1 and OneR, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

**Table 4.28** Results of Employing Degree of Confidence in the Combination of Naïve Bayes, IB1 and J48 using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | IB1 | Naïve Bayes + IB1 | J48 | Naïve Bayes + IB1 + J48 | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|---|---|
| Autos | 79.51 | 94.15 | 98.31 | 89.75 | 99.74 | 0.24 | 0.16 | 0.08 |
| Breast-Cancer-Wisconsin | 97.42 | 95 | 99.83 | 94.85 | 99.93 | 0.07 | 0.04 | 0.03 |
| Heart-Disease-Cleveland | 56.43 | 54.45 | 56.10 | 52.47 | 53.93 | 14.1 | 9.57 | 4.53 |
| Heart-Disease-Hungary | 85.35 | 59.18 | 88.63 | 78.57 | 96.22 | 2.61 | 1.85 | 0.76 |
| Hepatitis | 70.32 | 66.45 | 80.76 | 58.70 | 83.85 | 7.66 | 5.35 | 2.32 |
| Iris | 96 | 95.33 | 99.78 | 96 | 99.95 | 0.04 | 0.03 | 0.01 |
| Labor | 89.47 | 82.45 | 97.32 | 73.68 | 98.86 | 0.81 | 0.57 | 0.24 |
| Soybean | 92.97 | 89.89 | 98.93 | 91.50 | 99.79 | 0.18 | 0.13 | 0.05 |
| Thyroid | 96.74 | 97.20 | 99.87 | 92.09 | 99.91 | 0.08 | 0.05 | 0.03 |
| Wine | 96.62 | 94.94 | 99.73 | 93.82 | 99.92 | 0.07 | 0.05 | 0.02 |
| AVERAGE | 86.08 | 82.90 | 91.22 | 82.14 | 93.21 | 4.36 | 1.78 | 0.81 |

**Table 4.29** Results of Employing Degree of Confidence in the Combination of Naïve Bayes, IB1 and OneR using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | IB1 | Naïve Bayes + IB1 | OneR | Naïve Bayes + IB1 + OneR | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|---|---|
| Autos | 79.51 | 94.15 | 98.31 | 83.41 | 99.58 | 0.41 | 0.24 | 0.17 |
| Breast-Cancer-Wisconsin | 97.42 | 95 | 99.83 | 92.70 | 99.91 | 0.08 | 0.05 | 0.03 |
| Heart-Disease-Cleveland | 56.43 | 54.45 | 56.10 | 52.47 | 53.93 | 14.1 | 9.57 | 4.53 |
| Heart-Disease-Hungary | 85.35 | 59.18 | 88.63 | 78.91 | 96.22 | 2.61 | 1.85 | 0.76 |
| Hepatitis | 70.32 | 66.45 | 80.76 | 61.93 | 85.52 | 7.21 | 5.04 | 2.17 |
| Iris | 96 | 95.33 | 99.78 | 94 | 99.93 | 0.06 | 0.04 | 0.02 |
| Labor | 89.47 | 82.45 | 97.32 | 75.43 | 98.97 | 0.74 | 0.53 | 0.21 |
| Soybean | 92.97 | 89.89 | 98.93 | 39.97 | 96.88 | 1.2 | 0.85 | 0.35 |
| Thyroid | 96.74 | 97.20 | 99.87 | 91.16 | 99.90 | 0.09 | 0.06 | 0.03 |
| Wine | 96.62 | 94.94 | 99.73 | 76.40 | 99.68 | 0.24 | 0.17 | 0.07 |
| AVERAGE | 86.08 | 82.90 | 91.22 | 74.63 | 93.05 | 2.67 | 1.84 | 0.83 |

Examining Table 4.29 shows that uncertainty before the use of degree of confidence was 14.1% (highest) for Heart-Disease-Cleveland data, and 7.21% for the Hepatitis data. The average uncertainty was 2.67% for all the data sets.

After employing degree of confidence at the combination, uncertainty drops down to 9.57% (still highest) for Heart-Disease-Cleveland data, and 5.04% for the Hepatitis data which corresponds to an improvement in the uncertainty 4.53% and 2.17% for the data sets Heart-Disease-Cleveland and Hepatitis consecutively. At the same time the average uncertainty goes down to 1.84% which is an improvement of 0.83% on the average.

Table 4.30 shows the combination results of the classifiers Naïve Bayes, J48 and OneR, uncertainty of the combination before and after the use of degree of confidence and the improvement in the uncertainty of the combination.

Examining Table 4.30 shows that uncertainty before the use of degree of confidence was 14.21% (highest) for Heart-Disease-Cleveland data, and 8.15% for the Hepatitis data. The average uncertainty was 2.64% for all the data sets.

After employing degree of confidence at the combination, uncertainty drops down to 9.71% (still highest) for Heart-Disease-Cleveland data, and 6.28% for the Hepatitis data which corresponds to an improvement in the uncertainty 4.5% and 1.87% for the data sets Heart-Disease-Cleveland and Hepatitis consecutively. At the same time the average uncertainty goes down to 1.89% which is an improvement of 0.76% on the average.

Table 4.31 displays the combination results of the classifiers IB1, J48 and OneR. Examining the table shows that uncertainty before the use of degree of confidence was 14.24% for Heart-Disease-Cleveland data, 10.36% for the Hepatitis, and 3.87% for Heart-Disease-Hungary data. The average uncertainty was 3.23% for all the data sets.

After employing degree of confidence at the combination, uncertainty drops down to 9.77% for Heart-Disease-Cleveland data, 7.01% for the Hepatitis data and 2.73% for Heart-Disease-Hungary data which corresponds to an improvement in the uncertainty 4.47%, 3.35% and 1.14% for the data sets Heart-Disease-Cleveland, Hepatitis and Heart-Disease-Hungary consecutively. At the same time the average uncertainty goes down to 2.22% which is an improvement of 1.01% on the average.

**Table 4.30** Results of Employing Degree of Confidence in the Combination of Naïve Bayes, J48 and OneR using Demspter's Rule of Combination (%)

| Dataset | Naïve Bayes | J48 | Naïve Bayes + J48 | OneR | Naïve Bayes + J48 + OneR | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|---|---|
| Autos | 79.51 | 89.75 | 96.76 | 83.41 | 99.14 | 0.68 | 0.49 | 0.19 |
| Breast-Cancer-Wisconsin | 97.42 | 94.85 | 99.80 | 92.70 | 99.91 | 0.08 | 0.05 | 0.03 |
| Heart-Disease-Cleveland | 56.43 | 52.47 | 53.93 | 52.47 | 50.61 | 14.21 | 9.71 | 4.5 |
| Heart-Disease-Hungary | 85.35 | 78.57 | 95.14 | 78.91 | 98.52 | 1.09 | 0.78 | 0.31 |
| Hepatitis | 70.32 | 58.70 | 74.57 | 61.93 | 80.50 | 8.15 | 6.28 | 1.87 |
| Iris | 96 | 96 | 99.82 | 94 | 99.93 | 0.06 | 0.04 | 0.02 |
| Labor | 89.47 | 73.68 | 95.53 | 75.43 | 98.26 | 1.24 | 0.88 | 0.36 |
| Soybean | 92.97 | 91.50 | 99.14 | 39.97 | 98.43 | 0.60 | 0.43 | 0.17 |
| Thyroid | 96.74 | 92.09 | 99.63 | 91.16 | 99.90 | 0.09 | 0.06 | 0.03 |
| Wine | 96.62 | 93.82 | 99.68 | 76.40 | 99.68 | 0.24 | 0.17 | 0.07 |
| AVERAGE | 86.08 | 82.14 | 91.40 | 74.64 | 92.49 | 2.64 | 1.89 | 0.76 |

**Table 4.31** Results of Employing Degree of Confidence in the Combination of IB1, J48 and OneR using Demspter's Rule of Combination (%)

| Dataset | IB1 | J48 | IB1 + J48 | OneR | IB1 + J48 + OneR | Uncertainty w/o confidence | Uncertainty with confidence | Improvement in Uncertainty |
|---|---|---|---|---|---|---|---|---|
| Autos | 94.15 | 89.75 | 99.21 | 83.41 | 99.79 | 0.17 | 0.12 | 0.05 |
| Breast-Cancer-Wisconsin | 95 | 94.85 | 99.66 | 92.70 | 99.91 | 0.08 | 0.05 | 0.03 |
| Heart-Disease-Cleveland | 54.45 | 52.47 | 51.72 | 52.47 | 48.40 | 14.24 | 9.77 | 4.47 |
| Heart-Disease-Hungary | 59.18 | 78.57 | 82.67 | 78.91 | 94.01 | 3.87 | 2.73 | 1.14 |
| Hepatitis | 66.45 | 58.70 | 70.65 | 61.93 | 76.98 | 10.36 | 7.01 | 3.35 |
| Iris | 95.33 | 96 | 99.78 | 94 | 99.93 | 0.06 | 0.04 | 0.02 |
| Labor | 82.45 | 73.68 | 92.24 | 75.43 | 97.14 | 1.98 | 1.4 | 0.58 |
| Soybean | 89.89 | 91.50 | 98.78 | 39.97 | 96.88 | 1.2 | 0.85 | 0.35 |
| Thyroid | 97.20 | 92.09 | 99.73 | 91.16 | 99.90 | 0.09 | 0.06 | 0.03 |
| Wine | 94.94 | 93.82 | 99.52 | 76.40 | 99.68 | 0.24 | 0.17 | 0.07 |
| AVERAGE | 82.90 | 82.14 | 89.60 | 74.64 | 91.26 | 3.23 | 2.22 | 1.01 |

The Comparison of the proposed method with Mahajani and Aslandoğan's work (1993) is shown in Table 4.32. As seen from the table, Mahajani and Aslandoğan's implementation of K-Nearest Neighbour, Naïve Bayes and Decision Tree algorithms are different from the ones that we use in our study. Although it may not be very appropriate to compare our study with theirs, just checking the results of the combnations in the two sdudies show that the combination IB1-Naïve Bayes-J48 gives more successful classification result.

**Table 4.32** Comparison of the Proposed Method with Mahajani and Aslandoğan's Work (1993)

| DATA SET | kNN | Naïve Bayes | Decision Tree | KNN + Naïve Bayes+ Decision Tree | IB1 | Naïve Bayes | J48 | IB1 + Naïve Bayes+ J48 |
|---|---|---|---|---|---|---|---|---|
| Breast | 92 | 93 | 91 | 95.7 | 95 | 97.4 | 94.8 | 99.9 |

In order to sum up what we have done in terms of improvement in uncertainty so far we can check Table 4.33. Table 4.33 summarizes the uncertainty of different combinations. The first column shows the average uncertainty values without using degree of confidence. The second column displays the average uncertainty values obtained using degree of confidence of the classifiers taking part in the combination. The third column presents the improvement in the uncertainty.

**Table 4.33** Improvement in Uncertainty (%)

| COMBINATION | Average Uncertainty W/o using Degree of Confidence | Average Uncertainty using Degree of Confidence | Average Imprevement in Uncertainty |
|---|---|---|---|
| Naïve Bayes + IB1 | 3.35 | 2.49 | 1.10 |
| Naïve Bayes + J48 | 3.47 | 1.77 | 1.70 |
| Naïve Bayes + OneR | 3.83 | 2.52 | 1.31 |
| IB1 + J48 | 4.08 | 2.26 | 1.82 |
| IB1 + OneR | 4.48 | 3.20 | 1.28 |
| J48 + OneR | 4.78 | 3.32 | 1.46 |
| Naïve Bayes + IB1 + J48 | 4.36 | 1.78 | 0.81 |
| Naïve Bayes + IB1 + OneR | 2.67 | 1.84 | 0.83 |
| Naïve Bayes + J48 + OneR | 2.64 | 1.89 | 0.76 |
| IB1 + J48 + OneR | 3.23 | 2.22 | 1.01 |
| AVERAGE | 3.69 | 2.33 | 1.21 |

Average uncertainty without using degree of confidence during combination is 3.69 while the average uncertainty is 2.33 when degree of confidence is used at the combination. Average improvement in uncertainty is 1.21. Maximum improvement in the uncertainty is 4.5%.

## 5. CONCLUSION AND FUTURE WORK

In this study, we introduce a method for combining outputs of classification algorithms in order to improve the classification performance. The combination of the classification results is performed using Dempster's Rule of Combination, considering the classifier outputs as beliefs, with the use degree of condfidence of classifiers. The improvements achieved by this dissertation are summerized below:

- Employment of degree of confidence during the combination decreases the uncertainty in the combination.
- The reduction of uncertainty leads to more precise classification results.
- The proposed method improves performance of the classification when compared to the existing classification algorithms and existing hybrid algorithms.

Another important issue is that two different class of algorithms from two different areas, namely Dempster-Shafer Method from the field of data fusion and classification algorithms from data mining area come together to obtain improved classification performance.

We perform different experiments using UCI data sets. Firstly, we test the success rate of the current classifiers in WEKA using 10 different data sets taken from the UCI machine learning repository. In the experiments we use the default values of the classifiers. We then check the success rate of current hybrid classifiers using the same data sets with the default values. Afterwards we do tests with the proposed method of combining classifiers using Dempster's Rule of Combination using the same data sets.

In order to be able to make one-to-one comparison of the proposed method with the current hybrid classification algorithms we perform experiments with the hybrid algorithms which has the capability of using multiple classifiers on the same data sets used in the previous experiments.

According to the results of the experiments performed with the default values, the most successful classifier is on the average Logistic with a success rate of 89.22. In the mean time, the classifiers with the maximum success rate are BayesNet and Kstar classification algorithms in WEKA. They both hava a success rate of 98.87 on “Wine” data.

Testing the current hybrid classification algorithms on the same UCI data sets show that Decorate is the algorithm with the greatest average success rate of 86.25. In the mean time the hybrid classifier with the maximum success rate is Logitboost with a rate of 98.31 on “Wine” data.

Testing our proposed method on the same data sets show that combining classifiers using Dempster’s Rule of Combination not only performs better than each of the classifiers used in the combination but also the current hybrid classification algorithms. On the average, the most successful combination is “Naïve Bayes, IB1 and J48” with a success rate of 92.75 which is 3.5% more successful than the most successful WEKA classifiers Bayes Net and Kstar. At the same time, the same combination is 5.9% more successful than the most successful WEKA hybrid classifier Decorate (86.25%).

The analysis performed so far belongs to the classifiers tested with the default values. According to the results of the experiments performed for doing one-to-one correspondence between the proposed method and the current hybrid algorithms, the proposed method outperforms the current hybrid classification algorithms on the data sets used in the experiments.

In our proposed method, degree of confidence is the average success rate of a classification algorithm that it has displayed in the past on similar data. The experiments performed on UCI data sets show that adding degree of confidence in the combination of classifiers using Dempster’s Rule of Combination increases accuracy of the combination by 4.5 %.

In conclusion, we can say that combining the classifier outputs using Dempster’s Rule of Combination with the employment of degree of confidence yields better results than each of the classifiers taking place in the combination and existing hybrid classification algorithms. But we must always keep in mind that our proposed method is better than most of the current classifiers and current hybrid classification algorithms on the data sets used in the experiments. Changing the data sets may well reverse the situation.

Classification algorithms may assign class values to instances or may not do any class assignment. Some instances may be classified correctly and some instances may not be classified at all. Each classification algorithm may include uncertainty to some extent. The reason why Dempster’s Rule of Combination is more succesful than the current algorithms lies in the fact that Dempster’s Rule of Combination has the capability of uncertainty management which distinguishes the method from the current algorithms.

In the proposed method of combining classification algorithms using Dempster's Rule of Combination, the combination is performed in a pairwise fashion. Some further study on Dempster's Rule of Combination for making several combinations at once may be a promising subject for future work.

The proposed method may be used in the area of stream data mining. Application areas like network-traffic monitoring, computer-network security, e-commerce, sensor networks, financial monitoring require stream data mining techniques. Since the data arrive in streams, storage may not be enough to hold all the data; previous results must be updated with the new data. In such a scenario, uncertainty in performing classification is quite high. The proposed method of combining classification algorithm using  Dempster's Rule of Combination may be modified to be employed in stream data mining.

## RERERENCES

**Aha, D. and Kibler, D.,** 1991**.** Instance-based Learning Algorithms, *Machine Learning*, **6**, 37-66.

**Ali, K. and Pazzani, M. J.,** 1996**.** Error Reduction Through Learning Model Descriptions. *Machine Learning*, 173:202.

**Al-Ani, A. and Deriche, M.,** 2002**.** A New technique for Combining Multiple Classifiers using the Dempster-Shafer Theory of Evidence, *Journal of Artifical Intelligence Research*, 17, 333-361.

**Basak, J., Goyal, Z. and Kothari, R.,** 2004**.** A Modified Dempster's Rule Of Combination for Weighted Sources of Evidence, *IBM Research Report*, IBM Research Division, IBM India Research Lab.

**Bates, J.M. and Granger, C.W.J.,** 1969**.** Combination of forecasts, *Operational Research Quarterly*, 20 (4), 451-468.

**Breiman, L.** 1996**.** Stacked regression, *Machine Learning*, 24, 49-64.

**Bosse, E. and Roy, J.,** 1997. Fusion of identity declarations from dissimilar sources using the Dempster-Shafer theory, *Opt. Eng*. 36(3), 648-657

Breiman, L. ,**1996a**. **Stacked regression, *Machine Learning*, 24, 49-64.**

Breiman, L., **1996b**. **Bagging Predictors, *Machine Learning*, 24, 123-140.**

Breiman, L., **1996c**. **Arcing classifiers, *Annals of Statistics*, 26, 801-849.**

Cessie, S. and Houwelingen, J.C., **1992**. **Ridge Estimators in Logistic Regression, *Applied Statistics*,** 41**, No. 1, 191-201.**

**Clemen, R.T.,** 1989**.** Combining forecasts: A review and annotated bibliography, *International Journal of Forecasting*, 5, 559—583.

**Cohen, W.W.,** 1995**.** Fast Effective Rule Induction *Machine Learning: Proceedings of the Twelfth International Conference*, 115-123.

**Dempster, A. P.,** 1967**.** Upper and lower probabilities induced by a multivalued mapping. *The Annals of Mathematical Statistics*, 38, 325-339.

**Dietterich, T.G.,** 2000a**.** An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization, *Machine Learning*, 40 (2), 139-158

**Dietterich, T.G.,** 2000b**.** Ensemble methods in machine learning, *Proceedings of the First International Workshop on Multiple Classifier Systems,* 1-15.

**Donker, J.C.,** 1991**.** Reasoning with uncertain and incomplete information in aerospace application. *AGARD Symposium on Machine Intelligence for Aerospace Electronic Systems,* Lisbon, 30, 30.1-30.16.

**Dzeroski, S. and Zenko, B.,** 2004. Is Combining Classifiers Better than Selecting the Best One?, *Machine Learning*, 255-273.

**Gama, J. and Brazdil, P.,** 2000. Cascade Generalization, *Machine Learning*, 315-343.

**Geoffrey, I.,W.,** 2000**.** MultiBoosting: A Technique for Combining Boosting and Wagging, *Machine Learning,* 159-196.

**Granger, C.W.J. and Ramanathan, R.,** 1984**.** Improved methods of combining information, *Journal of Forecasting*, 3, 197—204.

**Fabiani, P.,J.,** 1994. A New Approach in Temporal Representation of Belief for Autonomous Observation and Surveillance Systems , *11th European Conference on Artifical Intelligence,* 391-395.

**Frank, E. and Hall, M.,** 2001. A simple approach to ordinal prediction, *12th European Conference on Machine Learning,* 145-157.

**Frank, E., Hall, M., and Pfahringer, B**., 2003. Locally Weighted Naive Bayes, *Proceedings of the Conference on Uncertainty in Artificial Intelligence,* 249-256.

Frank, E., Wang, Y., Inglis, S., Holmes, G. and Witten, I.H. **1998. Using model trees for classification ,** ***Machine Learning,*** 32**, No.1, 63-76.**

**Frank E. and Witten I.H.,** 1998. Generating Accurate Rule Sets Without Global Optimization. *Proceedings of the Fifteenth International Conference,* 249-256.

Freund, Y. and Schapire, R.E., **1996. Experiments with a new boosting algorithm, *Proceeding of International Conference on Machine Learning*, 148-156.**

**Friedman, J., Hastie, T. and Tibshirani, R.,** 2000. Additive Logistic Regression: a Statistical View of Boosting, *Technical Report*. Stanford University.

**John, G.C., Leonard, E. and Trigg,** 1995. K*: An Instance- based Learner Using an Entropic Distance Measure, *Proceedings of the 12th International Conference on Machine learning*, 108-114.

**John, G. H. and Langley, P.,** 1995**.** Estimating Continuous Distributions in Bayesian Classifiers., *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*., Morgan Kaufmann, San Mateo, 338-345.

**Han, J. and Kamber, M.,** 2000. Data Mining: Concepts and Techniques, Morgan Kaufmann.

**Hansen, L.K. and Salomon, P.,** 1990**.** Neural network ensembles, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12 (10), 993—1000.

**Hashem, S.,** 1997**.** Optimal linear combination of neural networks, *Neural Networks*, 10 (4), 599-614.

**Hastie, T., Tibshirani, R. and Friedman, J.,** 2001. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series.

**Ho, H.S,** 1994. Decision combination in multiple classifier systems, *IEEE Transactions on Pat5tern Analysis and Machine Intelligence*, 66-75.

**Holte, R.C.,** 1993**.** Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, **11**, 63-91.

Huang, Y. and Suen, C., **1994**. **A method of combining multiple classifiers — a neural network approach, *Proceedings of the 12th International Conference on Pattern Recognition*, 473-475.**

Kang, H.J., Kim, K. and Kim, J.H., **1997. Optimal approximation of discrete probability distribution with kth-order dependency and its application to combining multiple classifiers, *Pattern Recognition Letters*, 18 (6), 515—523.**

Keerthi, S.S., Shevade, S.K., Bhattacharyya, C. and Murthy, K.R.K., **2001. *Improvements to Platt's SMO Algorithm for SVM Classifier Design*, Neural Computation, 13(3), 637-649.**

**Kittler, J., Hatef, M., Duin, R.P.W. and Matas, J.,** 1998. On combining classifiers, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 20 (3), 226-239.

**Kohavi, R.,** 1995**.** Wrappers for Performance Enhancement and Oblivious Decision Graphs, *PhD Thesis*, Department of Computer Science, Stanford University.

**Kohavi, R.,** 1995. The Power of Decision Tables, *Proceedings of the 8th European Conference on Machine Learning*, 174-189

**Kohavi, R.,** 1996**.** Scaling up the Accuracy of Naive-Bayes Classifiers: A Decision Tree Hybrid. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 202-207

**Lam, L. and Suen, C.Y.,** 1995**.** Optimal combination of pattern classifiers, *Pattern Recognition Letters*, 16 (9), 945-954.

Landwehr, N., Hall, M. and Frank E., **2003**. **Logistic Model Trees,** ***European Conference on Machine Learning (ECML-2003),*** **241-252**

**Liu, W. and Bundy, A.,** 1992. The combination of different pieces of evidence using incidence calculus. *Technical Report* RP 599, Department of Artificial Intelligence, University of Edinburgh.

**Mahajani, G.A. and Aslandogan, Y.A.,** 2003. Evidence Combination in Medical Data Mining, *Technical Report* CSE-2003-23, Department of Computer Science and Engineering, University of Texas at Arlington.

**Martin, B.,** 1995**.** Instance-Based learning : Nearest Neighbor With Generalization, *Master Thesis*, University of Waikato, Hamilton, New Zealand

**Melville, P. and Mooney, R.** 2003**.** Constructing diverse classifier ensembles using artificial training examples, *Proceedings of 18th International Joint Conference on Artificial Intelligence*, 505-510.

**Merz, C. J.** 1999**.** Using correspondence analysis to combine classifiers. *Machine Learning*, 36, 33—58.

**Murphy, P.M. and Alia, D. W.,** 1994. UCI repository of machine leaning databases (machine-readable data repository - http://www.ics.uci.edu/~mlearn/

MLRepository.html). University of California.-Irvine, Department of Information and Computer Science.

**Perrone, M.P. and Cooper, L.N.,** 1993**.** When networks disagree: Ensemble methods for hybrid neural networks, In R.J. Mammone (eds), *Neural Networks for Speech and Image Processing*, Chapman-Hall.

**Platt, J.,** 1998**.** Fast Training of Support Vector Machines using Sequential Minimal Optimization. Advances in Kernel Methods - *Support Vector Learning,* MIT Press.

**Quinlan, R.,** 1993**.** C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA

**Seewald, A.K. and Furnkranz, J.,** 2001**.** An Evaluation of Grading Classifiers, *Proceedings of 4th International Symposium on Advances in Intelligent Data Analysis*, 115-124.

**Seewald, A.K.,** 2002. How to Make Stacking Better and Faster While Also Taking Care of an Unknown Weakness, *Proceedings of the Nineteenth International Conference on Machine Learning***,** 554-561.

Shafer, G., **1976. A Mathematical Theory of Evidence, Princeton University Press.**

**Smets, Ph.,** 1990**.** What is Dempster-Shafer's model?

**Smets, Ph.,** 1998**.** The transferable belief model for quantified belief representation

**Smets, Ph.,** 1992**.** The Transferable Belief Model And Random Sets.

**Ting, K. M. and Witten, I.H.,** 1999**.** Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 271-289.

**Todorovski, L. and Dzeroski, S.,** 2000. Combining multiple models with meta decision trees, *Proceedings of the Fourth European Conference on Principles of Data Mining and Knowledge Discovery*, 54-64.

**Todorovski, L. and Dzeroski, S.,** 2003. Combining classifiers with meta decision trees, *Machine Learning*, 223-249.

**Turner, K. and Ghosh, J.,** 1996. Analysis of Decision Boundaries in Linearly Combined Neural Classifiers, *Pattern Recognition*, 29 (2), 341-348.

Wachowicz, M. and Carvalho, L.M.T., **2002**. **Data Fusion And Mining For The Automation Of A Space-Time Reasoning** ***Process. 4th International Conference on Fusion of Earth Data*****, French Riviera, France.**

**Waltz, E.,** 1999**.** Information Understanding: Integrating Data Fusion and Data Mining Processes, in workshop along with *IEEE 1999 International Symposium on Signals and Systems*.

**White, Jr., F.E**., 1987**.** Data Fusion Lexicon, Joint Directors of Laboratories, *Technical Panel for C3, Data Fusion Sub-Panel*, Naval Ocean Systems Center, San Diego.

**Witten, I.H. and Frank, E.,** 2000**.** Data Mining: Practical machine learning tools with Java implementations, Morgan Kaufmann, San Francisco.

**Wolpert, D.,H.,** 1992. Stacked generalization. *Neural Networks*, 5:241-259.

Woods, K., Bowyer, K. and Kegelmeyer, W. P., **1997. Combination of multiple classifiers using local accuracy,** ***IEEE Transactions on Pattern Analysis and Machine Intelligence*****, 19 (4), 405-410.**

Wu, H., Siegel, M., Stiefelhagen, R. and Yang, J., **2002**. **Sensor Fusion Using Dempster-Shafer Theory,** ***IEEE Instrumentation and Measurement Technology Conference*****, Anchorage, AK, USA.**

**Xu, L., Krzyzak, A. and Suen, C.Y.,** 1992**.** Methods of combining multiple classifiers and their applications to handwriting recognition, *IEEE Transactions on Systems, Man and Cybernetic*, 22 (3), 418-435.

**Zenko, B., Todorovski, L. and Dzeroski, S.,** 2001. A comparison of stacking with MDTs to bagging, boosting, and other stacking methods. *Proceedings of the First IEEE International Conference on Data Mining*, 669-670

# A. Appendix: The Characteristics of the Data Sets Used in the Experiments

## A.1 Autos

This data set consists of three types of entities:

- the specification of an auto in terms of various characteristics,
- its assigned insurance risk rating,
- its normalized losses in use as compared to other cars.

Cars are initially assigned a risk factor symbol associated with its price. Then, if it is more risky (or less), this symbol is adjusted by moving it up (or down) the scale. Actuarians call this process "symboling". A value of +3 indicates that the auto is risky, -3 that it is probably pretty safe.

The third factor is the relative average loss payment per insured vehicle year. This value is normalized for all autos within a particular size classification (two-door small, station wagons, sports/speciality, etc...), and represents the average loss per car per year.

Number of Instances is 205 and number of attributes is 26 in total. The types of the attributes are: 15 continuous, 1 integer, 10 nominal.

The attribute characteristics of the data set "Autos" is shown in Table A.1. Missing attribute values in this data set are denoted by "?" and these are shown in Table A.2.

## A.2 Breast-Cancer-Wisconsin

This data set consists of 699 number of instances and 11 attributes. The attribute characteristics of the data set "Breast-Cancer-Wisconsin" are shown in Table A.3.

Class distribution of the data set "Breast-Cancer-Wisconsin" is as follows:

| | | |
|---|---|---|
| Benign | : 458 | (65.5%) |
| Malignant | : 241 | (34.5%) |

### A.3 Heart-Disease-Cleveland

This database contains 76 attributes, but all published experiments refer to using a subset of 14 of them.

**Table A.1** Attribute characteristics for the “Autos” data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | Symboling | -3, -2, -1, 0, 1, 2, 3 |
| 2 | normalized-losses | Continuous from 65 to 256 |
| 3 | Make | alfa-romeo, audi, bmw, chevrolet, dodge, honda, isuzu, jaguar, mazda, mercedes-benz, mercury, mitsubishi, nissan, peugot, plymouth, porsche, renault, saab, subaru, toyota, volkswagen, volvo |
| 4 | fuel-type | diesel, gas |
| 5 | Aspiration | std, turbo |
| 6 | num-of-doors | four, two |
| 7 | body-style | hardtop, wagon, sedan, hatchback, convertible |
| 8 | drive-wheels | 4wd, fwd, rwd |
| 9 | engine-location | front, rear |
| 10 | wheel-base | Continuous from 86.6 120.9 |
| 11 | Length | Continuous from 141.1 to 208.1 |
| 12 | Width | Continuous from 60.3 to 72.3 |
| 13 | Height | Continuous from 47.8 to 59.8 |
| 14 | curb-weight | Continuous from 1488 to 4066 |
| 15 | engine-type | dohc, dohcv, l, ohc, ohcf, ohcv, rotor |
| 16 | num-of-cylinders | eight, five, four, six, three, twelve, two |
| 17 | engine-size | Continuous from 61 to 326 |
| 18 | fuel-system | 1bbl, 2bbl, 4bbl, idi, mfi, mpfi, spdi, spfi |
| 19 | Bore | Continuous from 2.54 to 3.94 |
| 20 | Stroke | Continuous from 2.07 to 4.17 |
| 21 | compression-ratio | Continuous from 7 to 23 |
| 22 | Horsepower | Continuous from 48 to 288 |
| 23 | peak-rpm | Continuous from 4150 to 6600 |
| 24 | city-mpg | Continuous from 13 to 49 |
| 25 | highway-mpg | Continuous from 16 to 54 |
| 26 | Price | Continuous from 5118 to 45400 |

**Table A.2** Missing attribute information for the “Autos” data set

| Attribute | Number of instances missing a value |
|---|---|
| normalized-losses | 41 |
| num-of-doors | 2 |
| bore | 4 |
| stroke | 4 |
| horsepower | 2 |
| peak-rpm | 2 |
| price | 4 |

There are 303 instances in the Cleveland heart disease database. The number of attributes, including the predicted attribute, is 76. The attribute information belonging to generally used attributes is shown in Table A.4. There exist several missing attribute values. These are distinguished with -9.0.

Class distribution for the Cleveland heart disease database is:

| 0 | 1 | 2 | 3 | 4 | Total |
|---|---|---|---|---|---|
| 164 | 55 | 36 | 35 | 13 | 303 |

**Table A.3** Attribute characteristics for the “Breast-Cancer-Wisconsin” data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | Sample code number | id number |
| 2 | Clump Thickness | 1 - 10 |
| 3 | Uniformity of Cell Size | 1 – 10 |
| 4 | Uniformity of Cell Shape | 1 – 10 |
| 5 | Marginal Adhesion | 1 – 10 |
| 6 | Single Epithelial Cell Size | 1 – 10 |
| 7 | Bare Nuclei | 1 - 10 |
| 8 | Bland Chromatin | 1 – 10 |
| 9 | Normal Nucleoli | 1 – 10 |
| 10 | Mitoses | 1 – 10 |
| 11 | Class | 1 - 10 |

**Table A.4** Attribute information for the Cleveland heart disease database

| Attribute No | Attribute | Attribute Range |
|---|---|---|

| | | |
|---|---|---|
| 1 | age | |
| 2 | sex | 1:male,0:female |
| 3 | Cp:chest pain type | 1:typical angina<br>2:atypical angina<br>3:non-anginal pain<br>4:asymptomatic |
| 4 | Trestbps: resting blood pressure | |
| 5 | chol: serum cholestoral in mg/dl | |
| 6 | fbs: fasting blood sugar > 120 mg/dl | 1:true<br>0:false |
| 7 | restecg: resting electrocardiographic results | 0: normal<br>1: having ST-T wave abnormality<br>2:showing probable or definite left ventricular hypertrophy by Estes' criteria |
| 8 | thalach: maximum heart rate achieved | |
| 9 | exang: exercise induced angina | 1:yes<br>0:no |
| 10 | oldpeak = ST depression induced by exercise relative to rest | |
| 11 | slope: the slope of the peak exercise ST segment | 1: upsloping<br>2: flat<br>3: downsloping |
| 12 | ca: number of major vessels (0-3) colored by flourosopy | |
| 13 | thal | 3:normal<br>6:fixed defect<br>7:reversable defect |
| 14 | num: diagnosis of heart disease | 0: < 50% diameter narrowing<br>1: > 50% diameter narrowing |

### A.4 Heart-Disease-Hungary

This database contains 76 attributes, a subset of 14 of them is generally used by researchers. The "goal" field refers to the presence of heart disease in the patient. It is integer valued from 0 (no presence) to 4. Our experiments with the Hungary heart disease database have concentrated on simply attempting to distinguish presence (values 1,2,3,4) from absence (value 0).

There exist 294 instances in the Hungarian heart disease database. The number of attributes, including the predicted attribute, is 76. The attribute information belonging to generally used attributes is the same as the Cleveland heart disease database shown in Table A.4. There are several missing attribute values which are distinguished with -9.0.

Class distribution for the Hungarian heart disease database is shown below:

| Class | 0 | 1 | 2 | 3 | 4 | Total |
|---|---|---|---|---|---|---|
| Number | 188 | 37 | 26 | 28 | 15 | 294 |

## A.5 Hepatitis

This database contains 20 attributes including the class attribute. There are 155 instances in the Hepatitis database. The attribute information is shown in Table A.5. There exist some missing attribute values which are shown in Table A.6

Class distribution for the Hungarian heart disease database is shown below:

| | |
|---|---|
| DIE | : 32 |
| LIVE | : 123 |

## A.6 Iris

The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are not linearly separable from each other. The predicted attribute is the class of iris plant. Number of Instances is 150 . There are 50 instances in each of three classes.

Attribute Information for the “Iris” data set is presented in Table A.7. As seen from the table sepal length, sepal with, petal length and petal with are expressed in centimeters.

There are no missing attribute values.

Class distribution for the Iris data set is 33.3% for each of following 3 classes:

- Iris Setosa
- Iris Versicolour
- Iris Virginica

**Table A.5** Attribute characteristics for the “Hepatitis” data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | Class | DIE, LIVE |
| 2 | AGE | 10, 20, 30, 40, 50, 60, 70, 80 |

| | | |
|---|---|---|
| 3 | SEX | male, female |
| 4 | STEROID | no, yes |
| 5 | ANTIVIRALS | no, yes |
| 6 | FATIGUE | no, yes |
| 7 | MALAISE | no, yes |
| 8 | ANOREXIA | no, yes |
| 9 | LIVER BIG | no, yes |
| 10 | LIVER FIRM | no, yes |
| 11 | SPLEEN PALPABLE | no, yes |
| 12 | SPIDERS | no, yes |
| 13 | ASCITES | no, yes |
| 14 | VARICES | no, yes |
| 15 | BILIRUBIN | 0.39, 0.80, 1.20, 2.00, 3.00, 4.00 |
| 16 | ALK PHOSPHATE | 33, 80, 120, 160, 200, 250 |
| 17 | SGOT | 13, 100, 200, 300, 400, 500 |
| 18 | ALBUMIN | 2.1, 3.0, 3.8, 4.5, 5.0, 6.0 |
| 19 | PROTIME | 10, 20, 30, 40, 50, 60, 70, 80, 90 |
| 20 | HISTOLOGY | no, yes |

### A.7 Labor

This database contains 16 attributes including the class attribute. There are 57 instances in the Labor database. There exist no missing attribute values. The attribute information is as in Table A.8.

### A.8 Soybean

The Soybean database includes 35 attributes There are 307 instances in the data set. The attribute information is shown in Table A.9. The number of missing attribute values denoted by "?" is presented in Table A.10.

**Table A.6** Missing attribute information for the “Hepatitis” data set

| Attribute | Number of instances missing a value |
|---|---|
| STEROID | 1 |
| FATIGUE | 1 |
| MALAISE | 1 |
| ANOREXIA | 1 |
| LIVER BIG | 10 |
| LIVER FIRM | 11 |
| SPLEEN PALPABLE | 5 |
| SPIDERS | 5 |
| ASCITES | 5 |
| VARICES | 5 |
| BILIRUBIN | 6 |
| ALK PHOSPHATE | 29 |
| SGOT | 4 |
| ALBUMIN | 16 |
| PROTIME | 67 |

The Soybean data set includes 19 classes which are diaporthe-stem-canker, charcoal-rot, rhizoctonia-root-rot, phytophthora-rot, brown-stem-rot, powdery-mildew, downy-mildew, brown-spot, bacterial-blight, bacterial-pustule, purple-seed-stain, anthracnose, phyllosticta-leaf-spot, alternarialeaf-spot, frog-eye-leaf-spot, diaporthe-pod-&-stem-blight, cyst-nematode, 2-4-d-injury, herbicide-injury.

The class distribution is displayed in Table A.11.

**Table A.7** Attribute characteristics for the “Iris” data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | sepal length | in cm |
| 2 | sepal width | in cm |
| 3 | petal length | in cm |
| 4 | petal width | in cm |
| 5 | class | Iris Setosa<br>Iris Versicolour<br>Iris Virginica |

**Table A.8** Attribute information for the “Labor” data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | dur: duration of agreement | [1..7] |
| 2 | wage1.wage : wage increase in first year of contract | [2.0 .. 7.0] |
| 3 | wage2.wage : wage increase in second year of contract | [2.0 .. 7.0] |
| 4 | wage3.wage : wage increase in third year of contract | [2.0 .. 7.0] |
| 5 | cola : cost of living allowance | [none, tcf, tc] |
| 6 | hours.hrs : number of working hours during week | [35 .. 40] |
| 7 | pension : employer contributions to pension plan | [none, ret_allw, empl_contr] |
| 8 | stby_pay : standby pay | [2 .. 25] |
| 9 | shift_diff : shift differencial : supplement for work on II and III shift | [1 .. 25] |
| 10 | educ_allw.boolean : education allowance | [true false] |
| 11 | holidays : number of statutory holidays | [9 .. 15] |
| 12 | vacation : number of paid vacation days | [ba, avg, gnr] |
| 13 | lngtrm_disabil.boolean :employer's help during employee longterm disability | [true , false] |
| 14 | dntl_ins : employers contribution towards the dental plan | [none, half, full] |
| 15 | bereavement.boolean : employer's financial contribution towards the covering the costs of bereavement | [true , false] |
| 16 | empl_hplan : employer's contribution towards the health plan | [none, half, full] |

## A.9 Thyroid

The Soybean database includes 6 attributes There are 215 instances in the data set. The attribute information is shown in Table A.12. All attributes are continuous. There are no missing attribute values.

Number of instances per class is as follows:

| | |
|---|---|
| • Class 1: (normal) | 150 |
| • Class 2: (hyper) | 35 |
| • Class 3: (hypo) | 30 |

**Table A.9** Attribute information for the "Soybean" data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | date | april, may, june, july, august, september, october |
| 2 | plant-stand | normal,lt-normal |
| 3 | precip | lt-norm,norm,gt-norm |
| 4 | temp | lt-norm,norm,gt-norm |
| 5 | hail | yes,no |
| 6 | crop-hist | diff-lst-year, same-lst-yr, same-lst-two-yrs, same-lst-sev-yrs |
| 7 | area-damaged | scattered,low-areas, upper-areas, whole-field |
| 8 | severity | minor,pot-severe,severe |
| 9 | seed-tmt | none,fungicide,other |
| 10 | germination | 90-100%,80-89%,lt-80% |
| 11 | plant-growth | norm,abnorm |
| 12 | leaves | norm,abnorm |
| 13 | leafspots-halo | absent, yellow-halos, no-yellow-halos |
| 14 | leafspots-marg | w-s-marg,no-w-s-marg,dna |
| 15 | leafspot-size | lt-1/8,gt-1/8,dna |
| 16 | leaf-shread | absent,present |
| 17 | leaf-malf | absent,present |
| 18 | leaf-mild | absent,upper-surf,lower-surf |
| 19 | stem | norm,abnorm |
| 20 | lodging | yes,no |
| 21 | stem-cankers | absent,below-soil,above-soil,above-sec-nde |
| 22 | canker-lesion | dna,brown,dk-brown-blk,tan |
| 23 | fruiting-bodies | absent,present |
| 24 | external decay | absent,firm-and-dry,watery |
| 25 | mycelium | absent,present |
| 26 | int-discolor | none,brown,black |
| 27 | sclerotia | absent,present |
| 28 | fruit-pods | norm,diseased,few-present,dna |
| 29 | fruit spots | absent,colored,brown-w/blk-specks, distort, dna |
| 30 | seed | norm,abnorm |
| 31 | mold-growth | absent,present |
| 32 | seed-discolor | absent,present |
| 33 | seed-size | norm,lt-norm |
| 34 | shriveling | absent,present |
| 35 | roots | norm,rotted,galls-cysts |

**Table A.10** Missing attribute information for the “Soybean” data set

| Attribute No | Attribute | Missing |
|---|---|---|
| 1 | date | 0 |
| 2 | plant-stand | 1 |
| 3 | precip | 8 |
| 4 | temp | 11 |
| 5 | hail | 7 |
| 6 | crop-hist | 41 |
| 7 | area-damaged | 1 |
| 8 | severity | 1 |
| 9 | seed-tmt | 41 |
| 10 | germination | 41 |
| 11 | plant-growth | 36 |
| 12 | leaves | 1 |
| 13 | leafspots-halo | 0 |
| 14 | leafspots-marg | 25 |
| 15 | leafspot-size | 25 |
| 16 | leaf-shread | 25 |
| 17 | leaf-malf | 26 |
| 18 | leaf-mild | 25 |
| 19 | stem | 30 |
| 20 | lodging | 1 |
| 21 | stem-cankers | 41 |
| 22 | canker-lesion | 11 |
| 23 | fruiting-bodies | 11 |
| 24 | external decay | 35 |
| 25 | mycelium | 11 |
| 26 | int-discolor | 11 |
| 27 | sclerotia | 11 |
| 28 | fruit-pods | 11 |
| 29 | fruit spots | 25 |
| 30 | seed | 35 |
| 31 | mold-growth | 29 |
| 32 | seed-discolor | 29 |
| 33 | seed-size | 35 |
| 34 | shriveling | 29 |
| 35 | roots | 35 |

## A.10 Wine

These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. There are 178 instances in the data set.  All attributes are continuous. There are no missing attribute values.

Class distribution is as follows:

| | |
|---|---|
| • class 1 | 59 |
| • class 2 | 71 |
| • class 3 | 48 |

**Table A.11** Class distribution for the “Soybean” data set

| Class No | Class | Number |
|---|---|---|
| 1 | diaporthe-stem-canker | 10 |
| 2 | charcoal-rot | 10 |
| 3 | rhizoctonia-root-rot | 10 |
| 4 | phytophthora-rot | 40 |
| 5 | brown-stem-rot | 20 |
| 6 | powdery-mildew | 10 |
| 7 | downy-mildew | 10 |
| 8 | brown-spot | 40 |
| 9 | bacterial-blight | 10 |
| 10 | bacterial-pustule | 10 |
| 11 | purple-seed-stain | 10 |
| 12 | anthracnose | 20 |
| 13 | downy-mildew | 10 |
| 14 | brown-spot | 40 |
| 15 | bacterial-blight | 40 |
| 16 | bacterial-pustule | 6 |
| 17 | purple-seed-stain | 6 |
| 18 | anthracnose | 1 |
| 19 | herbicide-injury | 4 |

**Table A.12** Attribute information for the “Thyroid” data set

| Attribute No | Attribute | Attribute Range |
|---|---|---|
| 1 | Class attribute | 1:normal,<br>2:hyper,<br>3:hypo |
| 2 | T3-resin uptake test | a percentage |
| 3 | Total Serum thyroxin as measured by the isotopic displacement method | |
| 4 | Total serum triiodothyronine as measured by radioimmuno assay | |
| 5 | basal thyroid-stimulating hormone (TSH) as measured by radioimmuno assay | |
| 6 | Maximal absolute difference of TSH value after injection of 200 micro grams of thyrotropin-releasing hormone as compared to the basal value | |

## BIOGRAPHY

Hüseyin Aygün graduated from the Turkish Naval High School in 1981. He received his B.S. degree from the Operations Research Department, Turkish Naval Academy in 1985. He received his M.Sc. in 1991 from the Computer Science Department, Naval Postgraduate School, Monterey, CA. He worked for the NATO Maritime Command Control and Communication and Information System as Computer Security Officer for five years. He has been teaching several courses at the Computer Engineering Department of the Turkish Naval Academy since 1998. He enrolled in the Ph.D. program of the Institute of Science and Technology of Istanbul Technical University in 2001. His research interests include data fusion, data mining, stream data mining, more specifically classification using Dempster's Rule of Combination.

Publications related to the Ph.D. study:

- Aygün, H.,Adalı, E., Combining Classifier Outputs by Assigning Degree of Confidence, *Proc. of Second International Conference on Intelligent Computing and Information Systems,* Cairo, Egypt, March 5-7, 2005, pp. 506-511.
- Aygün, H.,Adalı, E., Combining Classification Algorithms Using Dempster's Rule of Combination, *18th International Conference on Computer Applications in Industry and Engineering,* Honolulu, Hawaii, Nov. 9-11, 2005.
- Aygün, H.,Adalı, E., 2005, Dempster'in Birleştirme Kuralı ile Sınıflandırma Algoritmalarının Birleştirilmesi, *TBV Bilgisayar Bilimleri ve Mühendisliği Dergisi,1*,pp.99-103.
- Aygün, H.,Adalı, E., 2006, A Novel Algorithm for Combining Classification Algorithms, *Deniz Bilimleri ve Mühendisliği Dergisi* (to be published).
- Aygün, H.,Adalı, E., 2006, Dempster'in Birleştirme Algoritması ile Sınıflandırıcı Sonuçlarının Birleştirilmesi, *itüdergisi/d-mühendislik* (to be published).