Evidential Bagging: Combining Heterogeneous Classifiers in the Belief Functions Framework

Nicolas Sutton-Charani, Abdelhak Imoussaten, Sébastien Harispe, Jacky Montmain

Evidential Bagging: Combining Heterogeneous Classifiers in the Belief Functions Framework

Nicolas Sutton-Charani, Abdelhak Imoussaten, Sébastien Harispe, and Jacky Montmain

LGI2P, IMT Mines Ales, Univ Montpellier, Ales, France
{firstname.name}@mines-ales.fr
http://lgi2p.mines-ales.fr

Abstract. In machine learning, *Ensemble Learning* methodologies are known to improve predictive accuracy and robustness. They consist in the learning of many classifiers that produce outputs which are finally combined according to different techniques. *Bagging*, or Bootstrap Aggregating, is one of the most famous Ensemble methodologies and is usually applied to the same classification base algorithm, i.e. the same type of classifier is learnt multiple times on bootstrapped versions of the initial learning dataset. In this paper, we propose a *bagging* methodology that involves different types of classifier. Classifiers' probabilist outputs are used to build mass functions which are further combined within the belief functions framework. Three different ways of building mass functions are proposed; preliminary experiments on benchmark datasets showing the relevancy of the approach are presented.

Keywords: belief functions; information fusion; bagging; supervised learning.

1 Introduction

As the amount of learning algorithms and methodologies in the literature has reached a point where it is almost impossible to stay up to date on all of them, many users or even researchers tend to use them as black boxes, without focusing much on their understanding or interpretation, often setting model parameters to default values. Beside some obvious computational time differences between them, it has been proven that there is no *optimal* learning algorithm in the sense that most models are optimal for certain types of learning data and have advantages and drawbacks [1]. Indeed, dataset dimensions, attribute types, variance and noise make each learning dataset more suited to some learning algorithms than others. To illustrate this fact, six standard learning algorithms have been applied to six benchmark datasets and the resulting mean accuracies (i.e. the correct predictions rate) from 1000 10-fold cross validation simulations are presented in Figure 1. We can easily observe some disparities between the different types of classifier (decision trees, SVM, etc) used on those six datasets. For example, *neural network* seems to be one of the less accurate classifier on almost

all datasets except for "Balance scale" where it is the most accurate one. We also can observe that decision trees and neural network have the highest accuracy variances.

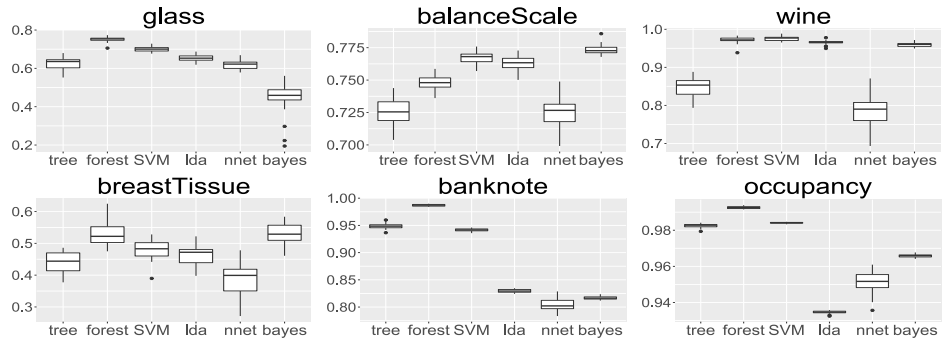

Fig. 1. Learning algorithms mean accuracies on 1000 simulations with default parameters tuning in R.

Ensemble learning methodologies consist in the learning of many classifiers from the same initial dataset. In that context, classifiers are then aggregated, or the predictions they provide (their outputs) are combined in order to get final predictions. The resulting classifier is usually more accurate and robust [2–4]. One particular ensemble method, called *bagging* or bootstrap aggregating, has an additional advantage: it tends to avoid over-fitting [3]. Bagging uses some re-sampling methods in order to decrease the dependency of classifiers on the learning data, i.e. to decrease the learning data's bias and variance. Unlike other ensemble methods as *Bucket-of-models* [2, 5], bagging usually involves the simultaneous and multiple use of a single algorithm, to further combine their predictions with a simple *vote* procedure.

This problem can be seen as an information fusion problem if we consider the trained classifiers as information sources and their predictions as the information, or evidence, to fuse. Traditionally, in bagging methods classifiers' outputs are combined through a vote procedure as they usually involve the same type of classifiers. With that approach, fusing the same type of classifiers should indeed lead to a uniform weighting.

In the evidential framework, researchers have proposed some evidential bagging methods, handling uncertain data and the combination procedure is also performed between the same type of classifiers [6–8]. To the best of our knowledge, no work has been proposed that combine heterogeneous types of classifier with belief functions.

In this paper we present a bagging method that involves the fusion of different types of classifiers' outputs, or predictions. We propose to take into account the classifiers' outputs reliability during the aggregation step within the formalism of the belief functions theory [9, 10]. This choice is motivated by its generalization power (including probabilistic and possibilistic cases) and the flexible tools it provides. In addition many approaches have been developed in different contexts,

especially for information fusion problems when we have evidence about the sources reliability [11]. In this work, we chose to use predictive performance of the classifiers as reliability evidences. Then the classifiers' probabilistic outputs and their reliabilities are used to build one predictive belief function per classifier. Finally, a suitable combination procedure is used to merge those belief functions into a global predictive mass function.

After defining our classification formalism, recalling bagging basis, the theory of belief functions with some basic fusion tools are presented in Section 2; our evidential bagging model is described in Section 3 experiments are provided on benchmark datasets in Section 4 and finally results and perspectives are discussed in Section 5.

2 Related works and positioning

In this section, first the general classification formalism is given, then a succinct overview of *Ensemble learning* methods is presented; finally *Bagging* is more precisely described.

2.1 Classification formalism

Starting from a dataset D containing N learning examples $(x, y)_{i=1,\ldots,N}$, classification tasks aim at learning a model f able to predict the *class label* y^* of any new unlabeled example from its *attribute* (or feature) values x such that $y^* = f(x)$. The attributes $X = (X^1, \ldots, X^J)$ take their values in $\mathcal{X} = \mathcal{X}^1 \times \cdots \times \mathcal{X}^J$, the class Y in a finite set Ω. Spaces $\mathcal{X}^j$ can be categorical or numerical.

$$D = \begin{pmatrix} x_1, y_1 \\ \vdots \\ x_N, y_N \end{pmatrix} = \begin{pmatrix} x_1^1 & \ldots & x_1^J & y_1 \\ \vdots & & \vdots & \vdots \\ x_N^1 & \ldots & x_N^J & y_N \end{pmatrix}.$$

Samples are assumed to be i.i.d. but in practice data are often noised, casual correlations can randomly occur and some outliers or very rare examples can occur even in very small datasets. In order to discount those outliers' influence and to reduce bias, bootstrapping techniques [12] can be applied. By doing so, overfitting can be decreased without the need of large datasets but the different bootstrapped subsamples' results (i.e. predictions) have to be aggregated in a conservative way so that the most uncertain predictions can be weakened.

In this context, a classifier is a function $c : \mathcal{X} \to \Omega$, the notation $c(x)$ corresponds to the prediction obtained by c from attribute values $x \in \mathcal{X}$.

2.2 Ensemble Learning methods

As almost each learning algorithm is optimal for specific types of learning data and problems [1], many researchers do not confine in single classifiers and tend to use many of them when dealing with real data. In this context, *Ensemble*

Learning methods have become very common tools to improve classifiers performance (in term of accuracy and robustness) by using predictive algorithms' heterogeneity and resampling techniques to avoid overfitting. In fact, Ensemble Learning has become a whole research field [4, 3] where we can distinguish three main types of methodology: *boosting*, *stacking* and *bagging*. Boosting [13] is an iterative procedure where the classifier is *re*-learnt in order to better classify misclassified examples. *AdaBoost* is the most famous boosting algorithm [14]. Stacking [15] focuses on the classifiers outputs aggregation step with the learning of an aggregation algorithm. *Bagging*, or Bootstrap-Aggregation, is based on resampling and was first presented by Breiman [16] with the well known *Random forest* algorithm which consists in the learning of several decision trees from different bootstrapped subsamples and in their aggregation through a *vote* procedure. In this paper, a *Bagging* methodology is proposed based on a bucket of diverse learning algorithms with an aggregation step formalized in the evidential framework and involving belief functions generated from the single classifiers' predictions and evaluations.

2.3 Bagging

In most applications, getting a dataset truly representative of the actual population is a complex task as sampling often implies noise intrusion and thus bias and variance creation [17]. As a matter of fact, in most learning datasets, some rare examples of the reality can be over-represented which often leads to overfitted models. During the learning process, those examples should not be given too much weight as they can be considered outliers. It is for that matter that *Bagging* involves resampling, in order to discount that kind of bias [12].

Nevertheless, in most bagging approaches, the same learning algorithm is used to fill the *bag* because of the heterogeneity of the classifiers' natural outputs (scores, class probability, etc). During the aggregation step, trainable combiners have shown some asymptotically optimal properties [18] but involves a common training set for all single classifiers. In this work we use a combiner learnt on such a validation set which will be also used for the single classifiers evaluations.

2.4 Theory of Belief Functions

The theory of *belief functions*, also known as *Evidence* or *Dempster-Shafer* theory was first presented by A. Dempster in 1967 in a statistical context [9] as an attempt to reconcile frequentists and Bayesians. Dempster laid the foundation of a mathematical theory that deals with uncertainty in a much more general framework than standard probability theory and which handles aleatory (or *objective*) uncertainties and epistemic (or *subjective*) ones.

Even if some evidential works are related to statistical and classification contexts, many researchers use belief functions as a convenient framework for information fusion problems. As a matter of fact, fusion problems occur in many contexts, even in the classification ones. In the evidential framework two levels are considered: the credal and the decision ones. The credal level is dedicated to

the uncertainty representation whereas the decision step uses the credal one to make decision.

Credal level Let Y be an uncertain quantity whose value y lies in a finite set Ω called the *frame of discernment*.

Definition 1. *A mass function m regarding Y's value $y \in \Omega$ is a function defined on the set of subsets*[1] *of Ω, which is usually written 2^Ω or $\mathcal{P}(\Omega)$ and called the powerset, with its values in $[0,1]$ and verifying $\sum_{B \in 2^\Omega} m(B) = 1$*

The quantity $m(\emptyset)$ is usually fixed to 0 in many cases. From m, two uncertainty measures can be defined about y, the *belief* function Bel and the *plausibility* function Pl, which express the information contained in m in different ways, more conservatively for Bel than for Pl. Since there is *1 to 1* correspondancies between m, Bel and Pl, belief functions can indifferently refer to a mass function m or its corresponding inferior uncertainty measure Bel.

Decision level In a practical matter, the decision step of many problems is often handled by transforming belief functions into *pignistic* probabilities [19] (cf. Definition 2) to further consider the most *probable* event in the *pignistic sense*.

Definition 2. *The pignistic probability distribution attached to a mass function m is defined by:*

$$\forall \omega \in \Omega,\ BetP(\omega) = \sum_{A \subseteq \Omega | \omega \in A} \frac{m(A)}{|A|}. \quad (1)$$

Sources reliability and evidential Fusion. Information fusion problems aim at combining different informative contents coming from different sources. In the evidential framework, sources are represented by mass functions.

Three main concepts must be taken into account when combining evidential sources: dependence, reliability and conflict. The first combination method proposed in the belief function theory was Dempster's conjunctive combination rule. Nevertheless, this rule handles conflict in a way that has been criticized [20, 21]. Moreover, this rule requires independence between sources.

To take more conveniently into account the conflict and the dependence between sources, several combination rules have been proposed [21, 20, 22]. To avoid dependence hypothesis, some works [11, 23] use the *average* operator to combine beliefs functions, which is in line with voting procedures. In this paper we chose this operator as a basis.

3 Evidential Bagging

This section presents the three evidential generative models defined in this paper, as well as the fusion approach proposed to aggregate the predictions provided by the different single classifiers. Finally the general scheme of our approach is given.

[1] The expressions $B \subseteq \Omega$ and $B \in 2^\Omega$ are equivalent.

3.1 Generative models

Even if the different learning algorithms may provide outputs of different structures, most data science software enable a calibration step in order to get a probability distribution on class labels from those outputs. Those probabilities express an uncertainty which should be integrated in any *bagging* approach that presents an uncertainty focus. Nevertheless, those uncertainty measures are computed on the learning data and are therefore subject to over-fitting. Better estimators of classifiers' performance can be computed on other dataset. In this paper we compute single classifiers' accuracy on such a separate dataset which we recall as the *validation set*.

If we consider the single classifiers as information sources, and their probabilist outputs as the uncertain information to merge, one way to evaluate the classifiers reliability is to evaluate their accuracy on the *validation set*. Our approach consists in discounting the classifiers' outputs according to their reliabilities. To this aim, belief functions are generated in order to enrich the probabilist outputs of the single classifiers. Those belief functions are based both on class probabilist predictions, and accuracies computed on validation sets. They are finally merged into a final belief function and its pignistic probability gives class predictions. Proposed models are introduced hereafter:

1. **simple discounting ($EBag_{SD}$):** using this model we consider that the less accurate classifiers should be the most discounted during the aggregation step. Otherwise stated, we consider that the *reliability* of a classifier c is a function of its global accuracy denoted acc_c (computed on the validation set). To this aim, the mass function associated to c is defined as $\forall i \in \{1, \ldots, N\}$, $\forall \omega \in \Omega$:

$$\begin{cases} m_i^c(\{\omega\}) = P_i^c(\omega) \times acc_c \\ m_i^c(\Omega) \quad = 1 - acc_c \end{cases} . \tag{2}$$

   where $P_i^c(\omega)$ stands for the probability of the class label ω provided by the classifier c on the example x_i.

2. **class dependent model ($EBag_{CD}$):** some classifiers are more accurate in predicting some specific class labels. Such accuracy variations can be observed by analysing the confusion matrix associated to each classifier. We therefore propose to take advantage of this information to spread uncertainty about classification involving specific class labels from Ω. Using this model the following definitions of mass functions are considered; $\forall c \in \{1, \ldots, C\}$, $\forall i \in \{1, \ldots, N\}$, $\forall \omega \in \Omega$:

$$\begin{cases} m_i^c(\{\omega\}) = P_i^c(\omega) \times P^c(Y = y_i | c(x_i) = y_i) \\ m_i^c(\Omega) \quad = \qquad\qquad P^c(Y \neq y_i | c(x_i) = y_i) \end{cases} . \tag{3}$$

   Note that for a given classifier c, $P^c(Y|c(X))$ is estimated by using the confusion matrix of the classifier c computed on the validation set as follows: $P^c(Y = y_i | c(x_i) = y_i) = \frac{|(x,y) \in \mathcal{X} \times \Omega : \{c(x_i) = y_i\} \cap \{Y = y_i\}|}{|(x,y) \in \mathcal{X} \times \Omega : \{c(x) = y_i\}|}$.

3. **contextual model ($EBag_{con}$):** we consider two types of regions of $\mathcal{X}$: one containing instances often misclassified and its complementary. In addition, we consider that not all classifiers will misclassify instances of the same regions of $\mathcal{X}$. We are therefore interested by the definition of a contextual model that will consider an estimation of the single classifiers' misclassification risk or *probability* of each examples. Formally, to estimate this risk, that is both dependent on the classifier and the processed instance, we consider that $\forall c \in \{1, \ldots, C\}$, we have $mis_c : \mathcal{X} \rightarrow [0, 1]$ a function used to assess the misclassification risk of a classifier c regarding a given $x_i \in \mathcal{X}$ whose predicted label is y_i, i.e. $y_i = c(x_i)$; Intuitively, higher is the estimated risk, lower the confidence on the provided class will be. Based on this function, we consider the following definition of the mass function, $\forall c \in \{1, \ldots, C\}$, $\forall i \in \{1, \ldots, N\}, \forall \omega \in \Omega$:

$$\begin{cases} m_i^c(\{\omega\}) & = P_i^c(\omega) \times (1 - mis_c(x_i)) \\ m_i^c(\Omega \setminus \{y_i\}) & = mis_c(x_i) \end{cases} . \tag{4}$$

   We consider that mis_c is obtained using a binary classifier (SVM in our case) that will be trained using examples defined by the error and success classifications of c during the validation phase with a regression approach. Otherwise stated the classifier predicts a numerical output in $[0, 1]$ standing for the misclassifications observed in the validation set (1 for misclassified examples and 0 for well classified ones).

Whereas generative models $EBag_{SD}$ and $EBag_{CD}$ are based on the single classifiers global accuracy, model $EBag_{con}$ uses their local misclassification risk to discount their predictions. Therefore, the focal elements of models $EBag_{SD}$ and $EBag_{CD}$ are the class labels and the frame of discernment Ω. Model $EBag_{con}$ considers the class labels and their complementary sets - intuitively, if a single classifier misclassifies an instance, our belief should be focused on its prediction's complementary.

3.2 Combination and prediction

Classifiers' mass combination: As evidential independence is needed to apply Dempster's combination rule and disjunctive semantic has no sense in case of different classifiers predicting different class labels, the aggregation step of our model was done by averaging the mass functions generated by the single classifiers. Actually, in the evidential bagging contents, most authors have combined the classifiers' outputs through a vote procedure or with the average operator applied to their class probabilities because of the dependencies between classifiers [24, 6, 25, 8].

Final prediction: At the decision or prediction step, we chose the most likely class labels in the *pignistic* sense, i.e. the average belief function (computed from all single classifiers outputs) is transformed into its *pignistic* probability and then the most likely class label is predicted.

$$prediction(m_i^c) = \underset{\omega \in \Omega}{\operatorname{argmax}}\, BetP_i^c(\{\omega\}).$$

3.3 Global procedure

Considering a set of classifiers, and a set of labelled data, the global procedure of our evidential bagging model is composed of four main steps illustrated in Fig. 2:

1 ***Training*** of each classifier. This step is used to estimate the best parameters of each classifiers based on training data. A set of trained classifiers, i.e. tuned models, is obtained.

2 ***Belief function generation*** through a post-analysis of training performance evaluations: the step used to compute data that will be used to define the mass functions. The treatments applied for each model vary; all rely on the analysis of the single classifiers' performance on the validation set. For the simple discounting model (Eq. 2), the accuracy of each classifier is computed on the validation set. For the class dependent model (Eq. 3) the confusion matrix is computed on the validation set. The conditional probabilities that will be used to define the mass functions are estimated based on the confusion matrix. Finally, using the contextual model (Eq. 4) the misclassification risk is estimated by training a SVM classifier in the aim of distinguishing cases for which the classifier fails to provide a good classification.

3 ***Mass combination***: given an evidential generative model, the various evidential predictions provided by the classifiers are combined to output a single global mass function with the *average* operator.

4 ***Prediction*** (or decision) step: the global mass function is transformed into its corresponding pignistic probability in order to predict the most *probable* class labels.

4 Experiments

In this section, we evaluate our evidential bagging approach on several UCI[1], Kaggle[2] and KEEL[3] benchmark datasets. Number of examples, attributes and class labels of those datasets are summarized in Table 1.

The considered learning algorithms were: decision tree (*'tree'*), random forest (*'forest'*), support vector machine (*'SVM'*), linear discriminant analysis (*'lda'*) and naive Bayes classifier (*'bayes'*). The implementation was handled in R with the following functions (and packages)*rpart* (rpart), *randomForest* (randomForest), *svm* (e1071), *nnet* (nnet), *naiveBayes* (e1071) and *lda* (MASS). All the

[1] https://archive.ics.uci.edu/ml/datasets.html
[2] https://www.kaggle.com/datasets
[3] http://sci2s.ugr.es/keel/category.php?cat=clas

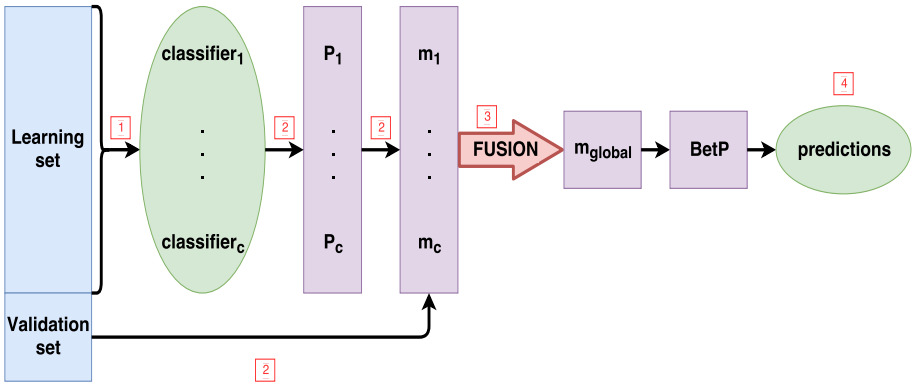

Fig. 2. Global process

dataset	N	J	K
Balance scale	625	3	3
Banana	5300	2	2
Banknote	1372	4	2
Breast tissue	106	9	6
Contraceptive method	1473	9	3
E.coli	336	5	8
Glass	214	9	6
Iris	150	4	3
Mammographic	830	5	2
Nursery	12958	8	4
Occupancy	8143	6	2
Pima	768	8	2
Satimage	6435	36	6
Tic tac toe	958	9	2
Titanic	2201	3	2
Wine	178	13	3

Table 1. Number of examples (N), attributes (J) and class labels (K) of several benchmark datasets from *UCI*, *Kaggle* and *KEEL*

learning algorithms were implemented in R with their default parameters. For each dataset, 100 10-fold cross validation procedures were implemented for different bagging methods involving different aggregation approaches:

- *voteBag:* simple vote procedure from the precise single classifiers' predictions
- P_{mean}*:* averaging of the single classifiers' probabilistic predictions and prediction of most probable class label
- $EBag_{SD}$*: simple discounting* model (see Eq. (2))
- $EBag_{CD}$*: class dependent model* (see Eq. (3))
- $EBag_{con}$*: contextual model* (see Eq. (4))

For each fold, a learning set and a validation set were built as follows: a quarter of the examples contained in the nine other folds were randomly selected as the validation set; the remaining examples were then used to build the learning set. The accuracy means and standard deviations results are summarised in Table

2. T-tests were performed between the *voteBag* models and the most accurate ones, bold accuracies stand for the significantly highest ones. R implementations of tested methods, links to datasets, as well as complete technical details about the evaluation are provided at https://github.com/lgi2p/evidentialBagging.

dataset	$voteBag$	P_{mean}	$EBag_{SD}$	$EBag_{CD}$	$EBag_{con}$
Balance scale	**0.767**	0.765	0.765	0.759	0.764
Banana	0.759	0.808	0.816	0.820	**0.830**
Banknote	0.895	0.904	0.912	**0.923**	0.918
Breast tissue	0.519	0.519	0.524	0.512	**0.538**
Contraceptive method	0.558	0.559	0.560	0.559	**0.561**
E. Coli	0.864	0.863	0.864	0.658	**0.866**
Glass	0.689	0.688	0.690	0.675	**0.700**
Iris	0.959	0.959	**0.959**	0.960	0.959
Mammographic	0.833	0.834	0.834	0.835	**0.836**
Nursery	0.957	**0.969**	0.970	0.968	0.970
Occupancy	0.983	0.983	0.983	0.984	**0.984**
Pima	0.766	0.765	0.765	0.712	0.766
Satimage	**0.881**	0.873	0.875	0.880	0.881
Tic tac toe	0.810	0.838	0.846	**0.887**	0.851
Titanic	0.782	0.783	0.783	0.782	0.783
Wine	0.978	0.976	0.977	0.978	0.978

Table 2. Mean accuracies on 30 10-fold cross validation procedures

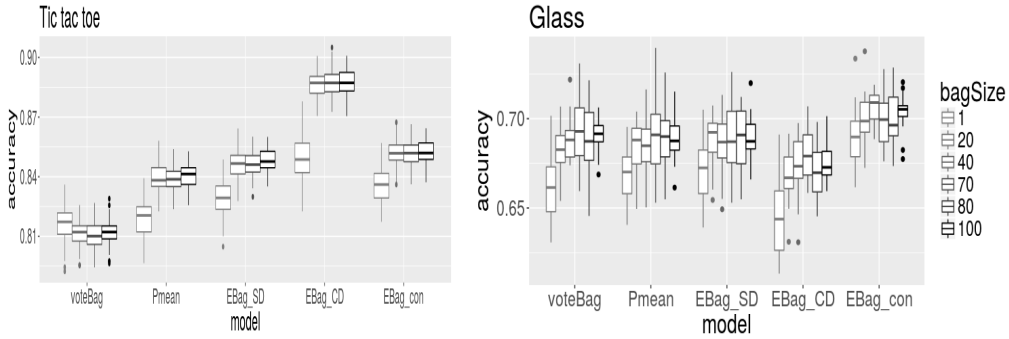

Fig. 3. Bag size effect

Globally, there is no systematically and significantly outperformance between models even if $EBag_{con}$ seems to be the most accurate model. This is not too surprising as the general spirit of classifiers is to link attributes (i.e. context) and class labels, whereas inferring global reliability or treating class labels non-symmetrically is not at the basis of standard classification tasks. $EBag_{CD}$ model is the less accurate and robust one for the datasets (Breast Tissue, E.Coli and Glass) containing the more class labels (i.e. for $K \geq 6$). This suggests that expressing single classifiers' reliability on specific class labels has a sense for limited number of labels.

As in any bagging methods, the number of bagged classifiers has an impact on the resulting classifier's accuracy, bigger bags implying higher accuracies. In Figure 3, the accuracies over 100 10-fold cross validations are represented for different bag sizes (1 to 100 per learning algorithm) on the 'Tic tac toe' and 'Glass' datasets. For the dataset 'Tic tac toe', the most accurate model is $EBag_{CD}$ whereas it is $EBag_{con}$ for the dataset 'Glass'. This corroborates the fact that

our *class-dependent* model ($EBag_{CD}$) is more accurate on dataset containing few class labels (2 for 'Tic tac toe', 6 for 'Glass'). As enhanced by Table 2, for datasets containing many class labels, our *contextual model* ($EBag_{con}$) is preferable.

5 Conclusions and perspectives

A general *bagging* approach has been proposed that involves belief mass generation from each classifiers and evidential fusion between classifiers' evidential predictions. Different aspects of the classifiers (global accuracy, confusion matrix or local misclassification risk) can be separately evaluated on a validation set and used to generate those belief functions. Experiments on benchmark datasets show encouraging results especially for the model $EBag_{con}$, which is based on a misclassification risk learnt on a separated validation set and depending on attributes values. That approach can refer to the concepts of *taking into account the learning context* or *learning the local context* (before the actual learning process) and should be further studied.

A more complete study of this evidential bagging approach should include a global sensitivity analysis over the type of classifiers to bag, the combination rule (between single classifiers' evidential predictions) and the prediction, or *decision* step. It is noticeable that considering the pignistic transform after averaging the mass functions is equivalent to some straightforward probabilistic modelling. Nevertheless, since probabilities are some particular belief functions, the evidential framework provides many tools in term of fusion and decision for any future extension. In some recent works [26, 27], likelihood-based tools have been presented that could represent an alternative to pignistic transformation. From an optimisation point of view, some clustering over class labels should improve the $EBag_{CD}$ model (i.e. the class dependent one). Moreover, ideas behind the three presented evidential generative models could be used to define a single model. In addition, in this paper, one of the used single classifier is based on a bagging approach: the random forest. By doing so, we actually made a *second order* bagging (or bagging of classifiers bags). To go further on this aspect, mathematical properties of bagging approaches should be taken into account in order to solve one pragmatic dilemma: should we make larger bags or should we nest single classifiers bags?

References

1. Wolpert, D.H. In: The Supervised Learning No-Free-Lunch Theorems. Soft Computing and Industry: Recent Applications, Springer, 25–42, London (2002)
2. Qu, G., Wu, H.: Bucket learning: Improving model quality through enhancing local patterns. Knowledge-Based Systems **27, 51–59** (2012)
3. Zhou, Z.H.: Ensemble Methods: Foundations and Algorithms. 1st edn. Chapman & Hall/CRC (2012)
4. Polikar, R.: Ensemble based systems in decision making. IEEE Circuits and Systems Magazine **6**(3,21–45) (2006)

5. Džeroski, S., Ženko, B.: Is combining classifiers with stacking better than selecting the best one? Machine Learning **54**(3) (Mar 2004) 255–273
6. Vannoorenberghe, P.: On aggregating belief decision trees. Information Fusion **5**(3, 179–188) (2004)
7. Xu, P., Davoine, F., Zha, H., Denœux, T.: Evidential calibration of binary svm classifiers. International Journal of Approximate Reasoning, 72, 55–70 (2016)
8. Ma, L., Sun, B., Li, Z.: Bagging likelihood-based belief decision trees. In: 2017 20th International Conference on Information Fusion (Fusion), 1–6. (2017)
9. Dempster, A.P.: Upper and lower probabilities induced by a multivalued mapping. Annals of Mathematical Statistics, 38, 325–339 (1967)
10. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
11. Denoeux, T., El Zoghby, N., Cherfaoui, V., Jouglet, A.: Optimal object association in the Dempster-Shafer framework. IEEE Transactions on Cybernetics **44**(22, 2521-2531) (2014)
12. Efron, B.: Bootstrap methods: Another look at the jackknife. Ann. Statist. (1979)
13. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci., 119–139 (1997)
14. Schapire, R.E. In: Explaining AdaBoost. Springer Berlin Heidelberg, Berlin, Heidelberg, 37–52 (2013)
15. Wolpert, D.H.: Stacked generalization. Neural Networks, 5(2), 241–259 (1992)
16. Breiman, L.: Random Forests. Machine Learning, 45, 5–32 (2001)
17. Cortes, C., Mohri, M., Riley, M., Rostamizadeh, A. In: Sample Selection Bias Correction Theory. Springer Berlin Heidelberg, Berlin, Heidelberg, 38–53 (2008)
18. Duin, R.P.W.: The combining classifier: to train or not to train? Object recognition supported by user interaction for service robots, 2, 765–770 (2002)
19. Smets, P., Kennes, R.: The transferable belief model. Artif. Intell., 66(2), 191–234 (1994)
20. Yager, R.R.: On the dempster-shafer framework and new combination rules. Information sciences, 41(2), 93–137 (1987)
21. Smets, P.: Belief functions: the disjunctive rule of combination and the generalized bayesian. International Journal of Approximate Reasoning, 9(1), 1–32 (2005)
22. Dubois, D., Prade, H.: Representation and combination of uncertainty with belief functions and possibility measures. Computational intelligence, 4(3), 244–264 (1988)
23. Florea, M.C., Dezert, J., Valin, P., Smarandache, F., Jousselme, A.: Adaptative combination rule and proportional conflict redistribution rule for information fusion. CoRR **abs/cs/0604042** (2006)
24. François, J., Grandvalet, Y., Denceux, T., Roger, J.M. In: Bagging Improves Uncertainty Representation in Evidential Pattern Classification. Physica-Verlag HD, 295–308 (2002)
25. Xu, P., Davoine, F., Denoeux, T.: Evidential combination of pedestrian detectors. In: British Machine Vision Conference, Nottingham, 1–14 (2014)
26. Denœux, T.: Maximum likelihood estimation from uncertain data in the belief function framework. IEEE Trans. on Know. and Data Eng., 25, 119–130 (2011)
27. Sutton-Charani, N., Destercke, S., Denœux, T.: Learning decision trees from uncertain data with an evidential em approach. International Conference on Machine Learning and Applications (ICMLA) (2013)