Four Measures of Data Complexity for Bootstrapping, Splitting and Feature Sampling

Catherine A. Shipp and Ludmila I. Kuncheva
School of Informatics
University of Wales, Bangor
Bangor, Gwynedd, LL57 1UT, United Kingdom
{c.a.shipp,l.i.kuncheva}@bangor.ac.uk

Abstract

Multiple classifier systems achieve best results when the individual classifier outputs are diverse and of similar accuracy. If there is a substantial difference in the individual accuracies, then the ensemble might be worse than the single best or even worse than an average member of the team. Various methods have been proposed for creating subsets of the given labeled data set, subsequently used to train the individual classifiers. Three such methods are: taking bootstrap samples, splitting the data set into disjoint subsets, and random feature sampling. Following a recent publication by Ho [4], here we offer a small experimental study on complexity of the subsets obtained by the three methods. Four measures of complexity have been used – the three from [4]: the minimal spanning tree (MST), the adherence subsets measure (ADH), the maximal feature efficiency (MFE); and a cluster label consistency measure (CLC) proposed here. We used the UCI "wine" dataset (3 classes, 13 features) with 7 cases of class split: 1v2v3, 1v(2&3), 2v(1&3), and 3v(1&2), and the 3 pairwise splits. Surprisingly, the values of the four measures were substantially different from each other, although the "wine" data set is perceived as *easy*. Of the three methods for selecting data subsets, the feature sampling had the highest variability of the data complexity.

1 Introduction

Multiple classifier systems are deemed to be more accurate than the best individual classifier in the ensemble. However, this is not necessarily true for any classifier ensemble. For example, the majority vote across dependent classifiers of the same accuracy may be much worse or much better than the individual accuracy [6]. Diversity in the team is not the only factor that affects the system performance. If the individual classifiers have substantially different accuracies, it is unlikely that the combination will improve on the best individual. It seems that it is important to have similarly accurate classifiers (even weak ones [5]) but with high diversity.

Unfortunately, independent training of classifiers do not guarantee independence (in statistical sense) of their outputs [7]. One approach to enhancing diversity of the individual classifiers is to train them on different subsets of the available labeled data set. Let $Z = \{\mathbf{z}_1, \ldots, \mathbf{z}_N\}$ be a labeled data set, $\mathbf{z}_j \in \Re^n$, $j = 1, \ldots, N$ with N elements. Three methods for obtaining different data subsets have been used in recent studies

1. **Bootstrap sampling**. We sample from Z with repetition assuming uniform probability across the elements of Z. Typically the cardinality of the bootstrap sample is chosen to be N. Bootstrap sampling underlies the boosting method for designing classifier ensembles [1].

2. **Data splitting**. The individual classifiers can be built on disjoint subsets of Z, as in crossvalidation estimation of classification accuracy.

3. **Random feature sampling**. We can base the individual classifiers on different subsets of features, i.e., on different subspaces of the feature space $\Re^n$. Ho [3] shows that random sampling (without repetition) to get a set of $d < n$ features from the integers from 1 to n, is a viable line for building multiple classifier systems.

In this study we are interested in measuring the *complexity* of the data subsets obtained through the three methods. We apply four measures of complexity, the minimal spanning tree (MST),

the adherence subsethood (ADH) based on the ϵ-neighborhood measure as proposed in [4], the maximum feature efficiency (MFE), all three from [4], and a measure which we call the *Cluster Label Consistency (CLC)*. The complexity measures are explained in Section 2. In Section 3 we detail our experimental results with the "wine" data set from the UCI Machine Learning Repository database. Section 4 contains our analysis and conclusions.

2 Measures of complexity

2.1 Minimal Spanning Tree

This method for measuring the class boundary length, is used by Ho [4], having originally been proposed in a paper by Friedman and Rafsky [2]. Given a metric, a minimal spanning tree can be constructed which connects all the sample points regardless of their class labels. We use the Euclidean distance as the metric. Clearly some edges of the MST will connect points from different classes and the count of such edges gives us a measure of the length of the boundary between the classes. Since there are $N-1$ edges for N sample points, the count can be normalised as a percentage of N [4]. The edges connecting two different classes are shown with bold lines in the examples in Figures 1 and 2. Ho [4] only considers the two class case but the method can be extended to multiple classes (see Figure 1 a),b) and Figure 2 a),b)).

The MST method

1. Take a labelled data set of size N and construct the minimal spanning tree disregarding the class labels.
2. Count the number of edges in the minimal spanning tree connecting different classes, N_e.
3. The complexity measure is given by

$$MSTcomplexity = \frac{N_e}{N} \qquad (1)$$

2.2 Adherence Subsets

This method proposed by Ho [4] considers the clustering properties of the data. It is based on a reflexive, symmetric binary relation $\mathcal{R}$ between two points x, y in a set F. $\mathcal{R}$ is defined by $x\mathcal{R}y \Leftrightarrow d(x,y) < \epsilon$, where $d(x,y)$ is a given metric and ϵ is a given non-zero constant. We define $\Gamma(x) = \{y \in F | y\mathcal{R}x\}$ to be the ϵ-neighbourhood of x. For our study we have again taken the Euclidean distance as our metric. An adherence mapping, ad from the power set $\mathcal{P}(F)$ to $\mathcal{P}(F)$ is such that:

$$\begin{cases} ad(\phi) &= \phi \\ ad(x) &= \Gamma(x) \\ ad(A) &= \bigcup_{x \in A} ad(x) \quad \forall A \subseteq F. \end{cases}$$

The largest possible adherence subsets can be grown for each point by successively expanding the adherence subset at each stage whilst ensuring that all newly included points come from the same class. For example, $ad^0(\{x\}) = \{x\}$, $ad^1(\{x\}) = ad(\{x\})$, $ad^2(\{x\}) = ad(ad(\{x\}))\ldots$, gives us progressively higher order adherence subsets. For each point, only the highest order subset is retained such that all elements are from the same class. This procedure defines a partition of the data set where each cluster contains data points with the same class label. The number of such clusters is an indication of the complexity of the problem. If the classes are compact and far from each other, then each class will ideally form a single separate adherence subset. When the classes are overlapping, multiple clusters are likely to appear. Again, this method can be used not only on two class cases but also on multiple class cases (see Figure 1 a),c) and Figure 2 a),c)).

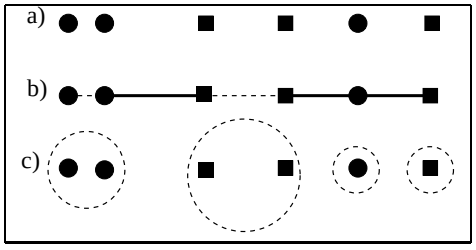

Figure 1: A 2 class example. a) distribution of the classes, b) a minimal spanning tree $MSTcomplexity = \frac{3}{6}$, c) the adherence subsets $ADHcomplexity = \frac{4}{6}$.

The ADH method

1. Take a labelled data set of size N and for each point grow the largest possible adherence subset such that all elements of the subset are from the same class.
2. Count the number of *different* adherence subsets N_s.
3. The complexity is then given by:

$$ADHcomplexity = \frac{N_s}{N} \qquad (2)$$

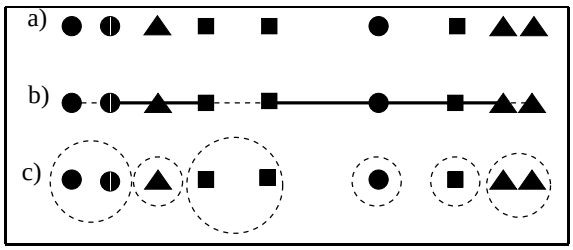

Figure 2: A 3 class example. a) distribution of the classes, b) A minimal spanning tree $MSTcomplexity = \frac{5}{9}$, c) the adherence subsets $ADHcomplexity = \frac{6}{9}$.

In Ho's paper [4] the choice of ϵ was $\epsilon = 0.55\delta$ where δ was the minimal distance between two points of different classes. In a preliminary experiment we studied the effect of ϵ on the complexity value. We used $\epsilon = min + k * (max - min)$ for various values of k, where the min and max were the minimum and maximum distances in the data set regardless of class labels. Table 1 shows the result with a two-class split of the data set used in our further experiments.

Table 1: Complexity values for varying values of ϵ

k	$ADHcomplexity$
0	1.0
0.1	0.7854
0.2	0.7528
0.3	0.7753
0.4	0.7528
0.5	1.0

We can see from the table that the relationship between ϵ and $ADHcomplexity$ is not monotonic. Indeed, if ϵ is too small, then each point will be a cluster on its own, and $ADHcomplexity = 1$. On the other hand, if ϵ is too large, then the ϵ-neighbourhood of $\mathbf{x}$ will contain point(s) from a different class. Again, $\mathbf{x}$ will be marked as a cluster on its own, leading to $ADHcomplexity = 1$. Since there seems to be no clear reason for choosing a particular ϵ, we picked $k = 0.1$.

2.3 Maximum Feature Efficiency

This method is suitable for 2 classes only. The complexity on each feature is assessed separately. All points are projected on that feature axis and the overlap interval is found. The the fraction of the points within this interval defines the complexity on that feature axis. For the example in Figure 1 c.), the overlap interval contains 3 of the 6 points, so the complexity on this feature is $MFEcomplexity = \frac{3}{6}$ (the efficiency is $1 - MFEcomplexity$).

The MFE method

1. Take a labelled data set (2 classes) of size N.
2. For each feature (i) project the data on its axis and identify the overlap region. Count the number of points in that region, N_i and calculate the MFE complexity as

$$MFEcomplexity_i = \frac{N_i}{N}. \quad (3)$$

3. Take as the final complexity value

$$MFEcomplexity = \min_i MFEcomplexity_i. \quad (4)$$

2.4 Cluster Label Consistency

This measure estimates how well the classes match the possible clusters in data. First c clusters are obtained on the whole data set regardless of the class labels and then the labels are used to count the number from each class within each cluster. "Pure" clusters will give low complexity values whereas "contaminated" clusters will give high complexity values.

The CLC method

1. Take a labelled data set of size N and cluster it into c clusters disregarding the class labels.
2. For each cluster (i) calculate the *cluster label consistency* C_i, as the fraction of the maximal number of points of the same class label n the cluster from the total number of points in that cluster.
3. Take as the final complexity value

$$CLCcomplexity = 1 - \frac{1}{c}\sum_{i=1}^{c} C_i. \quad (5)$$

In case of a perfect match, i.e., when each class is a cluster on its own, the complexity is 0.

2.5 Limits of Complexity

With c classes and N elements, for the MST method we have $N-1$ edges. The maximum possible number of edges connecting different classes is therefore $N-1$, and the maximum complexity for the MST method is $\frac{N-1}{N}$. As we have c classes the minimum possible number of edges is $c-1$ and the minimum complexity is $\frac{c-1}{N}$.

For the ADH method, as we have N elements and each of these can be in its own adherence subset, the maximum complexity is $\frac{N}{N}=1$. The minimum number of adherence subsets is one per class and the minimum complexity is therefore $\frac{c}{N}$.

The MFE method will give the lowest complexity of 0 when the two classes are completely separable on one or more feature axes (the overlap interval will contain 0 points). The highest possible value is 1, and it is reached when on all feature axes the lowest and the highest points correspond to more than one data point, and these have different class labels (then each overlap interval contains all N points).

The CLC method gives 0, the lowest value, when the classes correspond exactly to the clusters found. Similarly to the ADH method, the number of clusters, c, has to be picked in advance. We can take c to be the number of classes as a reasonable choice. It is easy to verify that in this case, the upper limit of the $CLCcomplexity$ is $1-\frac{1}{c^2}$.

3 Experiments

3.1 Data

The data set we used in this study was from the UCI Repository of Machine Learning Database[1]. It is called *wine* and contains 178 cases, with 13 features and 3 classes. All the feature values are continuous numerical values and there are no missing values. From this data set we derived four problems, the three-class case and three two-class cases:-

Case A: 1 v 2 v 3.
Case B: 1 v (2 and 3).
Case C: 2 v (1 and 3).
Case D: 3 v (1 and 2).
Case E: 1 v 2.
Case F: 1 v 3.
Case G: 2 v 3.

The data set can be stored as an array of size $N \times n$, consisting of $N=178$ elements with $n=13$ features. A separate array of size $N \times 1$ contains the class labels corresponding to each element.

To find out how "complex" the data set is, we ran the linear and quadratic discriminant classifiers (LDC and QDC), the nearest neighbor (1-nn), and the 5-nearest neighbor (5-nn). The test results from a 10-fold cross-validation and the leave-one-out are shown in Table 2.

Table 2: Classification accuracy (in %) with the "wine" data.

	LDC	QDC	1-nn	5-nn
10-fold cross-validation	99	99	79	70
leave-one-out	99	99	77	70

The results show that while the data set might be easy for LDC and QDC, it is not too easy for the 1-nn and 5-nn. So if there is single meaning of *complexity*, what should the complexity be for the "wine" data set? This hints that perhaps different components of complexity should be sought.

3.2 Methods

For each of the four cases we took 50 bootstrap samples of size N; for the data splitting method we used 100 random splits into halves; and for the random feature sampling, we formed 50 subsets by randomly choosing 5 of the $n=13$ features. The four complexity measures were calculated for each data set. The means, maxima and minima were found separately for the three data selection methods and each of the seven cases.

The varying number of repetitions reflects the variation in the time involved in running each method. Since the Data Splitting method is only working on half the data, it is considerably faster than the other two methods. Hence this method was repeated 100 times while the others were repeated 50 times.

3.3 Results

Before considering the values of complexity calculated using the different data manipulation methods. We calculated the complexity for the whole of the data set for the seven cases. The results we obtained are shown in Table 3.

These results show that the $ADHcomplexity$ considers each problem to be of the same high degree of complexity, whereas the other three give dif-

[1] [http://www.ics.uci.edu/ mlearn/MLRepository.html]

ferent values. The four measures do not agree on a single case being most complex or easiest.

Table 3: Complexity calculated by MST, ADH, MFE and CLC (in %) for the data set as a whole

Case	MST	ADH	MFE	CLC
A	25	100	N/A	28
B	8	100	24	9
C	21	100	42	26
D	21	100	28	21
E	8	100	21	8
F	9	100	0	11
G	27	100	25	34

Table 4 shows the means and the standard deviations for the 3 data selection methods, the 4 measures and the 7 cases. For the data splitting method, the results for the two halves are averaged.

The first interesting observation from Table 4 is that the measures give very different values. Knowing that the three of them (except CLC) span approximately the same intervals ($0.01 \leq MSTcomplexity \leq 0.99$, $0.02 \leq ADHcomplexity \leq 1$, and $0 \leq MFEcomplexity \leq 1$), the differences in the complexity values are puzzling.

Figures 3 and 6 plot the means for the 21 experiments (3 methods $\times$ 7 cases) and the minima and maxima as the error bars.

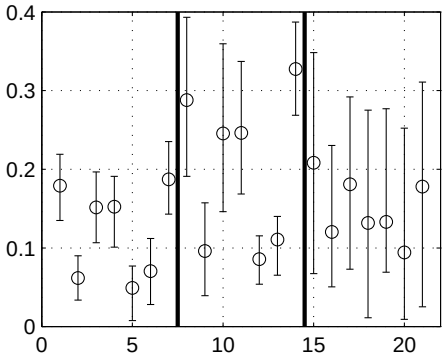

Figure 3: The mean and limits for $MSTcomplexity$. Bars 1-7: Bootstrapping, 8-14: Data Splitting, 15-21: Feature Sampling

In all 4 figures the first 7 (six for MFE) bars are for the Bootstrapping method, the next 7 (6) are for the Data splitting method, and the last 7 (6) correspond with the Feature Sampling method in

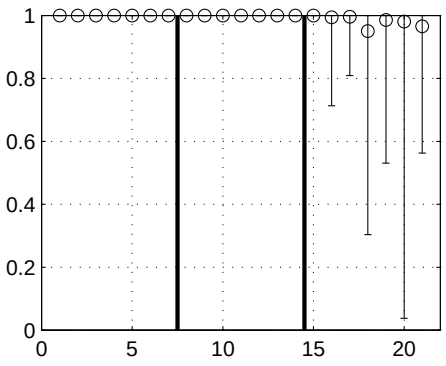

Figure 4: The mean and limits for $ADHcomplexity$. Bars 1-7: Bootstrapping, 8-14: Data Splitting, 15-21: Feature Sampling

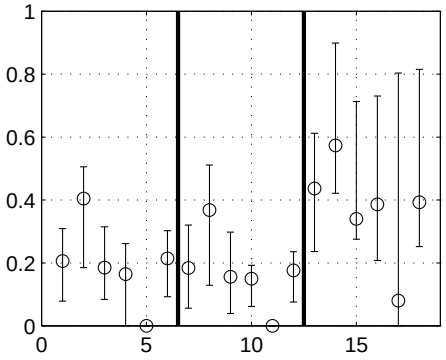

Figure 5: The mean and limits for $MFEcomplexity$. Bars 1-6: Bootstrapping, 7-12: Data Splitting, 13-18: Feature Sampling

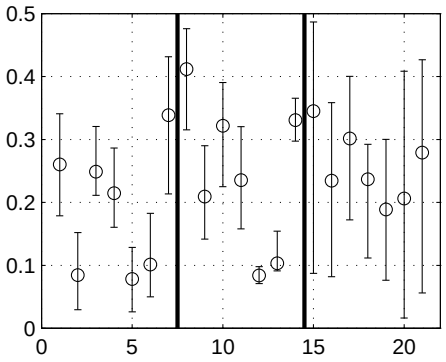

Figure 6: The mean and limits for $CLCcomplexity$. Bars 1-7: Bootstrapping, 8-14: Data Splitting, 15-21: Feature Sampling

Table 4: Complexity calculated by MST, ADH, MFE and CLC (in %) with the three methods for the 7 cases

Method	Case	MST		ADH		MFE		CLC	
		mean	std	mean	std	mean	std	mean	std
Bootstrapping (50 experiments)	A	18	2	100	0	–	–	26	3
	B	6	1	100	0	21	5	8	3
	C	15	2	100	0	40	7	25	2
	D	15	2	100	0	19	7	21	3
	E	5	2	100	0	16	5	8	2
	F	7	2	100	0	0	0	10	3
	G	19	2	100	0	21	5	34	4
Data splitting (100 experiments)	A	29	4	100	0	–	–	41	3
	B	10	2	100	0	18	5	21	3
	C	25	4	100	0	37	9	32	3
	D	25	4	100	0	16	6	24	3
	E	9	1	100	0	15	3	8	1
	F	11	2	100	0	0	0	10	1
	G	33	3	100	0	18	3	33	2
Feature sampling (50 experiments)	A	21	7	100	0	–	–	35	11
	B	12	5	99	4	44	14	23	11
	C	18	5	100	3	57	15	30	6
	D	13	8	95	17	34	12	24	4
	E	13	5	99	7	39	17	19	10
	F	9	5	98	14	8	17	21	14
	G	18	10	97	10	39	18	28	12

the same order of the cases (A to G).

A common finding of all complexity measures is that the feature sampling method for creating data subsets offers the highest variability of the complexity of the obtained sets. However, this seems to be the only finding where the four complexity measures agree. For example, while the ADH measure designates the feature sampling method as producing the least complex data (Figure 4), the MFE measures classes these data set as the hardest (Figure 5).

The Bootstrap method has the lowest standard deviations (on all four measures) indicating that the data sets obtained exhibit complexity of a similar value. Among other (theoretical) advantages, this makes the bootstrapping method for obtaining data sets a preferable candidate in designing classifier combination systems.

We studied further the relationship of the four complexity measures. We took the complexity values of the 18 experiments for the two-class cases only (from B to G), so that MFE can also participate in the comparison. The 18 means for each measure were ranked and the Spearman's rank correlation coefficient was calculated for each pair of measures (Table 5). Since there is no consensus on a single definition of complexity, we cannot choose one "right" measure. The positive correlation between MST and CLC is an indication that the aspects of complexity assessed by these two measures are similar (even though the measures might not be similar by value). Indeed, the MST method is based on nearest neighbor *clustering* and CLC uses clustering as well.

Table 5: Spearman's rank correlation coefficients between the 4 complexity measures

	MST	ADH	MFE
ADH	0.0426		
MFE	0.4258	-0.2955	
CLC	0.8913	-0.0815	0.5174

4 Conclusions

In this study we consider three methods for varying the data set for building ensembles of classifiers:

taking bootstrap samples, splitting the data set into disjoint subsets, and random feature sampling. We have carried out a small experiment on complexity of the subsets obtained by the three methods.

We described 4 measures of complexity the minimal spanning tree (MST), the adherence subsets measure (ADH), the maximal feature efficiency (MFE); and a cluster label consistency measure (CLC) proposed here. Our results with the UCI "wine" dataset led us to the following two conclusions

1. Of the three methods for selecting data subsets, the feature sampling had the highest variability of the data complexity on all four measures.

2. The pairwise relationships between the complexity measures suggest that the measures assess different aspects of complexity. Therefore it might be beneficial to define those complexity aspects in a more formal way and regard the measures as a group. Thus, a complexity vector can be used to guide us to an appropriate classifier model for a certain data set or indicate whether a collection of data sets is a suitable basis for a multiple classifier system.

Of course, our conclusions at this stage are only tentative, and more experiments are needed to clarify the behaviour of the complexity measures.

References

[1] L. Brieman. Combining predictors. In A.J.C. Sharkey, editor, *Combining Artificial Neural Nets*, pages 31–50. Springer-Verlag, London, 1999.

[2] J.H. Friedman and L.C. Rafsky, Multivariate generalisations of the Wald-Wolfowitz and Smirnov two-sample tests *Annals of Statistics*, **7**, 4, 1979, 697-717 In [4]

[3] T.K. Ho. The random space method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.

[4] T.K. Ho. Complexity of classification problems and comparative advantages of combined classifiers. In *Proc. First International Workshop on Multiple Classifier Systems*, pages 97–106, Cagliari, Italy, 2000.

[5] C. Ji and S. Ma. Combination of weak classifiers. *IEEE Transactions on Neural Networks*, 8(1):32–42, 1997.

[6] L.I. Kuncheva, C.J. Whitaker, C.A. Shipp, and R.P.W. Duin. Limits on the majority vote accuracy in classifier fusion. (submitted).

[7] B. Littlewood and D.R. Miller. Conceptual modeling of coincident failures in multiversion software. *IEEE Transactions on Software Engineering*, 15(12):1596–1614, 1989.