Finding the Number of Fuzzy Clusters by Resampling

Christian Borgelt and Rudolf Kruse

Dept. of Knowledge Processing and Language Engineering
Otto-von-Guericke-University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany
{borgelt,kruse}@iws.cs.uni-magdeburg.de

***Abstract*— Recently several papers studied resampling approaches to determine the number of clusters in prototype-based clustering. The core idea underlying these approaches is that with the right choice for the number of clusters basically the same cluster structures should be obtained from subsamples of the given data set, while a wrong choice should produce considerably varying cluster structures. In this paper we investigate whether these approaches can be transferred to fuzzy clustering. It turns out that they are applicable to fuzzy clustering as well, but that not all relative cluster evaluation measures that work for crisp clustering can also be used for fuzzy clustering.**

I. Introduction

A core problem of prototype-based clustering algorithms—like the classical c-means algorithm [1], [18], [23], its fuzzy counterpart (fuzzy c-means) [2], [4], [19], or the expectation maximization algorithm for estimating a mixture of Gaussians [9], [12], [5]—is that they require the number of clusters to be known in advance. This is, of course, inconvenient in practice, since in applications we rarely find ourselves in such a favorable position. Rather we would like to have a method to determine the number of clusters from the data set.

A common approach to tackle this problem is to cluster the given data set several times, each time with a different number of clusters from a user-specified range. The clustering results are evaluated and then the number of clusters yielding the best evaluation is chosen. In some cases the selection criterion may also be a (reasonably clear) local optimum for a certain number of clusters, or a clear change in the behavior of the evaluation over the number of clusters (for example, a knee or a maximum or minimum in the first or second derivative). In fuzzy clustering, this approach is very common in connection with so-called *internal cluster evaluation measures*, like, for example, the partition entropy [2], [27], the Fukuyama-Sugeno index [14], or the Xie-Beni index [28] (overviews can be found in [4], [19], [16], [17], [7]). However, most of these measures are pretty unreliable and often yield inconclusive results even if the cluster structure is actually fairly clear.

In this paper we study an alternative approach that has recently attracted a lot of attention in crisp and probabilistic clustering. The core idea of this approach is that if we cluster subsamples of the given data set with the "right" number of clusters, we should end up with basically the same cluster structure in each run. With a "wrong" number of clusters, however, the clustering result should be unstable, showing considerable variation between different subsamples. Thus, by measuring the stability of the clustering result w.r.t. subsampling (similarity of results from different runs), one may be able to determine the "best" number of clusters: it is the one for which the clustering results are most stable.

Intuitively, one may think of this as follows: if the "true" number of clusters is c and we try to find $c + 1$ clusters, one cluster has to be split. If we try to find $c - 1$ clusters, some pair of clusters has to be merged. As it depends on particular properties of the subsample which cluster is split or which clusters are merged, we should get somewhat differing structures in each run. By measuring how well the clustering results coincide, we can thus discover such situations and choose the number of clusters based on this information.

This paper is organized as follows: in Section II we review the basics of relative cluster evaluation measures for crisp clustering, and transfer them, in a straightforward way, to fuzzy clustering. In Section III we review two basic schemes for resampling, and corresponding cluster evaluation schemes, which have been suggested. In Section IV we report experimental results we obtained for hard and fuzzy clustering. They reveal that the approach is feasible, but that for fuzzy clustering one has to choose the evaluation measure with care.

II. Relative Cluster Evaluation Measures

Relative cluster evaluation measures compare two partitions of given data, one being a clustering result and the other either also a clustering result or given by a classification or a user-defined grouping. In the latter case one also speaks of *external cluster evaluation measures* [16], [17], although the methods used are usually the same. Two clustering results, however, may also be compared based on the cluster parameters alone, although we do not discuss such methods here.

We assume that a partition of the given data set is described by a $c \times n$ partition matrix $\mathbf{U} = (u_{ij})_{1 \leq i \leq c, 1 \leq j \leq n}$, where c is the number of clusters and n the number of data points. An element u_{ij} of such a matrix states, in the crisp case, whether the j-th data point belongs to the i-th cluster ($u_{ij} = 1$) or not ($u_{ij} = 0$). In the fuzzy case, u_{ij} is the degree of membership to which the j-th data point belongs to the i-th cluster (usually satisfying the constraint $\forall j; 1 \leq j \leq n : \sum_{i=1}^{c} u_{ij} = 1$).

We also assume that the two partition matrices we have to compare have the same dimensions, that is, refer to the same numbers of clusters and data points. However, it is also imaginable to compare matrices with different numbers of rows, although some measures may give misleading results in this case, since they are based on the assumption that it is possible to set up a bijective mapping between the cluster.

Regardless of whether the numbers of rows coincide or not, we face the problem of relating the clusters of the one partition to the clusters of the other partition. There are basically three solutions to this problem: (1) for each cluster in the one partition we determine the *best fitting* cluster in the other, (2) we find the *best permutation* of the rows of one partition matrix, that is, the best one-to-one mapping of the clusters, or (3) we compare the partition matrices indirectly by first setting up a *coincidence matrix* for each of them, which records for each pair of data points whether they are assigned to the same cluster or not, and then compare the coincidence matrices.

The first alternative has the advantage of being fairly efficient (time complexity $O(nc^2)$, but the severe disadvantage that we cannot make sure that we obtain a one-to-one relationship. Some clusters in the second partition may not be paired with any cluster in the first, which also renders the approach asymmetric. The second alternative has the advantage that it definitely finds the best one-to-one relationship. Its disadvantage is the slightly higher computational cost (time complexity $O(nc^2 + c^3)$, see below). The third alternative has the disadvantages that it does not yield a direct indication of how to relate the clusters to each other and that it can have fairly high computational costs (time complexity $O(n^2c)$), especially for a large number of data points. However, the fact that it does not need an explicit mapping between the clusters can also be seen as an advantage, because it renders this method very flexible. In particular, this method is well suited to compare partitions with different numbers of clusters.

A. Comparing Partition Matrices

The first two approaches outlined above directly compare two $c \times n$ partition matrices $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$. For both of them we need a measure that compares two rows, one from each matrix. Such measures can be derived from measures comparing binary classifications, like, for example, the *accuracy* or the F_1*-measure* [25]. Formally, we set up a 2×2 contingency table for each pair of rows, one from each matrix (cf. Table I). That is, for each pair $(i, k) \in \{1, \ldots, c\}^2$ we compute

$$
\begin{aligned}
n_{11}^{(i,k)}\big(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}\big) &= \sum_{j=1}^{n} u_{ij}^{(1)} \cdot u_{kj}^{(2)}, \\
n_{01}^{(i,k)}\big(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}\big) &= \sum_{j=1}^{n} \left(1 - u_{ij}^{(1)}\right) \cdot u_{kj}^{(2)}, \\
n_{10}^{(i,k)}\big(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}\big) &= \sum_{j=1}^{n} u_{ij}^{(1)} \cdot \left(1 - u_{kj}^{(2)}\right), \\
n_{00}^{(i,k)}\big(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}\big) &= \sum_{j=1}^{n} \left(1 - u_{ij}^{(1)}\right) \cdot \left(1 - u_{kj}^{(2)}\right).
\end{aligned}
$$

TABLE I

CONTINGENCY TABLE COMPARING ROWS OF TWO PARTITION MATRICES

	$u_{kj}^{(2)} = 1$	$u_{kj}^{(2)} = 0$	Σ
$u_{ij}^{(1)} = 1$	$n_{11}^{(i,k)}$	$n_{10}^{(i,k)}$	$n_{1.}^{(i,k)}$
$u_{ij}^{(1)} = 0$	$n_{01}^{(i,k)}$	$n_{00}^{(i,k)}$	$n_{0.}^{(i,k)}$
Σ	$n_{.1}^{(i,k)}$	$n_{.0}^{(i,k)}$	n

(In the following we generally drop the arguments $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ to make the formulae easier to read.) All of these numbers may also be computed from fuzzy or probabilistic membership degrees. Actually they have a fairly natural interpretation in fuzzy clustering. This can be seen as follows: in the crisp case, n_{11} is the number of data points that are assigned to the i-th cluster of the first partition *and* to the k-th cluster of the second partition, where the *and* is formally expressed by a product. Allowing membership degrees from $[0, 1]$ and drawing on the theory of fuzzy logic, we see that this is only a special case of a t-norm that combines the two statements. Hence, in the general case, we may replace the product by an arbitrary t-norm. Analogously, the expressions $1 - u_{ij}$ can be seen as resulting from an application of the standard fuzzy negation, and indeed: they refer to negated statements "The j-th data point does *not* belong to the i-th cluster." In this way we achieve a straightforward generalization of all following measures to fuzzy clustering results, even though we confine ourselves to the above formulae for this paper (which use the product to express a conjunction).

From the above numbers we may compute any measure that can be used to evaluate a binary classification, maximizing the result over all permutations, each of which provides a column mapping.[1] An example is the (averaged) F_1 *measure* [25]

$$
F_1\big(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}\big) = \max_{\varsigma \in \Pi(c)} \frac{1}{c} \sum_{i=1}^{c} \frac{2\,\pi_{i,\varsigma(i)} \rho_{i,\varsigma(i)}}{\pi_{i,\varsigma(i)} + \rho_{i,\varsigma(i)}},
$$

where $\Pi(c)$ is the set of all permutations of the c numbers $1, \ldots, c$ and cluster-specific precision and recall are

$$
\begin{aligned}
\pi_{i,k} &= \frac{n_{11}^{(i,k)}}{n_{01}^{(i,k)} + n_{11}^{(i,k)}} \qquad \text{and} \\
\rho_{i,k} &= \frac{n_{11}^{(i,k)}}{n_{10}^{(i,k)} + n_{11}^{(i,k)}}.
\end{aligned}
$$

Another example is the *(cross-classification) accuracy*, averaged over all columns, that is,

$$
Q_{\mathrm{acc}}\big(\mathbf{U}^{(1)}, \mathbf{U}^{(2)}\big) = \max_{\varsigma \in \Pi(c)} \frac{1}{cn} \sum_{i=1}^{c} \left(n_{00}^{(i,\varsigma(i))} + n_{11}^{(i,\varsigma(i))}\right).
$$

Two partition matrices $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$ are the more similar, the higher the values of the (averaged) F_1 measure or the (cross-classification) accuracy.

[1] Note that with the so-called *Hungarian method* for solving optimum weighted bipartite matching problems [24] the time complexity of finding the maximum over all permutations for given pairwise column comparison values is only $O(c^3)$ and not $O(c!)$.

TABLE II
CONTINGENCY TABLE FOR COMPARING COINCIDENCE MATRICES

	$\psi_{jl}^{(2)}=1$	$\psi_{jl}^{(2)}=0$	Σ
$\psi_{jl}^{(1)}=1$	N_{SS}	N_{SD}	$N_{S.}$
$\psi_{jl}^{(1)}=0$	N_{DS}	N_{DD}	$N_{D.}$
Σ	$N_{.S}$	$N_{.D}$	$N_{..}$

An alternative to these classification-based measures is a simple mean squared difference comparison of the partition matrices (which, at least to the authors' knowledge, has not been used before). That is, we compute

$$Q_{\text{diff}}\big(\mathbf{U}^{(1)},\mathbf{U}^{(2)}\big)=\min_{\varsigma\in\Pi(c)}\frac{1}{cn}\sum_{i=1}^{c}\sum_{j=1}^{n}\Big(u_{ij}^{(1)}-u_{\varsigma(i)j}^{(2)}\Big)^2.$$

The smaller this measure, the more similar are the partitions. Note that for crisp clustering (that is, for $u_{ij}\in\{0,1\}$) this measure may also be written as

$$Q_{\text{diff}}\big(\mathbf{U}^{(1)},\mathbf{U}^{(2)}\big)=\min_{\varsigma\in\Pi(c)}\frac{1}{cn}\sum_{i=1}^{c}\Big(n_{01}^{(i,k)}+n_{10}^{(i,k)}\Big).$$

Hence it is closely related to the (cross-classification) accuracy, since it is obviously $n_{01}^{(i,k)}+n_{10}^{(i,k)}=n-(n_{00}^{(i,k)}+n_{11}^{(i,k)})$ for crisp partitions. This measure is actually the most natural for fuzzy clustering and thus it is not surprising that, as we will see in Section IV, it performs best for fuzzy clustering.

B. Comparing Coincidence Matrices

As an alternative to comparing the partition matrices directly, one may first compute from each of them an $n\times n$ *coincidence matrix*, also called a *cluster connectivity matrix* [22], which states for each pair of data points whether they are assigned to the same cluster or not. Formally, a coincidence matrix $\mathbf{\Psi}=(\psi_{jl})_{1\le j,l\le n}$ can be computed from a partition matrix $\mathbf{U}=(u_{ij})_{1\le i\le c,1\le j\le n}$ by

$$\psi_{jl}=\sum_{i=1}^{c}u_{ij}u_{il}.$$

Note again that these values may also be computed from fuzzy or probabilistic membership degrees, possibly replacing the product (which represents a conjunction) by another t-norm.

After coincidence matrices $\mathbf{\Psi}^{(1)}$ and $\mathbf{\Psi}^{(2)}$ are computed from the two partition matrices $\mathbf{U}^{(1)}$ and $\mathbf{U}^{(2)}$, the comparison is carried out by computing statistics of the number of data point pairs that are in the same group in both partitions, in the same group in one, but in different groups in the other, or in different groups in both. The main advantage of this approach is, of course, that we are freed of the need to pair the groups of the two partitions. We rather exploit that data points that are considered (dis)similar by one partition should also be considered (dis)similar by the other.

Formally, we compute a 2×2 contingency table (cf. Table II) containing the numbers (which are basically counts of the different pairs $\big(\psi_{jl}^{(1)},\psi_{jl}^{(2)}\big)$)

$$\begin{aligned}
N_{SS}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big) &= \sum_{j=2}^{n}\sum_{l=1}^{j-1}\psi_{jl}^{(1)}\psi_{jl}^{(2)},\\
N_{SD}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big) &= \sum_{j=2}^{n}\sum_{l=1}^{j-1}\psi_{jl}^{(1)}\Big(1-\psi_{jl}^{(2)}\Big),\\
N_{DS}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big) &= \sum_{j=2}^{n}\sum_{l=1}^{j-1}\Big(1-\psi_{jl}^{(1)}\Big)\psi_{jl}^{(2)},\\
N_{DD}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big) &= \sum_{j=2}^{n}\sum_{l=1}^{j-1}\Big(1-\psi_{jl}^{(1)}\Big)\Big(1-\psi_{jl}^{(2)}\Big),
\end{aligned}$$

where the index S stands for "same group" and the index D stands for "different groups" and the two indices refer to the two partitions. Again the product may be replaced by any t-norm. (To make the formulae easier to read, the arguments $\mathbf{\Psi}^{(1)}$ and $\mathbf{\Psi}^{(2)}$ are dropped in the following.) From these number a large variety of measures may be computed. Well-known examples include the *Rand statistic*

$$Q_{\text{Rand}}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big)=\frac{N_{SS}+N_{DD}}{N_{..}},$$

which is a simple ratio of the number of data point pairs treated the same in both partitions to all data point pairs, and the *Jaccard coefficient*

$$Q_{\text{Jaccard}}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big)=\frac{N_{SS}}{N_{SS}+N_{SD}+N_{DS}},$$

which ignores negative information, that is, pairs that are assigned to different groups in both partitions. Both measures are to be maximized. Another frequently encountered measure is the *Folkes–Mallows index*

$$Q_{\text{FM}}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big)=\frac{N_{SS}}{\sqrt{(N_{SS}+N_{SD})(N_{SS}+N_{DS})}},$$

which can be interpreted as a cosine similarity measure, because it computes the cosine between two binary vectors, each of which contains all elements of one of the two coincidence matrices $\mathbf{\Psi}^{(1)}$ and $\mathbf{\Psi}^{(2)}$. Consequently, this measure is also to be maximized. A final example is the *Hubert index*

$$Q_{\text{Hubert}}\big(\mathbf{\Psi}^{(1)},\mathbf{\Psi}^{(2)}\big)=\frac{N_{..}N_{SS}-N_{S.}N_{.S}}{\sqrt{N_{S.}N_{.S}N_{D.}N_{.D}}},$$

which may either be interpreted as a product-moment correlation, computed from the set of pairs $\big(\psi_{jl}^{(1)},\psi_{jl}^{(2)}\big)$, $1\le j,l\le n$. Alternatively, it may be interpreted as the square root of the (normalized) χ^2 measure, as it can be computed from the 2×2 contingency table shown in Table II.[2] Hence this measure is also to be maximized (like the preceding ones).

It should be clear that this list does not exhaust all possibilities. Basically all of the abundance measures by which (binary) vectors and matrices can be compared are applicable.

[2]The χ^2 measure can be seen as measuring the strength of dependence between two random variables, one for each partition, which indicate for each data point pair whether the data points are in the same group or not.

III. Resampling

Resampling [15] can be seen as a special *Monte Carlo method*, that is, as a method for finding solutions to mathematical and statistical problems by simulation [13], [16]. It has been applied to cluster estimation problems already fairly early [20], [8] and it seems to have gained increased attention in this domain recently [22], [26], [21]. Its main purpose in clustering is the validation of clustering results as well as the selection of an appropriate cluster model—in particular the choice of an appropriate number of clusters—by estimating the variability (or, equivalently, the stability) of the result.

Resampling methods can be found with basically two sampling strategies. In the first place, one may use *subsampling*, that is, the samples are drawn without replacement from the given data set, so that each data point appears in at most one data subset. This strategy is usually applied in a cross validation style, that is, the given data set is split into a certain number of disjoint subsets (with two subsets being the most common choice). The alternative is *bootstrapping* [11], in which samples are drawn with replacement, so that a data point may even appear multiple times in the same data subset. There are good arguments in favor and against both approaches, but the result often do not differ much.

The general idea of applying resampling for cluster validation and model selection was already outlined in the introduction: a cluster model can usually be applied as a classifier with as many classes as there are clusters (i.e. one class per cluster). In this way data points that have not been used to build the cluster model can be assigned to clusters (or the corresponding classes). Thus we obtain, with the same algorithm, two different groupings of the same set of data points. For example, one may be obtained by clustering the data set, the other by applying a cluster model that was built on another data set. These two groupings can be compared using, for example, one of the measures discussed in the preceding section. By repeating such comparisons with several samples drawn from the original data set, one can obtain an assessment of the variability of the cluster structure (or, more precisely, an assessment of the variability of the evaluation measure for the similarity of partitions).

Specific algorithms following this general scheme have been proposed in [22], [26], [21]. The approaches by [22] and [21] are basically identical. Both are based on a bootstrapping approach and work as follows: first the full given data set is clustered with the chosen algorithm. Formally, this may be seen as an estimate of the "average" partition [21]. Then a user-defined number of random samples of user-defined size are drawn (with replacement) from the data set and clustered as well. The cluster models obtained from the samples are applied to the full data set, thus obtaining two groupings of this data set. These two groupings are compared by one of the relative evaluation measures based on coincidence matrices that were discussed in Section II. Finally, the average of the evaluation measure values for each of these comparisons is taken as an assessment of the cluster variability. As an alternative, [21] mention that one may do without an estimate for the "average" partition (which is estimated by the cluster model obtained from the full data set) and rather assess the variability of the cluster structures by comparing all pairs of cluster models obtained from the samples on the full data set.

This resampling approach may be applied to select the most appropriate cluster model, in particular, the "best" number of clusters, by executing the above algorithm for different parameterizations of the clustering algorithm and then to select the one showing the lowest variability. Experimental results reported by [21] indicate that this approach is very robust and a fairly reliable way of choosing the number of crisp clusters.

In contrast to the bootstrapping approaches, [26] rely on a (repeated) two-fold cross validation sampling scheme. In each step the given data set is split randomly into two parts of about equal size. Both parts are processed with the same clustering algorithm and the cluster model obtained on the second half of the data is applied to the first half. Thus one obtains two groupings for the first half of the data, which are compared with a risk-based evaluation measure. This (relative) measure is defined on the two partition matrices and thus has to find the best matching of the clusters of the two groupings (see above). However, in principle all relative measures discussed in the preceding section (including those based on coincidence matrices) may be applied (just as measures based on partition matrices may be applied in the bootstrapping approaches by [22], [21]). [26] report experimental results on several data sets, which show that the number of Gaussian distribution clusters can thus be selected in a fairly reliable way.

When applying these resampling methods it should be noted that all approaches in this direction only assess the variability in the results obtained with some clustering algorithm. Although a low variability is surely a desirable property, it is not sufficient to guarantee a good clustering result. For example, a clustering algorithm that always yields the same partition of the data space, regardless of the data it is provided with, has no variability at all, but surely yields unsatisfactory clustering results [21]. Hence the clustering algorithms that are compared with such schemes should not differ too much in their flexibility, because otherwise the simpler and thus more stable algorithm may be judged superior without actually being.

Furthermore, [16], [17] remark that the power of many such statistical tests, like the estimation of the variability of the clustering structure as it was discussed above, decreases quickly with increasing data dimensionality. This is not surprising, because due to what is usually called the *curse of dimensionality*, the data space necessarily is less and less densely populated, the more dimensions there are. In addition, the noise in the different dimensions tends to sum, which in combination with the tendency of larger average distances between the data points [10], makes it more and more difficult for a clustering algorithm to find reasonable groups in the data. This, of course, must lead to a higher variability in the clustering result. For low-dimensional data sets, however, resampling is a very powerful technique and seems to be the best available approach to determine the number of clusters.

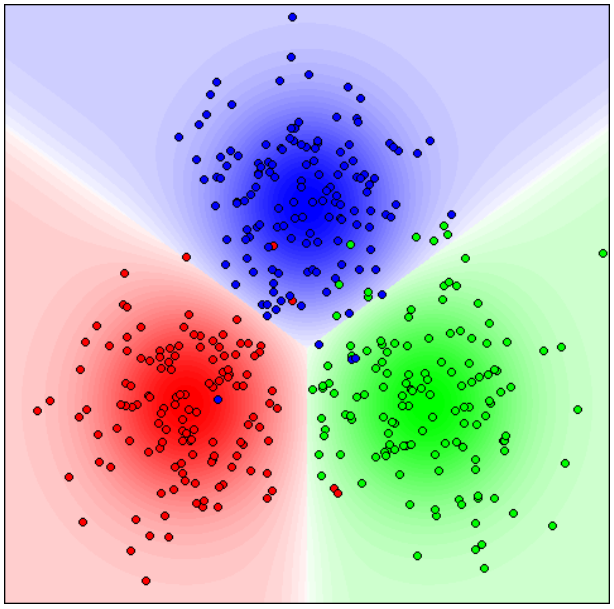

Fig. 1. An artificial data set with 3 clusters (equally populated)

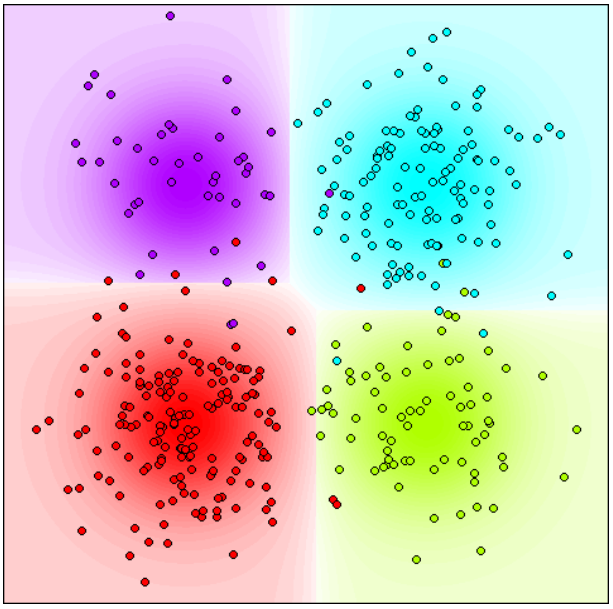

Fig. 2. An artificial data set with 4 clusters (differently populated)

TABLE III
HARD CLUSTERING RESULTS ON ARTIFICAL DATA SET (3 CLUSTERS)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.2364	.7637	.7622	.6994	.6046	.7222	.3950
3	**.0032**	**.9968**	**.9953**	**.9936**	**.9810**	**.9903**	**.9855**
4	.1235	.8765	.6849	.8833	.6546	.7859	.7058
5	.1281	.8719	.6107	.8613	.5520	.7058	.6171
6	.1074	.8926	.6475	.8707	.4898	.6507	.5716
7	.1169	.8831	.5607	.8610	.4114	.5792	.4967
8	.0919	.9081	.6155	.8855	.3946	.5643	.4987

TABLE IV
FUZZY CLUSTERING RESULTS ON ARTIFICAL DATA SET (3 CLUSTERS)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.0460	.6513	.6510	.5553	**.3851**	**.5554**	.1105
3	**.0004**	**.8082**	**.7119**	.6992	.3775	.5481	**.3227**
4	.0090	.7936	.5802	.7134	.2733	.4293	.2379
5	.0363	.7846	.4557	.7364	.2109	.3482	.1830
6	.0119	.8280	.4848	.7719	.1877	.3161	.1792
7	.0164	.8393	.4352	.7943	.1662	.2849	.1648
8	.0122	**.8582**	.4309	**.8154**	.1548	.2681	.1624

TABLE V
HARD CLUSTERING RESULTS ON ARTIFICAL DATA SET (4 CLUSTERS)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.0831	.9169	.9166	.8538	.7569	.8542	.7076
3	.0810	.9190	.8388	.9091	.8110	.8794	.8070
4	**.0044**	**.9956**	**.9882**	**.9938**	**.9810**	**.9894**	**.9850**
5	.1001	.8999	.7292	.8981	.6575	.7863	.7202
6	.0983	.9017	.7045	.8818	.5129	.6776	.6066
7	.0918	.9082	.6551	.8932	.5286	.6822	.6186
8	.0739	.9261	.6769	.9089	.5021	.6664	.6144

TABLE VI
FUZZY CLUSTERING RESULTS ON ARTIFICAL DATA SET (4 CLUSTERS)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	**.0001**	.7861	**.7849**	.6636	**.4982**	**.6650**	.3271
3	.0157	.7867	.6598	.6886	.3800	.5503	.3121
4	**.0009**	**.8351**	.6441	.7402	.3520	.5207	**.3425**
5	.0203	.8183	.5373	.7503	.2541	.4052	.2472
6	.0150	.8358	.5079	.7755	.2119	.3497	.2141
7	.0132	.8518	.4813	.8004	.1881	.3166	.1997
8	.0159	**.8607**	.4384	**.8176**	.1706	.2914	.1868

IV. EXPERIMENTS

We applied a resampling approach for hard and fuzzy clustering based on the above explanations to four data sets. The first three are artifical two-dimensional data sets of 400 data points each with three, four, and six clusters, respectively, which are shown in Figures 1 to 3. They were generated by sampling from normal distributions (variance 1), located at $(0,0)$, $(4,0)$, and $(2,3)$ for the first data set (equal cluster probabilities), at $(0,0)$, $(4,0)$, $(0,4)$, and $(4,4)$ for the second data set (different cluster probabilities), and at $(0,0)$, $(2,-3)$, $(6,-3)$, $(8,0)$, $(6,3)$, and $(2,3)$ for the third data set (equal cluster probabilities). The fourth data set is the well-known wine data set from the UCI machine learning repository [6]. It comprises three classes of Italian wines and thus one expects to find three clusters.

When clustering all datasets were normalized in all dimensions to mean 0 and standard deviation 1 to rule out scaling effects. The experiments were carried out with a scheme that lies between the two schemes that were discussed in the preceding section. First the whole data set was clustered. Then 100 random samples without replacement were drawn from the data set, each of which comprised about half of the data

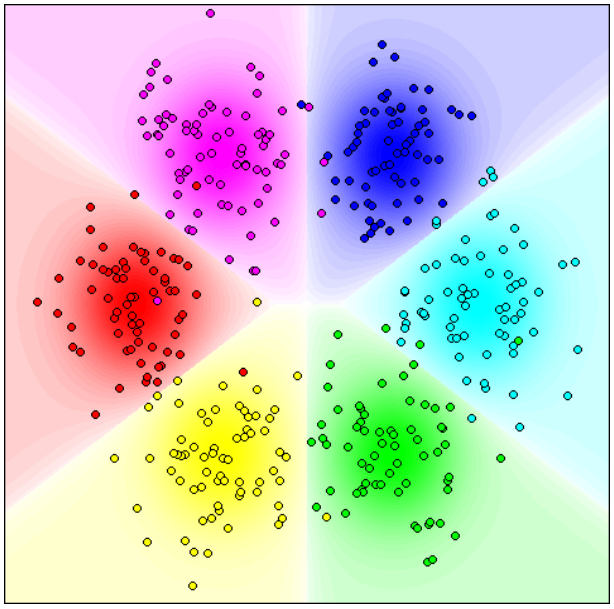

Fig. 3. An artificial data set with 6 clusters (equally populated)

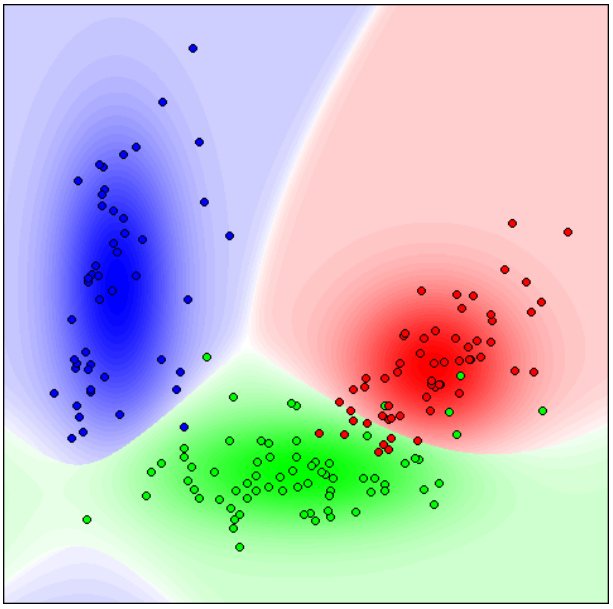

Fig. 4. The wine data set (attributes 7 and 10)

TABLE VII

HARD CLUSTERING RESULTS ON ARTIFICAL DATA SET (6 CLUSTERS)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.2393	.7607	.7561	.6827	.5607	.6933	.3667
3	.1989	.8011	.6816	.7435	.5131	.6669	.4597
4	.1527	.8473	.6698	.8296	.5619	.7074	.5894
5	.1397	.8603	.5665	.8457	.5448	.6969	.5955
6	**.0593**	**.9407**	**.7931**	**.9457**	**.7637**	**.8631**	**.8326**
7	**.0592**	**.9408**	.7109	**.9465**	.7382	.8465	.8164
8	.0624	.9376	.6612	.9447	.6925	.8169	.7861

TABLE VIII

FUZZY CLUSTERING RESULTS ON ARTIFICAL DATA SET (6 CLUSTERS)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.0079	.5178	.5178	.5007	.3339	.5007	.0015
3	.0403	.6557	.4832	.5846	.2319	.3762	.0648
4	.0658	.7630	.5248	.6924	.2374	.3830	.1781
5	.0365	.8752	.6867	.8168	.3705	.5395	.4251
6	**.0000**	**.9628**	**.8885**	**.9308**	**.6535**	**.7904**	**.7490**
7	.0221	.9311	.7054	.9166	.5635	.7206	.6716
8	.0305	.9170	.6133	.9075	.4843	.6522	.5988

TABLE IX

HARD CLUSTERING RESULTS ON THE WINE DATA SET (3 CLASSES)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.1664	.8336	.8303	.7685	.6866	.7910	.5322
3	**.0239**	**.9761**	**.9625**	**.9528**	**.8752**	**.9328**	**.8967**
4	.1008	.8992	.7724	.8848	.6790	.7905	.7113
5	.0732	.9268	.8062	.8996	.6390	.7704	.7069
6	.0828	.9172	.7323	.8976	.5637	.7149	.6537
7	.0851	.9149	.6225	.8903	.5243	.6846	.6195
8	.0735	.9265	.6428	.9056	.5027	.6644	.6099

TABLE X

FUZZY CLUSTERING RESULTS ON THE WINE DATA SET (3 CLASSES)

	partition matrix			coincidence matrix			
#	diff	acc	F1	Rand	Jaccard	Folkes	Hubert
2	.0102	.7084	**.7048**	.5900	**.4231**	**.5944**	.1798
3	**.0013**	**.7945**	.6853	.6781	.3633	.5329	**.2874**
4	.0244	.7779	.5503	.6980	.2633	.4166	.2129
5	.0056	.8234	.5569	.7436	.2351	.3806	.2190
6	.0125	.8296	.4923	.7798	.1918	.3219	.1833
7	.0115	.8434	.4500	.7929	.1689	.2891	.1679
8	.0133	**.8546**	.4107	**.8114**	.1535	.2662	.1580

points. (The data set was split into two equal parts, one of which was used). Each sample was clustered with the same number of clusters as the full data set and then the two cluster structures (one obtained from the full data set and one from the sample) were compared on the full data set using the measures described in Section II. The evaluation results were averaged over the 100 samples, thus yielding a stability measure.

The results of our experiments are shown in the eight Tables III to X. They are grouped in pairs, with the first table referring to crisp clustering (classical c-means) and the second to fuzzy clustering (standard fuzzy c-means). The first column of these tables states the number of clusters, the next three results for measures comparing partition matrices, the last four results for measures comparing coincidence matrices.

As can be seen from these tables all measures work fairly well in the crisp case: the best value is obtained for the true or expected number of clusters. Only for the artificial data set with six clusters some measures seem to weakly prefer seven clusters instead of six. In the fuzzy case, however, for three and four clusters (on both the artificial and the real world data set) the Rand statistic fails completely, the Jaccard index and the Folkes–Mallows index fail or indicate the right number of

clusters only with a weak local maximum. Similar observations can be made about the (cross-classification) accuracy and the F_1-measure. For six clusters, on the other hand, all these measures clearly indicate the "correct" number of clusters, and in an even clearer way than they do for crisp clustering. However, the sum of squared differences and the Hubert index yield excellent results in all cases (regardless of the number of clusters) and thus appear to be the methods of choice for fuzzy clustering. Especially the sum of squared differences is particulary clear in its selection behavior and can even exhibit a local minimum at twice the number of clusters, proving its high sensitivity to the cluster structure.

V. Conclusions

In this paper we transferred resampling ideas that have been used in classical crisp clustering to fuzzy clustering and introduced the mean square error as a simple, but effective measure for comparing fuzzy and probabilistic partition matrices. As the experiments show, the resampling approach is applicable for fuzzy clustering as well, but one has to be careful which relative cluster evaluation measure to choose: not all measures that work with crisp clustering also work with fuzzy clustering, at least for a low number of clusters. The best results we obtained with a direct comparison of the partition matrices based on the mean squared difference. A close competitor is the Hubert index, which is equally clear.

References

[1] G.H. Ball and D.J. Hall. A Clustering Technique for Summarizing Multivariate Data. *Behavioral Science* 12(2):153–155. 1967

[2] J.C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York, NY, USA 1981

[3] J.C. Bezdek and N. Pal. *Fuzzy Models for Pattern Recognition*. IEEE Press, New York, NY, USA 1992

[4] J.C. Bezdek, J. Keller, R. Krishnapuram, and N. Pal. *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Kluwer, Dordrecht, Netherlands 1999

[5] J. Bilmes. A Gentle Tutorial on the EM Algorithm and Its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. University of Berkeley, Tech. Rep. ICSI-TR-97-021, 1997

[6] C.L. Blake and C.J. Merz. *UCI Repository of Machine Learning Databases*. University of California, Irvine, CA, USA 1998 http://www.ics.uci.edu/˜mlearn/MLRepository.html

[7] C. Borgelt. *Prototype-based Classification and Clustering*. Habilitation thesis, University of Magdeburg, Germany 2005

[8] J. Breckenridge. Replicating Cluster Analysis: Method, Consistency and Validity. *Multivariate Behavioral Research* 24:147–161. Lawrence Erlbaum Associates, Mahwah, NJ, USA 1989

[9] A.P. Dempster, N. Laird, and D. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society (Series B)* 39:1–38. Blackwell, Oxford, United Kingdom 1977

[10] C. Döring, C. Borgelt, and R. Kruse. Effects of Irrelevant Attributes in Fuzzy Clustering. *Proc. 14th IEEE Int. Conference on Fuzzy Systems (FUZZ-IEEE'05, Reno, NV, USA)*, on CDROM. IEEE Press, Piscataway, NJ, USA 2005

[11] B. Efron and R.J. Tibshirani. An Introduction to the Boostrap. Chapman & Hall, London, United Kingdom 2003

[12] B.S. Everitt and D.J. Hand. *Finite Mixture Distributions*. Chapman & Hall, London, United Kingdom 1981

[13] B.S. Everitt. *The Cambridge Dictionary of Statistics*. Cambridge University Press, Cambridge, United Kingdom 1998

[14] Y. Fukuyama and M. Sugeno. A New Method of Choosing the Number of Clusters for the Fuzzy c-Means Method. *Proc. 5th Fuzzy Systems Symposium (in Japanese)*, 247–256. Japan Society for Fuzzy Sets and Systems, Kobe, Japan 1989

[15] P. Good. *Resampling Methods*. Springer-Verlag, New York, NY, USA 1999

[16] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. Clustering Validity Checking Methods: Part I. *ACM SIGMOD Record* 31(2):40–45. ACM Press, New York, NY, USA 2002

[17] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. Clustering Validity Checking Methods: Part II. *ACM SIGMOD Record* 31(3):19–27. ACM Press, New York, NY, USA 2002

[18] J.A. Hartigan and M.A. Wong. A k-means Clustering Algorithm. *Applied Statistics* 28:100–108. Blackwell, Oxford, United Kingdom 1979

[19] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis*. J. Wiley & Sons, Chichester, England 1999

[20] A.K. Jain and J. Moreau. Bootstrap Technique in Cluster Analysis. *Pattern Recognition* 20:547–569. Pergamon Press, Oxford, United Kingdom 1986

[21] M.H.C. Law and A.K. Jain. *Cluster Validity by Bootstrapping Partitions*. Technical Report MSU-CSE-03-5, Dept. of Computer Science and Engineering, Michigan State University, Michigan, , USA 2003

[22] E. Levine and E. Domany. Resampling Method for Unsupervised Estimation of Cluster Validity. *Neural Computation* 13:2573–2593. MIT Press, Cambridge, MA, USA 2001

[23] S. Lloyd. Least Squares Quantization in PCM. *IEEE Trans. on Information Theory* 28:129–137. IEEE Press, Piscataway, NJ, USA 1982

[24] C.H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization, Algorithms and Complexity*. Prentice-Hall, Englewood Cliffs, NJ, USA 1982

[25] C.J. van Rijsbergen. *Information Retrieval*. Butterworth, London, United Kingdom 1979

[26] V. Roth, T. Lange, M. Braun, and J.M. Buhmann. A Resampling Approach to Cluster Validation. *Proc. Computational Statistics (CompStat'02, Berlin, Germany)*, 123–128. Springer-Verlag, Heidelberg, Germany 2002

[27] M.P. Windham. Cluster Validity for the Fuzzy c-Means Algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)* 4(4): 357–363. IEEE Press, Piscataway, NJ, USA 1982

[28] X.L. Xie and G.A. Beni. Validity Measure for Fuzzy Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 3(8):841–846. IEEE Press, Piscataway, NJ, USA 1991. Reprinted in [3], 219–226