A Cluster Validity Approach based on Nearest-Neighbor Resampling

Ulrich Möller, Dörte Radke
Leibniz Institute for Natural Product Research and Infection Biology – Hans-Knöll Institute
Ulrich.Moeller@hki-jena.de

Abstract

We introduce an approach for validating clustering results based on partition stability under a nearest-neighbor resampling. The approach is relatively robust, efficient, and avoids conceptual problems of other common validation strategies. Encouraging results compared to those of subsampling-based consensus clustering are presented for simulated data and (tumor) gene expression benchmark data sets. The proposed method is discussed in view of future applications to unsupervised learning from sample data.

1. Introduction

In various research fields cluster analysis is used for (unsupervised) statistical learning from sample data. Often the results are utilized for drawing inference about a population underlying the sample. Therefore, a measure of confidence in the learning result is desired. Confidence is extremely important, if the sample is sparse, if there are much more variables (features) than observations, and if the data are noisy (due to population variability and statistical errors in the measurement system). All of this applies when gene expression data are clustered for discovering classes among patients who suffer from a disease with a genetic background. We consider the currently most prominent application: the refinement of a tumor taxonomy that is expected to result in improved prognosis and therapy success. This is of considerable social and economical relevance. However, confidence in clustering results is a general problem and the challenge of sparse, high-dimensional, and noisy data may occur in any field where complex technologies permit the recording of many features simultaneously.

Statistical confidence in clustering results is mainly obtained in two ways [1]: i) partitions of the sample data are tested against partitions of 'null' data generated under a null hypothesis H_0 (e.g. [2]), ii) data sets are randomly resampled from the original data and a partition is determined whose cluster assignments were the most stable under resampling. The resampling approach avoids the (perhaps unjustified) assumption H_0 (e.g., uniform distribution) and makes use of the emerging computing power [3].

Applications of resampling to gene expression data utilized bootstrapping (drawing with replacement) [4], subsampling (drawing without replacement) [5], and cross validation (data splitting) [2]. Those resamples lack a considerable part of the original information. This may prevent the recognition of subpopulations if the sample is sparse already (i.e., the model may not be identified). Moreover, the information loss prevents a straightforward cluster alignment across resample partitions for assessing their stability. Instead, a consensus matrix M is calculated from pairwise cluster membership consensus across different resamples. To obtain a full partition, M is converted into a dissimilarity matrix used as input into a second clustering step. Here a method is required that accepts a distance matrix rather than a set of feature vectors. Due to the randomness of the information loss, a relatively large number of resamples is required to ensure that each original item is adequately represented in the set of resamples.

An alternative technique without information loss is perturbation (adding noise to the data). An empirical study robustly indicated that perturbation usually outperforms bootstrapping and subsampling [7]. Choosing the perturbation strength seems to be not as crucial [7], whereas the empirical choice of the subsampling size is often difficult [8]. However, adding to all data points noise of the same variance (cf. [6]) may restrict the identification of subpopulations with different variability which are observed in real data.

We present a nearest-neighbor resampling (NNR) approach that offers a solution to both problems: information loss and empirical control of the degree of change made to the original data. NNR techniques were used for time series analysis in hydrology and climatology [9]. We introduce a NNR method for cluster validation and compare the results to those of subsampling-based consensus clustering [5] with applications to simulated data and tumor gene expression data.

0-7695-2521-0/06 $20.00 © 2006 IEEE

2. Nearest-neighbor resampling method

Given a data set **X** an NNR resample was generated as follows:

(1) Calculate *D*, the mean (Euclidean) distance of a data point *X* and its original *k* nearest neighbors from the center of these *k*+1 points.

(2) Move *X* in the feature space to a random position on the hyper-sphere with radius *D* around *X*.

(3) Perform the steps 1 and 2 for all members of **X**. This approach aims at simulating samples with two characteristics: the local scattering of the data, as presumably caused by random intra-class variability, is randomized and inter-class differences are preserved. In contrast, adding noise of a fixed variance could dissolve compact well-separated clusters where clusters with a high variance remain almost unchanged. The nearest-neighbor technique is used as a simple means to obtain local parameter (variance) estimates from the original data without *a priori* knowing classes. Like in classification we assume that the *k* nearest neighbors of item *X* essentially belong to the same class as *X*. Hence, parameter *k* should be clearly smaller than the size of the smallest cluster. The steps 1 and 2 are somewhat arbitrary. Other definitions, including the metric, could be used. However, we are not interested in a perfect variance estimate. The perturbations should be large enough to induce differences in superimposed false cluster boundaries across the resamples. Then low confidence is assigned to the underlying (inappropriate) clustering settings, in particular, the specified number of clusters.

3. Choice of a clustering algorithm

We are interested here in a proof of principle of the NNR approach for cluster *validation*. Hence, for cluster *generation* any method could be used that is shown to provide reasonable partitions for simulated data and real benchmark (or gold standard) examples. We selected the fuzzy C-means (FCM) algorithm [1] with a fuzzy exponent between 1.1 and 1.5 and retained the best result of *T* runs in order to avoid poor local minima of the FCM objective function (*T* was set between 20 and 50). This setup proved to be appropriate in FCM applications to gene expression data [7, 11].

4. Method for assessing cluster stability

For a fixed number of clusters *C* we define the normalized overlap of cluster *i* of resample partition *r* (reference cluster Θ_{ri}) and the best-matching cluster (*j*) of resample partition *q*,

$$\hat{s}_{ri}(q) = \max_{j=1,..,C} \frac{|(\Theta_{ri} \cap \Theta_{qj})|}{|(\Theta_{ri} \cup \Theta_{qj})|},$$

where $q = 1,.., R$ and *R* is the number of resample partitions for each number of clusters. Let

$$\widetilde{S}_r(i) = \frac{1}{R_0} \sum_{l=1}^{R_0} \widetilde{s}_{ri}(l),$$

be the average stability of cluster Θ_{ri} over the R_0 largest pairwise similarities $\{\widetilde{s}_{ri}(1),..., \widetilde{s}_{ri}(l)..., \widetilde{s}_{ri}(R_0)\}$. Then the stability index of cluster Θ_{ri} is defined as

$$S_{ri} = \begin{cases} 0 \;; & \sum_{q=1}^{R} I(\hat{s}_{ri}(q) > s_0) < R_0, \\ \widetilde{S}_r(i); & \textit{otherwise} \end{cases}$$

where *I* is the indicator function. Here, cluster stability is filtered twice: the minimum pairwise similarity threshold s_0 has to be reached for at least R_0 resample partitions. $2 \leq R_0 \leq R$ and $0 < s_0 \leq 1$ can be specified. Values near the upper bounds extract very stable clusters, if any. Small values yield clusters of accordingly low stability. Then we calculate a (0,1)-scaled global stability for *C*-cluster resample partitions

$$GS(C) = \frac{1}{RC} \sum_{r=1}^{R} \sum_{i=1}^{C} S_{ri}.$$

We expect to identify a unique data structure with $\hat{C}$ clusters by a single large value of *GS* at $\hat{C}$. Structures at different levels of resolution could be inferred from a few local maxima of *GS*. If *GS* is large for many values of *C* either the stability requirements or the perturbation of the original data may have been too weak. In this case, the analysis should be repeated with increased values of R_0, s_0, and/or the NNR parameter *k*. If *GS* is small for increasing values of *C* and moderate stability requirements, evidence of structure is missing. s_0 (minimum overlap) and R_0 (# overlapping partitions) are intuitively interpretable; their choice does not *a priori* depend on the data. However, depending on the actual partition overlap, data-specific values s_0 and R_0 may have to be used so that partition stability is adequately deducible from *GS*. Therefore, some variation of s_0 and R_0 would confirm the robustness of a *GS* pattern. We varied s_0 from 0.6 to 0.95 and selected typical results for the presentation, where the influence of R_0 is demonstrated (Fig. 2). $R = 10$ was selected ad hoc with successful empirical results for the data described below. Choosing $R_0 < R$ can exclude the effect of a minority of results due to inappropriate resampling or clustering. s_0 and R_0 allow a detection even of stable substructure anywhere in the data. This flexibility is missing in other statistical measures of partition stability such as the Jaccard index or Rand index (cf. [1]).

5. Data sets and results

Results are presented for three simulated data sets and two benchmark data sets downloaded from the supplementary website to [5] (Table 1, Figure 1 top).

Simulated15: The maximum cluster stability $GS = 1$ for C = 2, 3, and 15 clusters clearly reveals both the fine and coarse data structure. Subsequent clustering of the (original) data with C = 15 provided an adequate description of the simulated gaussian clusters (Fig. 1).

Simulated4: When using sufficient (≥6) resamples GS exhibited a clear global maximum at 4 clusters. This is the correct estimate although the *fuzzy* clustering may have little overestimated the three small clusters (Fig. 1). Interestingly, an artificial (false) coarse structure, often 'detected' by a number of estimators, is not indicated. The local maximum of GS at $C = 2$ for $R_0 = 4$ becomes obvious here as an artifact of requiring stability across too few resamples.

Simulated5: This dataset has overlapping (gaussian) clusters and, among the simulated data, this model is most difficult to detect. Here we have a false positive peak at $C = 2$. Nevertheless, the next clear peak of GS provides the correct estimate $C = 5$ when requiring stability for at least $R_0 = 6$ resamples. For $R_0 = 10$ the spurious small peak at $C = 10$ disappears. Using $C = 5$ for clustering the original dataset leads to the true model (Fig. 1, right).

The two estimates obtained from consensus clustering for Simulated5 were 4-5 (unclear decision) and 4 [5]. Also the models of Simulated15 and Simulated4 with relatively distinct structures were not or not clearly identified. The original software offered by the authors of [5] was applied to our datasets.

Leukemia3: The stability index indicates a structure with 3 (GS = 1) or 4 (GS = 0.99) clusters. This result did not depend on R_0, but was the clearer the more stable resampling partitions were required. The 3-cluster partition of the original data agrees almost perfectly with the leukemia classes expected due to the data preprocessing that selected phenotype-specific genes (cf. [5]). The feature vector of one patient was not assigned to the same cluster as the other members of this phenotype class. This assignment, however, was consistent for $C > 3$ and in other studies using unsupervised and supervised classifiers. Consensus clustering estimated 5 and 4 clusters for Leukemia3.

Leukemia6: The stability index GS remained at a high level up to $C = 5$ and then markedly decreased for all values of R_0. Similarly, consensus clustering estimated 5 clusters. The smallest class (leukemia subtype BCR-ABL, $n = 15$) was not distinguished from another class ("hyperdiploid>50", $n = 64$). Further 16 (of 248)

Table 1. Characteristics of the data sets
p: number of features, *k*: NNR-parameter

Name	Cluster sizes	p	C_{true}	k
Simulated4	200, 3 × 50	2	4	10
Simulated5 [5]	5 × 100	2	5	5
Simulated15	15 × 20	2	15	10
Leukemia3 [5]	11, 8, 19	999	3	4
Leukemia6 [5]	15,27,64,20,43,79	985	6	10

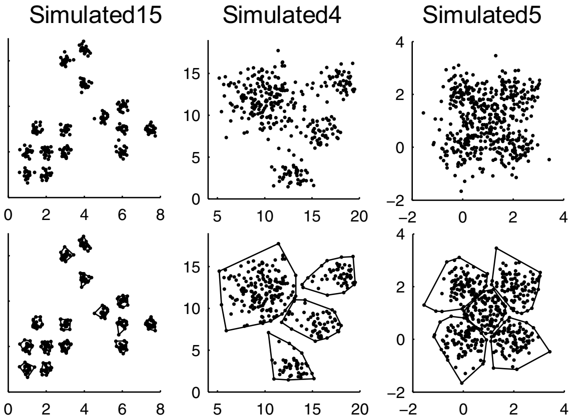

Figure 1. Data sets (top) and clustering results

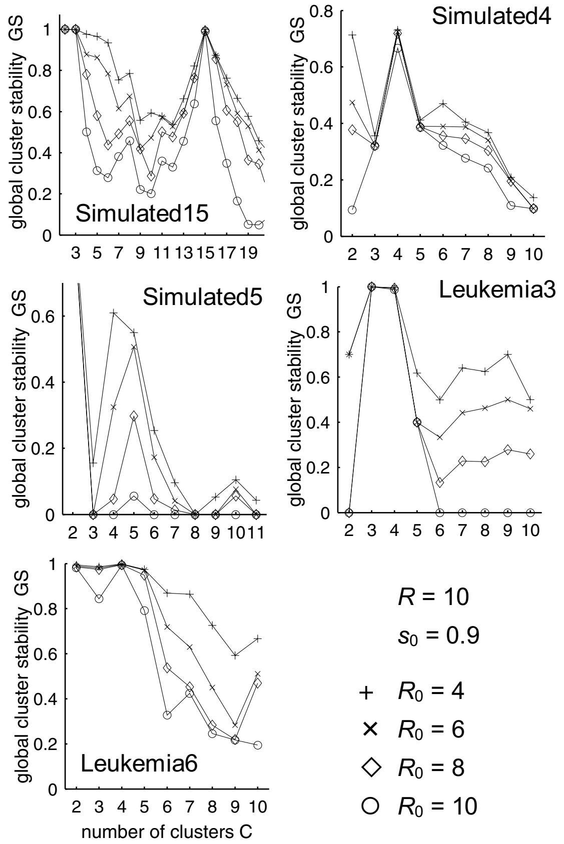

Figure 2. Global stability vs. number of clusters

patients were 'misclassified'. A good class separation by consensus clustering was reported (without a precise table) for C = 7 [5]. However, these results are based on a two-way data normalization and hierarchical clustering not used in our protocol.

5. Discussion and conclusions

Inspired from empirical results of a large-scale study [7], we introduced a resampling-based approach to cluster validation which has conceptual advantages over related approaches: i) the generation of a resample does not suffer from a loss of original information, ii) each resample contains a data point derived from each original data point (one-to-one mapping), iii) random intra-class variability is simulated via NNR without knowing the classes, where inter-class differences are more or less preserved (i.e., a simulation of unavailable original samples). As described in sections 1 and 2, common resampling schemes lack at least one of these features. i) is important if the original sample is sparse so that further increase of sparseness may impair model identification. ii) enables the use of straightforward methods for assessing stability of full partitions as well as individual clusters. A bias due to under- or over-representation of original data points in the set of resamples is avoided. Additional (heuristic) consensus clustering (cf. section 1 and [5]) is possible, but not necessary due to i) and ii). The number of clusters can be estimated based on cluster overlap (section 4). The final C-cluster partition or stable parts thereof can be derived from a one-to-one cluster alignment of R resample partitions with C clusters: the probability that item X belongs to cluster c, $1 \le c \le C$, can be estimated by the relative frequency of assignment of X to cluster c in the R aligned resample partitions

The results obtained in this study are encouraging. Distinct clustering structures were identified clearly and correctly. Indications of less distinct structures were also obtained. This was an improvement over state-of-the-art methods: consensus clustering (CC) and the so-called gap statistic, each based on either hierarchical clustering (HC) or self-organizing maps (SOM). The results for simulated and gene expression benchmark data suggest that the NNR-based partition stability approach can be successfully used for unsupervised learning to answer open research questions. For this purpose also the individual and cluster-specific stability values under NNR can be interpreted.

Apart from conceptual features, our NNR application showed a practical advantage: only 10 resamples were sufficient to identify the analyzed data structures. Accuracy and/or clarity of the results were similar to or even increased over those obtained via consensus clustering based on 500 and 200 subsamples for HC and SOM, respectively (the latter number was chosen in [5], because SOM is slower than HC). Subsamples of 80% of the data cause a smaller computational effort compared to NNR resamples. However, a feasible decrease in the number of data sets by more than one magnitude (500:10) is the most relevant factor that determines computation time. Clustering usually requires the by far largest computation times in this data processing line. Therefore, we can neglect the differences between the cluster validity methods considered with respect to the computational effort of resample generation and stability calculation.

The NNR parameter k could be optimized for each cluster or individual item based on cluster sizes estimated in a pre-analysis. Moreover, other modeling techniques than nearest neighbors could be used to determine the local perturbation parameter(s).

Acknowledgement: This work has been supported by the German BMBF, grant no. 0312704 D. We thank Martin Hoffmann who encouraged us in using nearest-neighbor information for adjusting data resamples. We also thank Christian Bodenstein for providing consensus clustering results for the simulated data sets.

6. References

[1] S. Theodoridis and K. Koutroumbas, *Pattern recognition*, San Diego: Academic Press, 1999.

[2] S. Dudoit and J. Fridlyand, A prediction-based resampling method for estimating the number of clusters in a dataset, *Genome Biol 3* (2002), RESEARCH0036.

[3] C.E. Lunneborg, *Data analysis by resampling - concepts and applications*, Pacific Grove: Duxbury Press, 2000.

[4] S. Dudoit and J. Fridlyand, Bagging to improve the accuracy of a clustering procedure, *Bioinformatics 19*, 2003, pp. 1090-1099

[5] S. Monti, P. Tamayo, J. Mesirov, and T. Golub, "Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data", *Machine Learning 52*, 2003, pp. 91-118.

[6] M. Bittner et al. (24 co-authors), "Molecular classification of cutaneous malignant melanoma by gene expression profiling", *Nature 406* (2000), 536-540.

[7] U. Möller and Dörte Radke, "Performance of data resampling methods based on clustering", to appear in *Intelligent Data Analysis 10(2)*, 2006.

[8] A.C. Davison, D.V. Hinkley, and G.A. Young, "Recent developments in bootstrap methodology", *Statistical Science 18*, 2003, pp. 141-157.

[9] T. Brandsma and T.A. Buishand, "Simulation of extreme precipitation in the Rhine basin by nearest-neighbour resampling", *Hydrology and Earth System Sciences 2*, 1998, 195-209.