International Journal of Computer Sciences and Engineering Systems
Vol. 10 No. 1 (June, 2016)

An Optimized Approach on Applying Genetic Algorithm to Adaptive Cluster Validity Index

Tzu-Chieh LIN[1], Hsiang-Cheh HUANG[2], Bin-Yih LIAO[1] & Jeng-Shyang PAN[1]

[1]Department of Electronic Engineering, National Kaohsiung University of Applied SciencesKaohsiung, 807, Taiwan
E-mail:{tclin, byliao, jspan}@bit.kuas.edu.tw

[2]Department of Electrical Engineering, National University of KaohsiungKaohsiung, 811, Taiwan
E-mail: hchuang@nuk.edu.tw

Abstract: *The partitioning or clustering method is an important research branch in data mining area, and it divides the dataset into an arbitrary number of clusters based on the correlation attribute of all elements of the dataset. Most datasets have the original clusters number, which is estimated with cluster validity index. But most methods give the error estimation for most real datasets. In order to solve this problem, this paper applies the optimization technique of genetic algorithm (GA) to the new adaptive cluster validity index, which is called the Gene Index (GI). The algorithm applies GA to adjust the weighting factors of adaptive cluster validity index to train an optimal cluster validity index. It is tested with many real datasets, and results show the proposed algorithm can give higher performance and accurately estimate the original cluster number of real datasets compared with the current cluster validity index methods.*

Keywords: *Clustering, Genetic Algorithm, Cluster Validity Index, Optimization, Data Mining*

1. INTRODUCTION

Data partitioning is commonly encountered in real applications. Lots of schemes are proposed to assess the performances for specific algorithms in literature. The main concern of data partitioning is how to correctly divide the data points into clusters. Some algorithms in literature are specifically designed for certain databases. Thus, these may perform well in some cases but not always good in general. In this paper, we would like to propose a generalized scheme, which is integrated with optimization techniques, for better partitioning the data.

There are a number of indices proposed in literature to assess the performances of data clustering. The main ideas are twofold: (1) data points within the same cluster should locate as close as possible, and (2) data points in different clusters should be as apart as possible. Based on the two concepts, a variety of the cluster validity indices are proposed. We make necessary simulations and verify that not all the indices perform well. Therefore, we employ the genetic algorithm (GA) [1] for resulting in better performances in data partitioning.

This paper is organized as follows. In Section 2 we point out the data partitioning schemes and the cluster validity indices. In Section 3 we describe the proposed algorithm by integrating existing indices and training with GA. Simulation results are demonstrated in Section 4. Finally, we conclude this paper in Section 5.

2. DATA PARTITIONING SCHEMES AND CLUSTER VALIDITY INDICES

In this paper, we employ the fuzzy C-means (FCM) [2] algorithm for data clustering, and then make comparisons among several indices. By using the concepts of fuzzy theory, every data point does not absolutely belong to a certain cluster; it is denoted by a floating number to represent the degree of belonging to a certain cluster.

The major drawback for FCM or other algorithms is that the correct number of clusters cannot be known exactly in advance. Thus, the cluster validity indices with several kinds of representations are proposed to evaluate the correct number of clusters. Every index has its advantages and drawbacks. We cite several commonly encountered indices; then we perform verifications in Sec. 2, and finally combine the advantages of these indices and propose the genetic-based cluster validity index in Sec. 3.

2.1 Cluster Validity Index: PC Index

PC (partition coefficient) index [3] was one of the measures used in early days, with the definition in Eq. (1):

$$V_{\mathrm{PC}}(U)=\frac{1}{n}\sum_{k=1}^{n}\sum_{i=1}^{c}u_{ik}^{2} \tag{1}$$

where u_{ik} denotes the degree of membership of x_i in the cluster k, x_i is the i^{th} of d-dimensional measured data

(and we use $d = 2$ here as an example), under the condition that

$$u_{ik} \in [0, 1], \quad \forall\, i, k\,;$$

$$\sum_{i=1}^{c} u_{ik} = 1, \quad \forall\, k\,.$$

To assess the effectiveness of clustering algorithm, the larger the PC index value, the better the performance.

2.2 Cluster Validity Index: PE Index

PE (partition entropy) index was also proposed in [3], with the definition in Eq. (2):

$$V_{\mathrm{PE}}(U) = \frac{-1}{n}\left\{\sum_{k=1}^{n}\sum_{i=1}^{c}\left[u_{ik}\cdot\log\left(u_{ik}\right)\right]\right\} \tag{2}$$

To assess the effectiveness of clustering algorithm, the smaller the PE index value, the better the performance.

2.3 Cluster Validity Index: XB Index

The XB index was proposed by Xie and Beni in [4] with the two important concepts of compactness and separation. For a good clustering result, the data points within the same cluster should be as compact as possible, while any two different clusters should be as far as possible. It can be formulated by Eq. (3):

$$V_{\mathrm{XB}}(U, V; X) = \frac{\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^{2}\left\|x_j - v_i\right\|^2}{n\cdot\min_{i\neq k}\left(\left\|v_i - v_k\right\|^2\right)} \tag{3}$$

where x_j is the i^{th} of d-dimensional measured data (and we use $d = 2$ here), v_k is the d-dimension center of the cluster.

In Eq. (3), the numerator implies the compactness and the denominator denotes the separation. Therefore, to assess the effectiveness of clustering algorithm, the smaller the XB index value, the better the performance.

2.4 Cluster Validity Index: K Index

The K index was proposed by Kwon [5] based on the improvement of the XB index. In Eq. (3), we find when $c \to n$, $V_{\mathrm{XB}} \to 0$, and it is generally incorrect for practical applications. By modifying Eq. (3), we obtain Eq. (4):

$$V_{\mathrm{K}}(U, V; X) = \frac{\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^{2}\left\|x_j - v_i\right\|^2 + \frac{1}{c}\sum_{j=1}^{c}\left\|v_i - \bar{v}\right\|^2}{\min_{i\neq k}\left(\left\|v_i - v_k\right\|^2\right)} \tag{4}$$

where $\bar{v}$ denotes the geometric center of data points.

To assess the effectiveness of clustering algorithm, the smaller the XB index value, the better the performance.

2.5 Cluster Validity Index: $\mathbf{B_{crit}}$ Index

$\mathrm{B_{crit}}$ index was proposed in [6]. It is also composed of the compactness and separation parameters in order to obtain the optimal number of clusters. The measure of compactness and separation are independently derived. First, the separation between clusters is denoted by $G(c)$,

$$G(c) = \frac{\max_{i,j}\delta(V_i, V_j)}{\min_{i\neq j}\delta(V_i, V_j)} \tag{5}$$

where $\delta(V_i, V_j)$ is a distance measure between the geometric centers of clusters i and j, with the definition of

$$\delta(V_i, V_j) = \left((V_i - V_j)^T A(V_i - V_j)\right)^{1/2} \tag{6}$$

and A denotes a positive definite matrix with dimension of $d \times d$ (or 2×2 here). For simplicity, people use the identity matrix I to replace the matrix A in Eq. (6) to verify the distance measure.

Next, the compactness is represented by the ratio of variances between the data points of the current cluster, and the data points within every cluster, denoted by $V_{wt}(c)$,

$$V_{wt}(c) = \frac{1}{c}\cdot\frac{\sum_{q=1}^{d}\sum_{k=1}^{c}\mathrm{var}_q(k)}{\sum_{q=1}^{d}\mathrm{var}_{\mathrm{total}}(q)} \tag{7}$$

where var_q denotes the current cluster and $\mathrm{var}_{\mathrm{total}}$ denotes the variance of the whole data set. From experimental results, the value of $G(c)$ is much larger than that of $V_{wt}(c)$ with the ranges of $G(c) \in [0, 20]$ and $V_{wt}(c) \in [0, 0.8]$, thus we need to include a weighting factor α to balance the effects from both factors, and we obtain

$$B_{\mathrm{crit}}(c) = G(c) + \alpha\cdot V_{\mathrm{wt}}(c) \tag{8}$$

where $\alpha = \frac{\max\ G(c)}{\max\ V_{wt}(c)}$ denotes the weighting factor.

From derivations above, when the smaller $\mathrm{B_{crit}}$ index is obtained, the clustering performance would be better.

2.6 Cluster Validity Index: SV Index

SV index was proposed in [7]. It also adopted the concepts of compactness and separation. Unlike the $\mathrm{B_{crit}}$ index in Sec. 0, both factors are normalized to the values between 0 and 1 to balance the effects from both factors. In measuring the compactness, the mean distance of the c clusters in the data set is calculated,

$$V_u(c, V; X) = \frac{1}{c}\sum_{i=1}^{c}\left[\frac{1}{n_i}\sum_{x\in X_i}\left\|V_i - x\right\|\right] \tag{9}$$

where n_i denotes the number of data points within cluster i, V_i is the geometric center of cluster i, and the total of c mean distances are calculated. The separation measure is simply denoted by $V_o = \frac{c}{d_{\min}}$, where $d_{\min}$ denotes the minimum distance between any two clusters.

Next, normalization of Eqs. (9) and (10) is performed by

$$V_{uN}(c, V; X) = \frac{V_u(c, V; X) - \min\left[V_u(c, V; X)\right]}{\max\left[V_u(c, V; X)\right] - \min\left[V_u(c, V; X)\right]}, \tag{10}$$

$$V_{oN}(c, V) = \frac{V_o(c, V) - \min\left[V_o(c, V)\right]}{\max\left[V_o(c, V)\right] - \min\left[V_o(c, V)\right]}. \tag{11}$$

Finally, the SV index is defined by

$$V_{SV}(c, V; X) = V_{uN}(c, V; X) + V_{oN}(c, V). \quad (12)$$

To assess the effectiveness of clustering algorithm, the smaller the SV index value, the better the performance.

2.7 Preliminary Results with Existing Indices

To evaluate the effectiveness of existing indices, we generate a two-dimensional, 2000-point, 9-cluster testing database called 'My_sample', illustrated in Fig. 1. All six indices are examined, and results are in Table 1.

With the database, we can expect that the column with $k = 9$ should perform the best, i.e., the largest PC value and the smallest values of the other five should be obtained. As we can see, not all of the indices indicate that the correct clustering result is when $k = 9$. Moreover, the criterion for PC is to search for its maximum value, while for the rest indices the criterion is to find their minimum values. Based on the two findings, the optimization techniques can be included into the clustering algorithm to search for the better and more correct results.

3. GENETIC-BASED CLUSTER VALIDITY INDEX

As we can see from Sec. 2.1 to 2.6, every index has its own specific concept for data clustering and the results in Sec. 2.7 have a diversity of performances. Therefore, we employ genetic algorithm (GA) for finding an optimized result based on the concept of every index above. GA constitutes of three major steps: *crossover*, *mutation*, and *selection*. Based on the fitness function, we try to integrate our watermarking scheme with GA procedures.

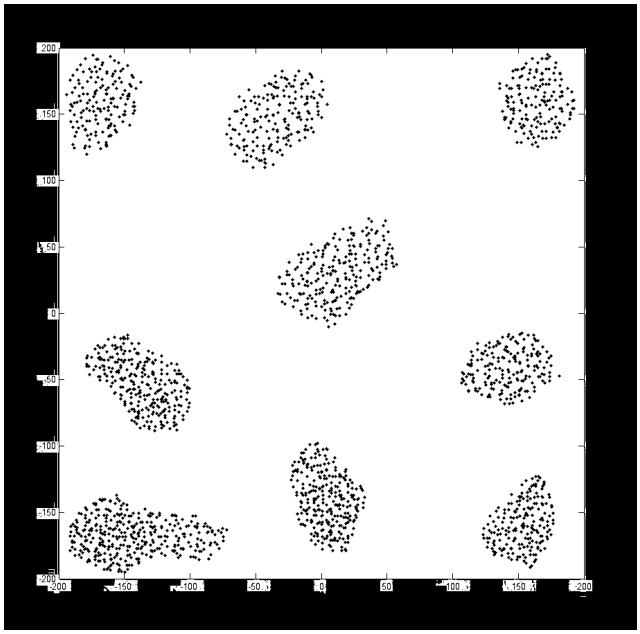

Figure 1: The Two-Dimensional, 2000-Point, 9-Cluster Database My_sample.

Table 1
The Index Values for Clustering from 2 to 10 Clusters in Six Different Schemes for My_sample Database. The Shaded Blocks Represent the Correct Clustering Results

index	*k=2*	*k=3*	*k=4*	*k=5*	*k=6*	*k=7*	*k=8*	*k=9*	*k=10*
PC	0.722	0.673	0.625	0.640	0.674	0.720	0.757	0.797	0.770
PE	0.631	0.852	1.051	1.071	1.030	0.936	0.855	0.754	0.834
XB	0.277	0.110	0.188	0.224	0.095	0.100	0.068	0.042	0.628
K	555	221	377	449	191	202	139	87	1300
B_{crit}	17.87	10.85	9.86	10.77	8.14	8.79	8.59	8.39	17.42
SV	1.000	0.691	0.619	0.485	0.334	0.312	0.261	0.220	1.000

3.1 Preprocessing in GA

We need to have chromosomes to perform the three steps in GA. We employ five popularly used databases, including auto-mpg [8], bupa [9], cmc [10], iris [11], and wine [12] in Table 2 for GA optimization. Half of the data set in each database is used for training, and the other half is used for testing.

3.2 Deciding the Fitness Function

After considering practical implementations in GA, and based on the indices described in Sec. 0 to 0, in this paper, we proposed the genetic-based index for data clustering. The fitness function is denoted by

$$V_{\text{gene}}(c, V; X) = \alpha \cdot \frac{\frac{1}{c}\sum_{k=1}^{c} \text{INTRA}(k)}{\text{MSD}_t} + \beta \cdot \frac{\max_{i,j} d(V_i, V_j)}{\min_{i \neq j} d(V_i, V_j)}. \quad (13)$$

In the first term, it denotes the compactness with

$$\text{INTRA}(k) = \frac{1}{n_k} \sum_{x \in x_k} \left\| \overline{V_t} - x_j \right\|, \text{ and} \quad (14)$$

$$\text{MSD}_t = \frac{1}{n_t} \sum_{j=1}^{n_t} \left\| \overline{V_t} - x_j \right\|. \quad (15)$$

In the second term, $d(V_i, V_j)$ is the same as that defined in Eq. (6). Also, α and β are the weighting factors, which act as the output after GA training.

The goal for optimization is to find the minimized value in the fitness function. Under the best condition, the fitness value reaches 0.

3.3 Procedures in GA Training

The GA procedures for optimized cluster validity index are described as follows.

Step 1: *Producing the chromosomes*: 40 chromosomes are produced. Each chromosome denotes the weighting factors in the fitness function, i.e., (α_i, β_i), $1 \le i \le 40$. Because the fitness function is composed of two opposing conditions, we only concern about the ratio between the two weights; we set $0 \le \alpha_i, \beta_i \le 1$.

Table 2
The Five Databases Used in This Paper

Training database	*# of data points*	*Testing database*	*# of data points*
auto-mpg_train	196	auto-mpg_test	196
bupa_train	173	bupa_test	172
cmc_train	737	cmc_test	736
iris_train	75	iris_test	75
wine_train	89	wine_test	89

Fitness values are calculated from the training databases in Table 2. At the beginning of first iteration, chromosome values are randomly set. In training, chromosome values are modified based on the output of the previous iteration.

Step 2: *Selecting the better chromosomes:* All the 40 sets of chromosomes are included into the fitness function and the corresponding fitness scores are calculated. The 20 chromosomes with smaller fitness values are kept for use in the next iteration, and the other 20 are discarded. 20 new chromosomes in the next iteration are produced from crossover and mutation based on the 20 chromosomes remained.

Step 3: *Crossover of chromosome:* From the 20 remained chromosomes, we randomly choose 10 of them, and gather into 5 pairs, to perform the crossover operation. By swapping the α or β values of every pair, 10 new chromosomes are produced.

Step 4: *Mutation of chromosome:* The 10 chromosomes that are not chosen in 0 are used in this step. The α values in the first five chromosomes are replaced by randomly set, new α values. Similar operation is performed on the β values of the other five.

Step 5: *The stopping condition:* Once the pre-determined number of iterations is reached, or when the fitness value equals 0, the training is stopped, and the weighting factors corresponding to the smallest fitness score in the final iteration, (α, β), is the output.

4. SIMULATION RESULTS

After training for 1000 iterations the GA optimization in Sec. 3.3, we obtain the optimized weighting factors (α, β) = (0.8561, 0.0826). With the two values, we can compare the GA optimized result with those in Sec. 2.1 to 2.6 by verifying the five test databases in Table 2. We depict the detailed results with the auto-mpg database in Table 3, the bupa database in Table 4, the iris database in

Table 5, the wine database in Table 6, respectively. Numerical values in Table 3 depict the results for the auto-mpg database, which has three clusters. We can see that only with the proposed GA-based index has the correct result. In bupa, cmc, iris, and wine databases, similar results can be obtained, and detailed comparisons can be found from Table 4 to Table 7, respectively. In addition, from Table 8, we see that the proposed GI results in correct cluster numbers in four of the five test databases. Comparing to other six indices that only result in one correct cluster number, our scheme gets better performance. In addition, regarding to the cmc database, none of the seven indices have the correct cluster number.

Table 3
Index Values from 2 to 10 Clusters in Seven Different Schemes. Shaded Blocks Show the Correct Results for Auto-mpg Database

index	*k=2*	*k=3*	*k=4*	*k=5*	*k=6*	*k=7*	*k=8*	*k=9*	*k=10*
PC	0.866	0.801	0.787	0.764	0.726	0.716	0.713	0.704	0.700
PE	0.330	0.522	0.596	0.679	0.804	0.847	0.869	0.915	0.944
XB	0.056	0.073	0.083	0.067	0.145	0.121	0.121	0.104	0.123
K	11.31	15.30	18.21	15.74	35.53	33.51	36.00	32.70	41.44
$\mathbf{B}_{crit}$	13.57	8.20	6.94	6.47	9.21	9.89	11.11	11.21	13.24
SV	1.000	0.633	0.466	0.415	0.548	0.592	0.705	0.771	1.000
GI	0.523	0.487	0.521	0.536	0.780	0.854	0.960	0.974	1.148

Table 4
Index Values from 2 to 10 Clusters in Seven Different Schemes. Shaded Blocks Show the Correct Results for Bupa Database

index	*k=2*	*k=3*	*k=4*	*k=5*	*k=6*	*k=7*	*k=8*	*k=9*	*k=10*
PC	0.882	0.664	0.562	0.476	0.411	0.383	0.346	0.328	0.295
PE	0.304	0.809	1.136	1.435	1.676	1.826	2.011	2.131	2.309
XB	0.065	0.511	0.587	0.623	1.480	1.307	1.073	1.395	1.407
K	11.64	94.16	110.2	118.5	284.2	256.1	212.8	271.9	282.5
$\mathbf{B}_{crit}$	59.03	46.69	45.75	47.62	67.55	83.79	49.41	56.06	63.48
SV	1.000	0.718	0.617	0.555	0.702	0.786	0.699	0.882	1.000
GI	1.088	1.225	1.286	1.328	1.754	1.787	1.690	1.903	1.981

5. CONCLUSION

In this paper, we discussed about data clustering schemes and proposed a new cluster validity index based on GA. GI index outperforms all the six existing indices in literature. However, clustering results for applications to some database are not correct even after GA training. And this is the motivation for our researches in the future.

ACKNOWLEDGMENTS

This work is partially supported by National Science Council (Taiwan) under grant NSC95-2218-E-005-034.

Table 5
Index Values from 2 to 10 Clusters in Seven Different Schemes. Shaded Blocks Show the Correct Results for cmc Database

index	*k=2*	*k=3*	*k=4*	*k=5*	*k=6*	*k=7*	*k=8*	*k=9*	*k=10*
PC	0.809	0.704	0.597	0.528	0.474	0.423	0.378	0.342	0.321
PE	0.459	0.773	1.089	1.323	1.523	1.723	1.905	2.066	2.189
XB	0.096	0.125	0.197	0.222	0.231	0.296	0.388	0.604	0.539
K	70.86	92.96	146.9	165.7	173.6	223.1	293.0	458.3	410.4
B_{crit}	18.57	13.26	11.96	13.35	13.37	17.26	16.77	19.17	22.55
SV	1.000	0.580	0.452	0.428	0.440	0.514	0.664	0.935	1.000
GI	0.617	0.595	0.660	0.721	0.771	0.866	0.990	1.214	1.206

Table 6
Index Values from 2 to 10 Clusters in Seven Different Schemes. Shaded Blocks Show the Correct Results for iris Database.

index	*k=2*	*k=3*	*k=4*	*k=5*	*k=6*	*k=7*	*k=8*	*k=9*	*k=10*
PC	0.888	0.790	0.738	0.678	0.610	0.584	0.562	0.538	0.535
PE	0.290	0.559	0.736	0.933	1.108	1.216	1.337	1.435	1.486
XB	0.058	0.115	0.160	0.265	0.316	0.549	0.239	0.227	0.289
K	4.622	9.920	14.72	25.25	33.21	61.43	26.73	28.05	36.45
B_{crit}	18.46	12.13	10.40	10.77	17.03	21.21	16.08	16.80	16.53
SV	1.000	0.724	0.598	0.628	0.695	0.907	0.700	0.832	1.000
GI	0.442	0.510	0.602	0.755	0.887	1.147	0.881	0.953	1.091

Table 7
Index Values from 2 to 10 Clusters in Seven Different Schemes. Shaded Blocks Show the Correct Results for Wine Database

index	*k=2*	*k=3*	*k=4*	*k=5*	*k=6*	*k=7*	*k=8*	*k=9*	*k=10*
PC	0.868	0.783	0.772	0.746	0.751	0.784	0.786	0.760	0.738
PE	0.328	0.572	0.636	0.720	0.738	0.663	0.677	0.764	0.830
XB	0.067	0.141	0.101	0.081	0.123	0.071	0.097	0.209	0.261
K	6.264	13.81	11.28	11.00	18.96	14.83	22.75	50.47	67.97
B_{crit}	22.85	14.17	11.64	9.169	11.01	10.06	12.82	19.18	22.89
SV	1.000	0.672	0.569	0.413	0.406	0.357	0.461	0.772	1.000
GI	0.570	0.566	0.605	0.641	0.841	0.828	1.061	1.594	1.896

Table 8
Comparisons of the Seven Indices for the Five Test Databases. Our Scheme Performs the Best

Database	*Original clusters*	*PC*	*PE*	*XB*	*K*	*B_{crit}*	*SV*	*GA*
auto-mpg	3	2	2	2	2	5	5	3
bupa	2	2	2	2	2	5	5	2
cmc	3	2	2	2	2	4	4	2
iris	3	2	2	2	2	4	5	3
wine	3	2	2	2	2	5	7	3

REFERENCES

[1] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning,* Boston, MA: Kluwer, 1989.

[2] J. C. Bezdeck, R. Ehrlich, and W. Full, "FCM: Fuzzy C-Means Algorithm", *Computers and Geosciences*, Vol. 10, No. 2-3, 1984, pp. 16-20.

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

[4] X. L. Xie and G. Beni, "A Validity Measure for Fuzzy Clustering", *IEEE Trans. Patt. Anal. Machine Intell.*, Vol. 13, No. 8, 1991, pp. 841-846.

[5] S. H. Kwon, "Cluster Validity Index for Fuzzy Clustering", *Electronics Letters*, Vol. 34, No. 22, pp. 2176-2177, 1998.

[6] A. O. Boudraa, "Dynamic Estimation of Number of Clusters in Data Sets", *Electronics Letters*, Vol. 35, No. 19, 1999, pp. 1606-1608.

[7] D. J. Kim, Y. W. Park, and D. J. Park, "A Novel Validity Index for Determination of the Optimal Number of Clusters", *IEICE. Trans. Inf. & Syst.*, Vol. E84-D, No. 2, 2001, pp. 281-285.

[8] R. Quinlan, "Auto-mpg data", *ftp://ftp.ics.uci.edu/pub/machine-learning-databases/auto-mpg/*, 1993.

[9] BUPA Medical Research Ltd, "BUPA Liver Disorders", *ftp://ftp.ics.uci.edu/pub/machine-learning-databases/liver-disorders/*, 1990.

[10] T. S. Lim, "Contraceptive method choice", *ftp://ftp.ics.uci.edu/pub/machine-learning-databases/cmc/*, 1999.

[11] R.A. Fisher, "Iris plants database", *ftp://ftp.ics.uci.edu/pub/machine-learning-databases/iris/*, 1988.

[12] S. Aeberhard, "Wine recognition data", *ftp://ftp.ics.uci.edu/pub/machine-learning-databases/wine/*, 1998.