Improving Clustering and Cluster Validation with Missing Data using Distance Estimation Methods

Marko Niemelä and Tommi Kärkkäinen

Abstract Missing data introduces a challenge in the field of unsupervised learning. In clustering, when the form and the number of clusters is to be determined, one needs to deal with the missing values both in the clustering process and in the cluster validation. In the previous research, the clustering algorithm has been treated using robust clustering methods and available data strategy, and the cluster validation indices have been computed with the partial distance approximation. However, lately special methods for distance estimation with missing values have been proposed and this work is the first one where these methods are systematically applied and tested in clustering and cluster validation. More precisely, we propose, implement, and analyze the use of distance estimation methods to improve the discrimination power of clustering and cluster validation indices. A novel, robust prototype-based clustering process in two stages is suggested. Our results and conclusions confirm the usefulness of the distance estimation methods in clustering but, surprisingly, not in cluster validation.

1 Introduction

The two main approaches for prototype-based clustering with missing values are imputation (Lin and Tsai [11]) and available data strategy. Combined with a statistically robust (see Kärkkäinen and Heikkola [9]) cluster prototypes like median or spatial median (Äyrämö [2]), the available data strategy

Marko Niemelä
University of Jyväskylä, Faculty of Information Technology, P.O. Box 35, FI-40014
University of Jyväskylä, Finland, e-mail: marko.p.niemela@jyu.fi

Tommi Kärkkäinen
University of Jyväskylä, Faculty of Information Technology, P.O. Box 35, FI-40014
University of Jyväskylä, Finland e-mail: tommi.karkkainen@jyu.fi

has proven to provide reliable results in a scalable fashion (Hämäläinen et al. [8]). However, in many applications the unsupervised tasks that need to be solved consist of estimation and determination of both the clusters and the number of them. The latter is addressed using cluster validation indices, which have been scarcely addressed with missing values although new techniques constantly emerge (Fu and Perry [5]).

As depicted in Hämäläinen et al. [7], Niemelä et al. [13], the cluster validation indices are composed of a quotient of estimates of *Inter* and *Intra* of a clustering result, i.e., the variability of data within clusters divided by the separation of clusters. Both of these measures are computed with a distance measure which is inhereted from the clustering problem formulation (Hämäläinen et al. [8]). Therefore, a key to reliable cluster validation indices with missing values is how to estimate the distances between the prototypes and the observations. For this purpose, in Niemelä et al. [13], the classical partial distance strategy (Gower [6]) was applied with promising results. However, more recently a set of papers have appeared (Eirola et al. [3, 4], Mesquita et al. [12]), which have addressed the distance estimation with missing values for both squared and euclidean (nonsquared) distances with better accuracy than in Gower [6].

This work continues the work in Niemelä et al. [13] by offering similar comparisons of cluster validation indices when the clustering method is replaced with the use of l_2-norm, i.e., optimized values of cluster prototypes minimize the Euclidean distance error with the target data instead using the squared Euclidean distance based error function (Äyrämö [2], Hämäläinen et al. [7]). Further, instead of the partial distance strategy, we utilize two previously presented distance estimation strategies (Eirola et al. [3], Mesquita et al. [12]) for calculating the distances between the possible incomplete data vectors during the cluster evaluation process. A novel, robust prototype-based clustering process in two stages is suggested when these strategies are applied in clustering. We then assess the usefulness of the distance estimation in cluster validation. As a whole, the purpose of this paper is to realize and test the distance estimation methods in an attempt to improve the reliability of clustering and cluster validation indices with missing values.

2 Methods

Prototype-based clustering methods, such as K-means, solve an optimization problem with K prototypes (Äyrämö [2], Hämäläinen et al. [7]). The objective function is defined to minimize the sum of the distances of the points to their closest prototypes. The prototype-based algorithm is composed of initialization and local improvement of the initial prototypes. This refinement is carried out in an iterative fashion by assigning individual observations to the closest prototypes and recomputation of the prototype with the assigned observations. These steps are repeated until the final converge is reached (Hämäläinen et al. [7]). The initial prototypes can be selected randomly but a more effective method is to use the K-means++ type of initial selection (Arthur and Vassilvitskii [1], Hämäläinen et al. [7]).

Spatial median is a statistically robust location estimate which can tolerate a large amount of missing values in data since it can handle up to 50 % of erroneous or missing components (Äyrämö [2]). The available data strategy (ADS) is a convenient way to omit the missing values during the cluster refinement phase. It is based on projecting all computations to the available values using a projection matrix $\mathbf{P}$, which represent the pattern of the available values similarly to Kärkkäinen and Toivanen [10]. This is obtained by setting $(\mathbf{P}_i)_j = 1$ if and only if the corresponding data component $(\mathbf{x}_i)_j$ exists, and zero otherwise. Using the available data strategy, the objective function for the spatial median based clustering can be written as follows:

$$\mathcal{J} = \sum_{k=1}^{K} \mathcal{J}_k = \underset{\{\mathbf{c}_k\}}{\arg\min} \sum_{\mathbf{x}_i \in \mathbf{C}_k} \|\mathbf{P}_i (\mathbf{x}_i - \mathbf{c}_k)\| , \tag{1}$$

where $\{\mathbf{x}_i\}_{i=1}^{N}$, $\mathbf{x}_i \in \mathbb{R}^n$, is the set of N observations with n-dimensions and $\{\mathbf{c}_k\}_{k=1}^{K}$ are the prototype vectors which are local minimizers of (1) defining the partition $\mathbf{C}_{k=1}^{K}$ of data into K disjoint subsets. We emphasize that the base of ADS, realized through the projection, lies in avoiding to introduce any additional assumptions on the data distribution.

In Eirola et al. [3], the expected squared Euclidean distance (ESD) estimation method for missing data was presented. The method assumes multivariate normally distributed data, which may be valid in many real world situations. Normality provides a rough approximation for nearly any continuous data distribution with relevant sample size, e.g., due to the central limit theorem (Rouaud [14]). In particular, it is assumed in Eirola et al. [3] that missing values in data vectors are random variables from the conditional normal distribution in which random variables are conditioned with the observed ones. In this case the incomplete parts of the vectors can be replaced with the conditional mean. If the missing components of $\mathbf{x}$ are denoted by $\mathbf{x}^{(1)}$ and the available components are denoted by $\mathbf{x}^{(2)}$ and n-dimensional incomplete multivariate data is partitioned as follows:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}^{(1)} \\ \mathbf{x}^{(2)} \end{bmatrix} , \tag{2}$$

then

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}^{(1)} \\ \boldsymbol{\mu}^{(2)} \end{bmatrix} , \quad \mathit{\Sigma} = \begin{bmatrix} \mathit{\Sigma}_{11} & \mathit{\Sigma}_{12} \\ \mathit{\Sigma}_{21} & \mathit{\Sigma}_{22} \end{bmatrix} ,$$

where $\boldsymbol{\mu}$ and $\mathit{\Sigma}$ denotes mean and covariance of $\mathbf{x}$. Further, conditional mean and variance for missing values can be expressed as follows:

$$\begin{aligned}\hat{\mathbf{x}}^{(1)} &= \boldsymbol{\mu}^{(1)} + \Sigma_{12}\Sigma_{22}^{-1}(\mathbf{x}^{(2)} - \boldsymbol{\mu}^{(2)}),\\ (\sigma^2)^{(1)} &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}.\end{aligned}$$

Notice that the multivariate normal parameters can be estimated for data sets even data cannot pre-partitioned as in (2). Thus, for conditional parameters, appropriate elements are required to be extracted from specific locations in $\boldsymbol{\mu}$ and Σ based on missingness pattern of individual observations.

It was proved in Eirola et al. [3] that the expected value for the squared Euclidean distance is the sum of the distance between the two estimated data vectors and the variances of the imputed components:

$$E[d_{il}^2] = E[||\mathbf{x}_i - \mathbf{x}_l||^2] = ||\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_l||^2 + \sigma_i^2 + \sigma_l^2,\ i \neq l,\ i, l \in [1, N].$$

A novel expected Euclidean distance (EED) method for estimating the nonsquared l_2-norm based distances with missing values was presented in a more recent study Mesquita et al. [12]. It uses the same basic principles as in Eirola et al. [3] for calculating the conditional distribution parameters. However, the EED is based on the assumption that the squared variables follow the Gamma distribution. This suggests use of the Nakagami distribution, where a random variable is obtained by taking the square root of a Gamma distributed variable. More precisely, the expected value of the Nakagami distribution can be written as

$$E[d_{il}] = \frac{\Gamma(m + \frac{1}{2})}{\Gamma(m)}\left(\frac{\Omega}{m}\right)^{\frac{1}{2}}, \tag{3}$$

where

$$m = \frac{E[d_{il}^2]^2}{Var[d_{il}^2]}, \quad \Omega = E[d_{il}^2].$$

Since the Nakagami distribution requires variances of distances, some extra calculations are needed. The variances can be calculated as follows (the details are given in Mesquita et al. [12]):

$$\mathrm{Var}[d_{il}^2] = E[\mathbf{x}_i^4 + \mathbf{x}_l^4 - 4\mathbf{x}_i^3\mathbf{x}_l - 4\mathbf{x}_i\mathbf{x}_l^3 + 6\mathbf{x}_i^2\mathbf{x}_l^2] - E[(\mathbf{x}_i - \mathbf{x}_l)^2]^2,$$

where the expected values can be obtained by using non-central moments of the normal distribution:

$$\begin{aligned}E[\mathbf{x}_i] &= \hat{\mathbf{x}}_i,\\ E[\mathbf{x}_i^2] &= \hat{\mathbf{x}}_i{}^2 + \sigma_i^2,\\ E[\mathbf{x}_i^3] &= \hat{\mathbf{x}}_i{}^3 + 3\hat{\mathbf{x}}_i\sigma_i^2,\\ E[\mathbf{x}_i^4] &= \hat{\mathbf{x}}_i{}^4 + 6\hat{\mathbf{x}}_i{}^2\sigma_i^2 + 3\sigma_i^4.\end{aligned}$$

Notice that we do not apply the weighted formulas in Mesquita et al. [12], because we assume similarly to ESD that the distributions are multivariate Gaussians instead of mixture of Gaussians.

Concerning cluster validation, we will apply the same cluster validation indices as in our previous study Niemelä et al. [13]. References to the original suggestions of the indices are given in Hämäläinen et al. [7], Niemelä et al. [13]. These read as follows (abbreviations given in parenthesis): `Calinski-Harabasz` (`CH`), `Davies-Bouldin` (`DB`), `Davies-Bouldin`* (`DB`*), `Generalized Dunn` (`GD`), `kCE-index` (`KCE`), `Pakhira-Bandyopadhyay-Maulik` (`PBM`), `Ray-Turi` (`RT`), `Silhouette` (`SIL`), `WB-index` (`WB`), and `Wemmert-Gançarski` (`WG`). Since clustering here is performed using the Euclidean distances (1), the indices were first implemented and preliminary tested by using the l_2-norm. We then noticed that `Calinski-Harabasz`, `kCE-index`, and `WB-index` obtained better results with their original forms of using the squared distances in the definitions of *Intra* and *Inter*. The reason might be that these indices include a scaling factor which was originally derived for the squared distances.

The formulas for the used indices are given in Table 1. There, $\mathbf{m}$ denotes the spatial median of the whole dataset. Moreover, the squared form $(\cdot)^2$ can also denote a componentwise application, for instance, within each cluster for $\mathcal{J}_k$ as in (1). We remind that the main focus of this work is that the distances both in clustering and in the CVIs afterwards can be computed with ADS, ESD, or EED, respectively.

In the `Silhouette` index, $\mathit{Intra}(\mathbf{x}_i)$ is the average Euclidean distance of the ith observation to all other points in the same cluster whereas $\mathit{Inter}(\mathbf{x}_i)$ is the average of the minimum distances of the ith point to points in a different cluster:

$$\mathit{Intra}(\mathbf{x}_i) = \frac{1}{n_k - 1} \sum_{\mathbf{x}_j \in \mathbf{C}_k} d(\mathbf{x}_i, \mathbf{x}_j), \ \mathit{Inter}(\mathbf{x}_i) = \min_{k \neq k'} \frac{1}{n_{k'}} \sum_{\mathbf{x}_j \in \mathbf{C}_{k'}} d(\mathbf{x}_i, \mathbf{x}_j). \tag{4}$$

Contrary to other indices in Table 1, in `Silhouette` one needs to calculate pairwise distances between the original, possible incomplete observations. Hence, the distance estimation techniques as presented above could be especially beneficial for the `Silhouette` index. On the other hand, because of the computations over each cluster and each observation within a cluster, the computational complexity is of the order $\mathcal{O}(N^2)$.

Notice that in Table 1 both *Intra* and *Inter* can be defined in three levels of abstraction concerning the clustering result: globally as, e.g., with `kCE-index`, clusterwise as, e.g., with `Davies-Bouldin`, and pointwise as, e.g., in `Silhouette`. This division is reflected in the actual Formula where no arguments is being given in the global case (*Intra* in `kCE-index`), arguments related to clusters are given in the clusterwise case ($\mathit{Intra}(k, k')$ in `Davies-Bouldin`), and index of an individual observation is given in the final case ($\mathit{Intra}(\mathbf{x}_i)$ in `Silhouette`), respectively.

Table 1 Formulas of cluster validation indices

Abbr	**Intra**	**Inter**	**Formula**
CH	$\mathcal{J}^2$	$\sum_{k=1}^{K} n_k d(\mathbf{c}_k, \mathbf{m})^2$	$\frac{K-1}{N-K} \times \frac{\mathit{Intra}}{\mathit{Inter}}$
DB	$\frac{\mathcal{J}_k}{n_k} + \frac{\mathcal{J}_{k'}}{n_{k'}}$	$d(\mathbf{c}_k, \mathbf{c}_{k^*})$	$\frac{1}{K} \sum_{k=1}^{K} \max_{k \neq k'} \frac{\mathit{Intra}(k,k')}{\mathit{Inter}(k,k')}$
DB*	$\frac{\mathcal{J}_k}{n_k} + \frac{\mathcal{J}_{k'}}{n_{k'}}$	$d(\mathbf{c}_k, \mathbf{c}_{k^*})$	$\frac{1}{K} \sum_{k=1}^{K} \frac{\max_{k \neq k'} \mathit{Intra}(k,k')}{\min_{k \neq k^*} \mathit{Inter}(k,k^*)}$
GD	$\max \frac{\mathcal{J}_k}{n_k}$	$\min_{k \neq k'} d(\mathbf{c}_k, \mathbf{c}_{k'})$	$\frac{2 \times \mathit{Intra}}{\mathit{Inter}}$
KCE	$\mathcal{J}^2$	1	$K \times \mathit{Intra}$
PBM	$\mathcal{J}$	$\sum_{i=1}^{N} d(\mathbf{x}_i, \mathbf{m}) \times \max_{k \neq k'} d(\mathbf{c}_k, \mathbf{c}_{k'})$	$\left(\frac{K \times \mathit{Intra}}{\mathit{Inter}}\right)^2$
RT	$\frac{1}{N}\mathcal{J}$	$\min_{k \neq k'} d(\mathbf{c}_k, \mathbf{c}_{k'})$	$\frac{\mathit{Intra}}{\mathit{Inter}}$
SIL	See text	See text	$\frac{1}{N} \sum_{i=1}^{N} \frac{\mathit{Inter}(\mathbf{x}_i) - \mathit{Intra}(\mathbf{x}_i)}{\max(\mathit{Intra}(\mathbf{x}_i), \mathit{Inter}(\mathbf{x}_i))}$
WB	$\mathcal{J}^2$	$\sum_{k=1}^{K} n_k d(\mathbf{c}_k, \mathbf{m})^2$	$K \times \frac{\mathit{Intra}}{\mathit{Inter}}$
WG	$d(\mathbf{x}_i, \mathbf{c}_k)$	$\min_{k \neq k'} d(\mathbf{x}_i, \mathbf{c}_{k'})$	$\sum_{k=1}^{K} \sum_{\mathbf{x}_i \in C_k} \frac{\mathit{Intra}(\mathbf{x}_i)}{\mathit{Inter}(\mathbf{x}_i)}$

3 Experiments and Results

Eight synthetic two dimensional data sets coinciding with our previous study were selected[1][2]. Experiment were performed using MATLAB (R2018b, 64-bit) and the same algorithm settings were used in clustering as in Niemelä et al. [13]: removing data components completely at random, discarding fully incomplete observations, minmax-scaling data to a range $[-1, 1]$, performing initialization in an iterative manner, using previously selected prototypes with K-means++ initialization algorithm, ranging K from 2 to 20, using 100 replicates in each clustering, and selecting final solutions as the lowest clustering error for the each value of K. Mean vectors and covariance matrices of incomplete multivariate normal data were estimated using `ecmnmle` method which was provided in MATLAB's Financial Toolbox.

Table 2 presents median calculation times and root mean square errors when clustering was performed with (EED, second row of results in each cell of Table 2) and without (ADS, first row of results in each cell of Table 2) distance estimation for all synthetic data sets. The clustering and missing

[1] `http://cs.uef.fi/sipu/datasets/`

[2] `http://users.jyu.fi/~mapeniem/CVI/Data/`

Table 2 The median calculation times and the obtained root mean square errors after repeated clustering. The numbers of observations are given in the brackets in the second row of the table.

ADS	**S1**	**S2**	**S3**	**S4**	**S2D2**	**S5D2**	**O200**	**O2000**
EED	(5000)	(5000)	(5000)	(5000)	(2000)	(2970)	(200)	(2000)
Time(s)$^{*+}$	12.670	15.520	21.030	23.270	1.090	4.090	1.470	5.030
	14.460	17.440	18.410	22.900	0.890	3.140	1.080	2.330
SD(s)$^{*+}$	2.100	2.300	2.920	3.060	0.140	0.670	0.120	0.610
	2.203	1.973	2.054	4.544	0.067	0.389	0.238	0.198
RMSE	0.005	0.006	0.013	0.013	0.049	0.073	0.054	0.034
	0.002	0.002	0.004	0.004	0.024	0.017	0.028	0.006

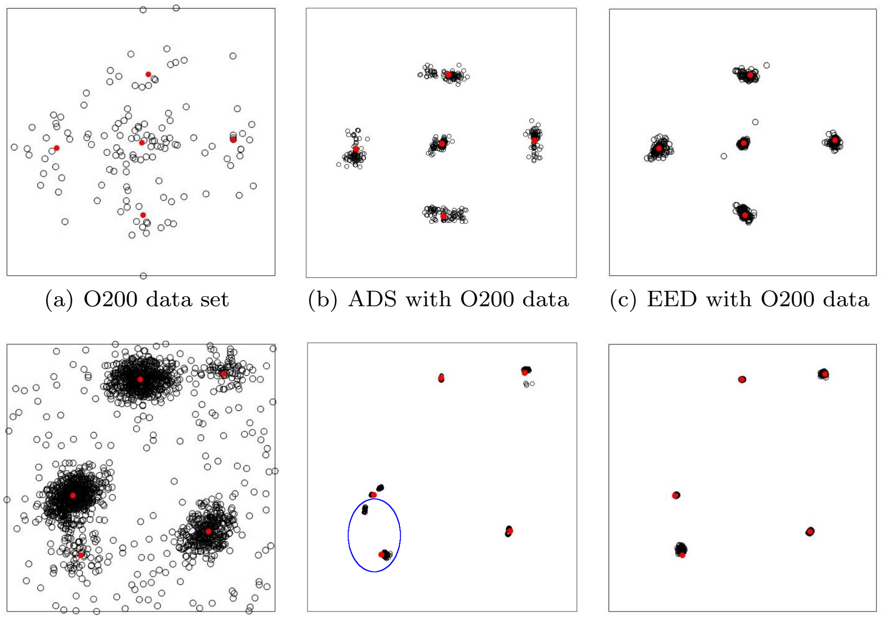
* By INTEL(R) XEON(R) CPU E5-2690 v4 @ 2.60GHz processor without parallelization

$^{+}$ Times were measured through 100 replicates in clustering

values generation were repeated 100 times using 20 % of missing values in the data. The correct numbers of clusters were used in every repetitions. The root mean square errors were calculated between the real centroids and the obtained clustering results. Regarding to the errors, the EED method provided better results with all data sets, especially with the *S5D2* and *O2000*. In addition, the EED showed almost the same computational complexity as the traditional ADS with the largest *S1–S4* data sets and to be faster with the rest of data sets.

Figure 1 shows clustering results through 100 repetitions for *O200* and *S5D2* data sets which consisted of 20 % re-generated missing values in each repetition. The obtained cluster prototypes are illustrated with the black circles. The original data centroids are visualized with the filled red circles. It can be seen from the figure that the variances of the clustered prototypes are smaller around the real prototypes when the EED distance estimation strategy was used. Further, Figure 1(e) shows some of the prototypes which were obtained with the ADS and should belong to the sparse bottom left cluster. However, these prototypes appeared to move towards to the dense cluster next to it. This is illustrated with an ellipse around prototypes.

The distance estimation strategy globally utilizes information on Gaussian distributed data while it makes decision of prototype locations and thus it appears to offer more stable results in the cases of sparse data sets. However, since the method is based on approximated quantities of the normal distributions, it can lead to nonoptimal solution locally, whereas the traditional ADS based clustering can be mathematically proofed to find a local minimum of an error function (Äyrämö [2]). This is the reason why we ended up using a two-stage clustering approach: distance estimation based clustering method first offers a high-quality initialization for the robust traditional method. The whole procedure is given in Algorithm 1. The new method was com-



(a) O200 data set (b) ADS with O200 data (c) EED with O200 data

(d) S5D2 data set (e) ADS with S5D2 data (f) EED with S5D2 data

Fig. 1 Clustering results of repeated clustering for two synthetic data sets using spatial median with and without distance estimation. The data sets consisted of 20 % missing values.

Algorithm 1 Spatial median clustering based on distance estimation

```
Input: Data set $\mathbf{X}_m$ with missing values and the number of clusters $K$
  Select initial prototypes in an iterative manner by using previously
    selected prototypes and K-means++ algorithm.
  Calculate a mean vector and a covariance matrix of the $\mathbf{X}_m$.
  repeat
     1. Estimate distances between observations and prototypes by Eq. (3).
     2. Assign individual observation to the closest prototype.
     3. Recompute prototypes with the assigned observations.
  until The final convergence
  Repeat steps 2 and 3 without distance estimation.
Output: $K$ partitions and prototypes of the given data set
```

pared against spatial median without distance estimation in the experiments related to the cluster validation.

Table 3 summarizes the results of the cluster validation indices. According to the table, the two-stage clustering approach notably improved the performance of most of the indices. Especially, the results improved in the cases of *O200* and *S5D2* data sets which were the most demanding for the indices. `Calinski-Harabasz` was the best performing index which always recommended the correct numbers of clusters with the new approach. The results of the Calinski-Harabasz were promising also without distance estimation since only in two out of 32 cases the index did not recommend the correct solutions. Other well performing indices were kCE-index, Ray-Turi, and Silhouette which recommended very often the correct numbers of clusters over all test cases.

The indices were implemented to use the ESD or EED distance estimation strategy. The strategy was selected based on the squared (ESD) or non-squared (EED) index formula (see Table 1). However, the distance estimation decreased the performance of most of the indices. Only Silhouette and Wemmert-Gançarski benefited from the estimation. Against other indices, Silhouette and Wemmert-Gançarski calculate *Inter* using distances between observations and their neighboring centroids or clusters (see Table 1 and Eq. (4)). Hence, distances were needed to be calculated more accurately for these two indices which is a good reason why the performance gain was obtained. Since distance estimation offered only marginal benefit with these two indices, we do not report results here.

4 Discussion

Let us briefly reflect the obtained results to the results of our previous study in Niemelä et al. [13]. The performance increased with most of the indices only by changing the clustering method to use the robust spatial median. The new estimation strategies yielded to performance gain. In eight over ten cases the results were at least equal, and in most of those (seven cases) better compared to the results of K-means with the partial distance strategy. However, Wemmert-Gançarski, which was the best performing index in Niemelä et al. [13], benefited the least from the current changes. Also, the results of Pakhira-Bandyopadhyay-Maulik were not improved, whereas especially Calinski-Harabasz and Ray-Turi were improved to recommend more often the correct number of clusters. The partial distance strategy was tested also in the current study but we noticed that the ADS performed better with the spatial median and, therefore, the results of the strategy were not reported here.

The new clustering method did not increase the computational complexity of the clustering. More specify, data vectors and variances were needed to be estimated only once for each observation which consisted missing values. This was done before the local refinement step of the prototype-based clustering (see Algorithm 1). Surprisingly, the calculation times were even smaller compared to the traditional spatial median clustering in cases of small data sets. However, all the data sets were only two dimensional and, hence, provided minimal challenge for the EED. In comparison, we tested the distance estimation through the whole clustering process such as estimations were repeated every time when cluster partitions were changed, i.e., as many times as final convergence was reached for each replicate of the clustering. As expected, this approach was computationally very intensive. Further, the performance of the indices did not improve as much that the method could be recommended to the clustering.

5 Conclusions

In this study, the internal cluster validation indices were compared to evaluate the number of clusters with data sets which included various ratios of missing values. The study differentiated from Niemelä et al. [13] by using similar experimental settings but extending the clustering method for more robust spatial median and utilizing the recently presented EED distance estimation strategy for clustering. The ESD and EED strategies were tested to implement to the actual indices. However, the most of the indices performed better without estimation. Thus, these results were not reported.

The study presented the new approach which performed clustering by using two stage clustering process where data sets were first clustered by using EED and, thereafter, the results were given as a starting point to the traditional ADS based spatial median clustering. On average, the new method improved the performance of the tested indices compared to the traditional ADS without distance estimation. Improved results were especially obtained when the data sets included 20 % of missing values. The best performing index was `Calinski-Harabasz`, which together distance estimation based clustering approach proposed always the correct number of clusters. The very promising results were also proposed by `kCE-index`, `Silhouette`, and `Ray-Turi` indices.

As it is well known, characteristics of real world data is rarely obvious. Therefore, it will be interesting to test the new method and the best indices with multiple of real world data sets. The special interest would be to measure the stability of indices against different ratios of missing values when the correct number of clusters is not clear.

Acknowledgements The authors would like to thank the Academy of Finland for the financial support (grants 311877 and 315550).

References

[1] D. Arthur and S. Vassilvitskii. K-means++: The advantages of careful seeding. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1027–1035. SIAM, 2007.

[2] S. Äyrämö. *Knowledge mining using robust clustering.* PhD thesis, University of Jyväskylä, 2006.

[3] E. Eirola, G. Doquire, M. Verleysen, and A. Lendasse. Distance estimation in numerical data sets with missing values. *Inform. Sci.*, 240: 115–128, 2013.

[4] E. Eirola, A. Lendasse, V. Vandewalle, and C. Biernacki. Mixture of Gaussians for distance estimation with missing data. *Neurocomput.*, 131:32–42, 2014.

[5] W. Fu and P. O Perry. Estimating the number of clusters using cross-validation. *J. Comput. Graph. Stat.*, 29(1):162–173, 2020.

[6] J. Gower. A general coefficient of similarity and some of its properties. *Biometrics*, 27(4):857–871, 1971.

[7] J. Hämäläinen, S. Jauhiainen, and T. Kärkkäinen. Comparison of internal clustering validation indices for prototype-based clustering. *Algorithms*, 10(3), 2017.

[8] J. Hämäläinen, T. Kärkkäinen, and T. Rossi. Scalable robust clustering method for large and sparse data. In *Proceedings of ESANN2018 – 26th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 449–454. ESANN, 2018.

[9] T. Kärkkäinen and E. Heikkola. Robust formulations for training multilayer perceptrons. *Neural Comput.*, 16(4):837–862, 2004.

[10] T. Kärkkäinen and J. Toivanen. Building blocks for odd-even multigrid with applications to reduced systems. *J. Comput. Appl. Math.*, 131(1–2): 15–33, 2001.

[11] W.-C. Lin and C.-F. Tsai. Missing value imputation: A review and analysis of the literature (2006–2017). *Artific. Intell. Rev.*, 53(2):1487–1509, 2020.

[12] D. P. P. Mesquita, J. P. P. Gomes, A. H. Souza Junior, and J. S. Nobre. Euclidean distance estimation in incomplete datasets. *Neurocomput.*, 248:11–18, 2017.

[13] M. Niemelä, S. Äyrämö, and T. Kärkkäinen. Comparison of cluster validation indices with missing data. In *Proceedings of ESANN2018 – 26th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pages 461–466. ESANN, 2018.

[14] M. Rouaud. *Probability, Statistics and Estimation: Propagation of Uncertainties in Experimental Measurement.* 2013.

Table 3 The determined number of clusters by internal cluster validation indices. The bolded numbers indicate correct solutions. Each column correspond different percentage (0, 5, 10, and 20 %) of missing values.

ADS **EED**$^{(**)}$	**CH**	**DB**	**DB***	**GD**	**KCE**
S1	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
	15 15 15 15	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
S2	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
	15 15 15 15	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
S3	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	4 **15 15 15**	**15 15 15 15**
	15 15 15 15	**15 15 15 15**	**15 15 15 15**	4 **15 15 15**	**15 15 15 15**
S4	**15 15 15 15**	17 17 17 **15**	13 13 13 13	4 3 3 4	**15 15 15 15**
	15 15 15 15	17 **15 15 15**	13 13 14 13	4 3 3 4	**15 15 15 15**
S2D2	**2 2 2 2**	**2 2 2 2**	**2 2 2 2**	**2 2 2 2**	**2 2 2 2**
	2 2 2 2	**2 2 2 2**	**2 2 2 2**	**2 2 2 2**	**2 2 2 2**
S5D2	**5 5 5** 4$^{(*)}$	3 3 3 3	3 3 3 3	3 3 3 3	**5 5 5** 4$^{(*)}$
	5 5 5 5	3 3 3 3	3 3 3 3	3 3 3 3	**5 5 5 5**
O200	**5 5 5** 20	**5 5 5** 20	**5 5 5** 20$^{(*)}$	4 4 **5 5**	**5 5** 20 20
	5 5 5 5	**5 5 5 5**	**5 5 5 5**	4 **5** 4 **5**	**5 5** 17 **5**
O2000	**5 5 5 5**	**5 5 5 5**	**5 5 5 5**	4 4 4 **5**	**5 5** 6 **5**
	5 5 5 5	**5 5 5 5**	**5 5 5 5**	4 4 **5** 4	**5 5** 6 6
Total	**8 8 8 6**	**6 6 6 6**	**6 6 6 5**	**3 4 5 6**	**8 8 6 6**
	8 8 8 8	**6 7 7 7**	**6 6 6 6**	**3 5 5 5**	**8 8 6 7**

ADS **EED**$^{(**)}$	**PBM**	**RT**	**SIL**	**WB**	**WG**
S1	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
	15 15 15 15	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
S2	**15 15 15 15**	**15 15 15** 14$^{(*)}$	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
	15 15 15 15	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**	**15 15 15 15**
S3	4 4 4 4	**15 15 15 15**	**15 15 15** 2	**15 15 15 15**	**15 15 15 15**
	4 4 4 4	**15 15 15 15**	**15 15 15** 2	**15 15 15 15**	**15 15 15 15**
S4	5 5 4 4	**15 15 15** 13	**15** 14 **15** 14	**15 15 15 15**	17 16 17 16
	5 5 5 4	**15 15 15** 14	**15 15 15 15**	**15 15 15 15**	17 16 16 **15**
S2D2	**2 2 2 2**	**2 2 2 2**	**2 2 2 2**	12 12 9 15	**2 2 2 2**
	2 2 2 2	**2 2 2 2**	**2 2 2 2**	12 12 8 9	**2 2 2 2**
S5D2	**5 5 5** 4$^{(*)}$	3 3 3 3	3 3 3 3	**5 5 5** 6$^{(*)}$	3 3 3 3
	5 5 5 5	3 3 3 3	3 3 3 3	**5 5 5 5**	3 3 3 3
O200	**5** 3 4 4	**5 5 5 5**	**5 5 5 5**	19 19 20 20	**5 5** 20 20
	5 3 4 3	**5 5** 4 **5**	**5 5 5 5**	19 19 17 20	**5 5** 20 20
O2000	3 4 4 4	**5 5 5 5**	**5 5 5 5**	**5 5 5 5**	**5 5 5 5**
	4 4 3 4	**5 5 5** 4	**5 5 5 5**	**5 5 5 5**	**5 5 5 5**
Total	**5 4 4 3**	**7 7 7 5**	**7 6 7 5**	**6 6 6 5**	**6 6 5 5**
	5 4 4 4	**7 7 6 5**	**7 7 7 6**	**6 6 6 6**	**6 6 5 6**

$^{(*)}$ Correct result was found using the known centers as initial prototypes
$^{(**)}$ Uses EED distance estimation in the first stage of clustering