Counterintuitive Behavior of Clustering Quality: Findings for K-Means on Synthetic and Real Data

Marco Loog[1(✉)], Jesse H. Krijthe[2], and Manuele Bicego[3]

[1] Radboud University, Nijmegen, The Netherlands
marco.loog@ru.nl
[2] Delft University of Technology, Delft, The Netherlands
j.h.krijthe@tudelft.nl
[3] University of Verona, Verona, Italy
manuele.bicego@univr.it

Abstract. Little is known about how the quality of a clustering changes when changing the size of the set used to determine the clustering model. We show that, for K-means clustering, the relationship between dataset size and clustering quality can display counterintuitive behavior. Notably, the quality can significantly *deteriorate* with *more data* to build the model. More generally, using artificial datasets and data from bioinformatics, we uncover a variety of learning curve behaviors for K-means. Our results clearly illustrate that the training sample size can have a nontrivial influence on the clustering performance. Our findings should appeal to both the clustering practitioner and the clustering researcher concerned with developing basic insights.

Keywords: Clustering quality · K-means · Counterintuitive behavior · Monotonicity · Gene ontology enrichment analysis

1 Introduction

Clustering [14,17] is a classic problem studied in machine learning, data mining, and abutting fields. One of its goals is to aggregate objects into meaningful groups so that objects in the same group are, in some sense, similar whereas objects from different groups are dissimilar. This is sometimes referred to as clustering's realist aim [15]. The goal can also be "constructive," aiming to split up data for more pragmatic reasons [15].

Clustering is ubiquitous in science and extensively applied for unsupervised exploratory data analysis in many scenarios. To roughly quantify how pervasive it is, we searched Scopus[1] and found around 39,000 papers published since 2020 with the word "clustering" in the title. Putting this in perspective: a similar

[1] Search performed in `www.scopus.com` in November 2024 with the search string `TITLE (''Clustering'') AND PUBYEAR > 2019 AND PUBYEAR < 2025.`

G. Krempl et al. (Eds.): IDA 2025, LNCS 15669, pp. 154–166, 2025.
https://doi.org/10.1007/978-3-031-91398-3_12

search for "CNN"[2], a popular machine learning method, returned about 25,000 hits. These papers come from more than 25 subject areas, ranging from computer to material science, medicine to the social sciences, and geology via psychology to the arts and humanities.

The foundational and theoretical aspects of clustering have generally received relatively little attention, compared to its widespread application. Existing work in this area investigates axiomatic approaches to clustering and issues of model selection and validation, but also addresses the question of clusterability and the aim and usage of clustering as such are considered (e.g., [1,15,20,46]).

This paper contributes to this foundational part of the field as well. However, it focuses on a matter entirely different from the above and asks the basic question: "Does clustering become better with more data?" This issue has received virtually no attention in the community, for none of the clustering techniques. Possibly this is due to the common belief that, if we avoid adversarial choices [9,49], *more data* should indeed imply *better cluster models*. We show that this belief is incorrect.

Until now, the only work to study this question is our 2023 contribution to ECML [24]. It shows that, when considering K-means clustering in the classical context of empirical risk minimization (ERM), performance can actually get worse with more data. That is, in terms of the minimum sum-of-squares (MSS or the within-group squared loss [24]) that K-means minimizes at training time, the average test loss can increase with more data. While this is an interesting theoretical finding, it merely shows that this behavior can occur in a specifically constructed probability distribution, and only in terms of the MSS, rather than more commonly accepted clustering quality measures. Here, we show such counterintuitive behaviors also occur for measures of quality that are broadly accepted within the clustering community. Moreover, we demonstrate this on both simulated and real-world bioinformatics data.

Our work identifies unknown and surprising behavior and, as such, furthers our fundamental understanding of K-means. However, our work is not theoretical. Due to the use of more commonly accepted performance measures and the employment of real data especially, we have to forego mathematical proofs and rely on empirical investigations. Our findings offer basic insight into K-means that should be of interest to both practitioners and researchers. To start with, Sect. 2 covers some additional related work and initial concepts used in the experiments. Section 3 then presents the datasets, experiments, and results. Finally, Sect. 4 offers a discussion and conclusion.

2 Further Background and Experimental Preliminaries

We cover directly relevant technical works in Subsect. 2.1. Subsection 2.2 then covers performance and quality measures and Subsect. 2.3 explains learning curves. These last two subsections cover some essentials for the experiments. First, however, we will specify the K-means method in more detail.

[2] Search string: `TITLE (''CNN'') AND PUBYEAR > 2019 AND PUBYEAR < 2025.`

Many methods have been proposed since the advent of clustering [16,18]. The K-means algorithm [22,37] is probably the most famous clustering approach. It belongs to the family of the so-called minimum sum-of-squares (MSS) clustering methods, where the clustering is found by minimizing the sum of the squared Euclidean distance of each object from the mean of the cluster to which it belongs. That is, given a set of K means $M = \{m_1, m_2, \ldots, m_K\}$ and N objects x_i, K-means aims to minimize

$$R_{\text{K-means}} = \frac{1}{N} \sum_{i=1}^{N} \min_{m \in M} \|x_i - m\|^2. \tag{1}$$

To find the best K means, the basic iterative procedure often used is to assign every data point to its nearest mean and then update the K means to the average of all points assigned to it. Typically, these two steps are repeated until convergence. To overcome the sensitivity of K-means to the initialization, this procedure is then run multiple times from a new random set of K means, and the solution with the minimum MSS is kept.

While this algorithm has known drawbacks [16,38] and many variations and alternatives have been presented [18,48], K-means is still widely applied for clustering.[3] Three factors contribute to its popularity: it is simple to implement, it is effective in many different contexts, and it is widely known—especially researchers from life sciences often prefer a thoroughly studied algorithm, with known limitations, over less studied alternatives.

2.1 Earlier Work Using Empirical Risk Minimization

Our earlier-mentioned work [24] shows that if one measures the clustering quality on a test set using the same measure as the one that is used during training, i.e., the minimum sum-of-squares (MSS), one can find unexpected behavior in which the expected loss can actually go up rather than down when the training set grows. This behavior is demonstrated to be structural, i.e., this deterioration of performance is in expectation over all possible training sets of a particular size, and therefore not due to an unlucky draw or other coincidence.

What is important in [24], and also in this work, is that training and test sets do not necessarily coincide. This may not be the most common setting in clustering, but it has its precedents. For instance, [7,35,45] consider K-means clustering in the context of the statistical learning theory and approach it in terms of classical empirical risk minimization (ERM) [42]. The result derived in [24] hinges on ERM as well. Where our work differs essentially is that we evaluate the performance of K-means in terms of commonly accepted and practically employed clustering quality measures. This situation is not unlike the classification setting where the error rate or accuracy is the de facto evaluation measure, but hardly any classifier optimizes these criteria directly [25].

[3] Just to get an idea of the numbers: Google Scholar finds about 37,000 articles with "K-means" in 2024 only.

We finally note that in recent years some similarly surprising findings have been reported for other learners, where these can start to perform worse with increasing numbers of training data [5,8,23,26,27,29]. Examples can be found for classification, regression, and density estimation. In the clustering case, however, a comprehensive analysis is still missing.

2.2 Performance and Quality of Clustering

Our study is concerned with the effect of dataset size on the quality of clustering [2,36,44,47]. Within clustering, as opposed to the supervised context, the definition of "the quality of a result" can be rather contentious. This is mainly due to the ground truth not being available and the initial problem being posed in a qualitative way.

In a seminal paper offering a related result pertaining to the definition of the quality of clustering, Kleinberg [20] showed that it is impossible to design a general-purpose clustering function obeying three simple and reasonable properties: scale-invariance, richness, and consistency. Subsequently, interesting alternatives for theoretical cluster validation have been proposed, such as in [4]. Generally, the problem of measuring the quality of a clustering result is still open, and, as suggested by [46], should probably be made dependent on the application under consideration.

This work presents results that go beyond the MSS loss in Eq. (1) that K-means intrinsically minimizes. The three widely accepted and commonly used clustering quality measures we consider are the adjusted Rand index (ARI), the normalized mutual information (NMI, type 1), and the purity. A range of variations on these measures exist [36], but our specific selection of these three indices can be considered representative.

Before providing their precise definitions, it should be noted that these measures rely on the availability of a true underlying clustering. This follows the well-known external validation strategy for clustering [17], which is widely employed, especially when developing novel clustering methods. For our artificial datasets, we will have to define ground truth labels ourselves. The ground truth used for our first real-world bioinformatics dataset is externally provided. Next to these, we also offer a completely different and, in a way, even more realistic evaluation through a gene ontology enrichment analysis for which the details follow in Subsect. 3.3.

Given N objects for which a clustering in K clusters is obtained, while the true labels have K' possible values. Let n_{ij} be the number of instances in cluster i that have true label j. In addition, let c_i be the size of cluster i and t_j be the number of points with true label j. The adjusted Rand index (ARI) then equals

$$\frac{\sum_{i=1}^{K}\sum_{j=1}^{K'}\binom{n_{ij}}{2} - \frac{\sum_i \binom{c_i}{2}\sum_j \binom{t_j}{2}}{\binom{N}{2}}}{\frac{1}{2}\left(\sum_{i=1}^{K}\binom{c_i}{2} + \sum_{j=1}^{K'}\binom{t_j}{2}\right) - \frac{\sum_i \binom{c_i}{2}\sum_j \binom{t_j}{2}}{\binom{N}{2}}}, \tag{2}$$

the (type 1) normalized mutual information (NMI) calculates

$$\frac{\sum_{i=1}^{K}\sum_{j=1}^{K'}\frac{n_{ij}}{N}\log\frac{Nn_{ij}}{c_i t_j}}{-\frac{1}{2}\left(\sum_{i=1}^{K}\frac{c_i}{N}\log\frac{c_i}{N}+\sum_{j=1}^{K'}\frac{t_j}{N}\log\frac{t_j}{N}\right)}, \tag{3}$$

and the third index we use is the purity, which is given by $\frac{1}{N}\sum_{i=1}^{K}\max_j n_{ij}$.

Note that ARI, NMI, and purity are quality measures, so larger is better, whereas for MSS, smaller is.

2.3 Learning Curves

Our analysis is based on learning curves. Such curves are widely used in supervised learning scenarios [32,43] and provide insight into the behavior of a given learner when varying the size of the set used to learn from. Within the clustering domain, such curves have hardly been utilized. A notable exception is [28] where both generalization and computational efficiency are plotted against the training set size for various clustering methods. We will use learning curves in combination with the clustering quality measures.

To generate our learning curves, we follow two different regimes. For the synthetic data, we know the underlying distribution, so we can sample training sets of a given size at random, train K-means on every draw, and then test its performance on a potentially large test set sampled from the same distribution. For the real data, performance is measured using the full datasets. Training sets of different sizes are drawn, without replacement, from this full dataset, so the training set size cannot exceed the number of instances in the full dataset.

It should be noted that, at test time, we measure the performance of the means obtained during training. First, all points in the test set are clustered using the training means, after which the quality of this clustering is determined using the measures introduced in the previous subsection.

3 Datasets, Experiments, Results

First, on the real-world data, we highlight some interesting learning curve behavior using the quality measures from Subsect. 2.2. Subsequently, we provide two synthetic datasets, for which we can understand, at least in part, why the learning curve behaves unexpectedly. We present the experiments and results in parts: every subsection introduces the specific data, presents the relevant learning curves, and makes provisional remarks on the results. Before that, let us present the last experimental details.

3.1 Experimental Parameter Settings

On the datasets in Subsects. 3.2 and 3.4, we repeat the procedure described in the previous section 1,000 times and report the mean and its standard deviation. We

consider 29 training set sizes for every dataset. The smallest size equals K, while the largest size S depends on the dataset: for the synthetic data, we consider 1,000; for the real-world data, we take the total set size as a maximum or, if the full size leads to excessive computation, we also settle for 1,000 as the maximum training set size. The other 27 sizes are taken linearly in between the extrema on a logarithmic scale and rounded to the nearest integer, i.e., the ith training set size is given by $\lfloor \log(e^K + \frac{i-1}{28}(e^S - e^K)) \rceil$. In Subsect. 3.3, we limit ourselves to 100 repetitions and a smaller number of training set sizes. Additional specifics are provided there.

Finally, where the theory in [24] assumes that we always find the globally optimal K-means for the particular training set at hand, we typically cannot guarantee this on the datasets we consider here. To get to acceptably good solutions, we use the widely used K-means++ algorithm [3] and take the best-performing solution over 100 random initializations from the training set. In the experiments of the first two subsections to follow, we consider two-class data for illustrative purposes and take $K = 2$.

3.2 Cancer Genome Atlas Data

We present results based on a dataset from The Cancer Genome Atlas (TCGA) with 4,434 specimens. The dataset has 19 cancer classes, but we consider cluster performance on subproblems comprising only two cancer types.[4] Originally, this dataset has 54,416 features made up of gene expressions, DNA-methylations, copy-number variations, and microRNA expressions. As suggested in [39], the dimensionality is reduced to 50 for every of these four feature subsets using PCA and then concatenated into a 200-dimensional final feature space. In bioinformatics, clustering is typically applied to datasets such as this. Reference [39] analyses this particular set with various techniques, among which K-means.

Figure 1 provides some interesting learning curves that show the surprising behavior these can display for K-means clustering. We picked these four to point out some particulars. Let us first note, however, that not every TCGA problem with two cancer subtypes leads to such nonmonotonic behavior. On the contrary, most of these learning curves show clustering improvement with more data to train on. However, the important point is that such non-monotonic learning curve behavior can at all happen and the community has no explanation for it.

First, on all four problems from Fig. 1, MSS behaves similarly monotonic, but the quality measures show quite some different behavior. The first row shows curves decreasing twice with increasing training set sizes. The second shows non-monotonic behavior for a clustering problem that K-means can solve perfectly, given enough data. The third row shows a drop in performance early on in the learning curve, after which its behavior becomes regular. This may not pose a problem practically, but it does challenge our understanding of K-means at a more fundamental level. The bottom row shows that, at least in for ARI and

[4] TCGA is kindly acknowledged for making this data available. The original data was generated by the TCGA Research Network: https://www.cancer.gov/tcga.

NMI, the performance loss can be rather dramatic. Granted, even their best performance may not be good, but K-means being a bad fit for the problem is not an a priori reason for it to lead to nonmonotonic learning.

3.3 Clustering Gene Expression Data

We now turn to another important clustering challenge involving gene expressions [11,19,33]. K-means is a typical strategy [13,40] and also industrially, it is routinely applied (for instance in the Affymetrix technology from Thermo Fisher Scientific, a prominent biotechnology company[5]). Where for TCGA the gene expressions are used as features to describe cancer types, in the current application every gene is described by its expression at different time points. Importantly, this clustering problem is one of the few real-world clustering problems where the quality can be assessed using a large collection of external information. This offers well-established procedures that we can use and complements quality measures like ARI, NMI, and purity. It offers an essentially different illustration of the counterintuitive clustering behavior this work exposes.

The specific procedure we employ is a gene ontology (GO) enrichment analysis [6]. The GO provides a computational representation of all available information on gene functions at molecular, cellular, and organism levels. This can be used to assess the coherence of a cluster of genes by checking if certain biological processes, molecular functions, or cellular components, as characterized by GO terms, appear more often than expected by chance in that cluster. The more clusters contain such an overrepresentation, the more successful the clustering.

We use the well-known `yeast cycle` dataset [10] describing the expression levels of 6,400 genes at seven different time points, i.e., the number of instances is 6,400 with a dimensionality of 7. As typically done in gene expression analysis [41], missing values were imputed using nearest neighbor imputation.

Now, determining the learning curve is somewhat different from the previous experiments. For a random draw of genes, we find the K-means solution and use the means to assign all 6,400 data points to the K clusters. The clustering quality is then measured by a GO enrichment analysis, following standard pipelines such as those from biclustering [12,30,34], in which only the largest gene clusters are evaluated, considering the smaller ones noise. In our case, we kept half of the clusters. A cluster is considered relevant if at least one GO term is overrepresented at a given significance level. The quality of the clustering is the fraction of evaluated clusters that are considered relevant.

The five significance levels, i.e., p-values, at which we perform the analysis are 0.05, 0.01, 0.005, 0.001, and 0.0001. We consider K-means with $K = 20$ and $K = 40$. Since the validation is very time-consuming,[6] we evaluated the learning curve for $K = 20$ over training set sizes 100, 200, 300, 500, 750, 1,000, 2,000,

[5] https://www.thermofisher.com/us/en/home/references/ambion-tech-support/transcriptome-analysis/tech-notes/analysis-of-microarray-data.html.

[6] The enrichment analysis is performed via the FuncAssociate web tool, available at http://llama.mshri.on.ca/funcassociate/, entailing a web call for every cluster at every level of significance.

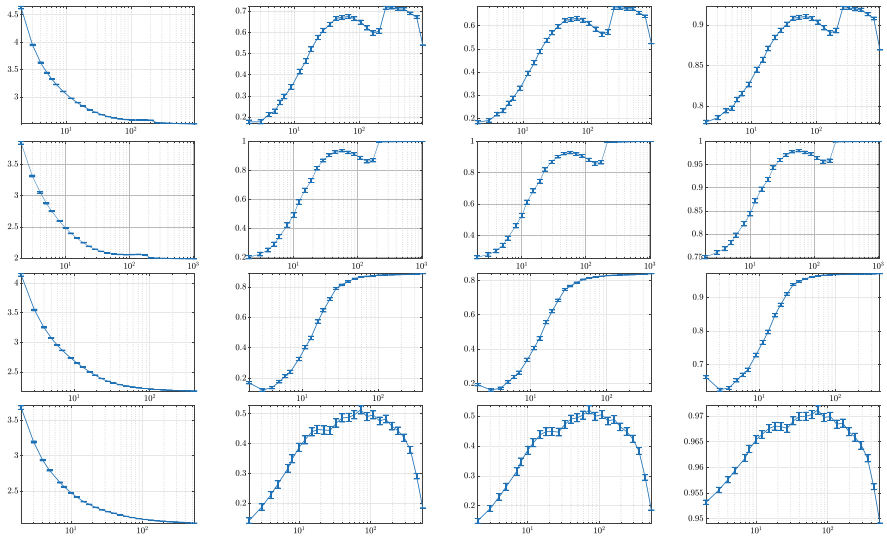

Fig. 1. Top to bottom, learning curves for a selection of two-class subproblems (4-letter abbreviations for the cancer types are from TCGA): `blca` vs `brca`, `brca` vs `prad`, `cesc` vs `lusc`, and `dlbc` vs `lusc`. Vertical axes: performance in terms of MSS, ARI, NMI, and purity, from left to right respectively. Horizontal axes are logarithmic and show the training set size used. Axes scalings may differ. The figure is best viewed electronically.

3,000, 4,000, and 6,400. For $K = 40$ we took 250, 500, 1,000, 2,000, 4,000, and 6,400 genes. We repeated the experiments 100 times.

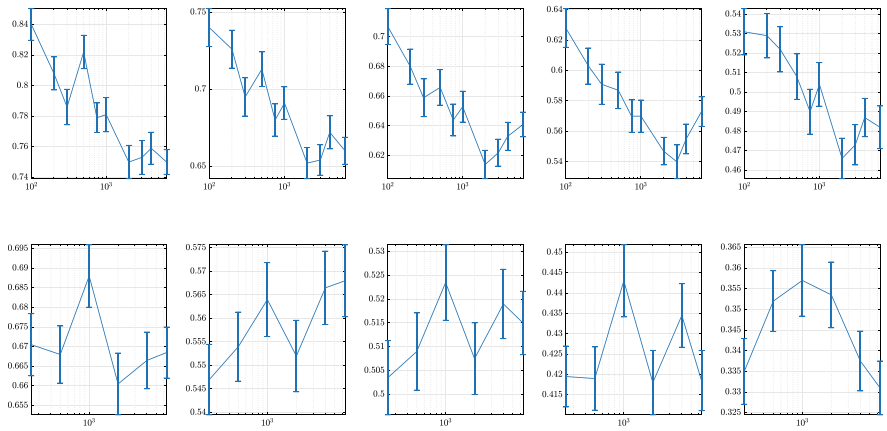

Fig. 2. Learning curves on `yeast cycle`. The top row is for $K = 20$, the bottom for $K = 40$. The columns contain the results for different p-values in the GO validation from left to right: 0.05, 0.01, 0.005, 0.001, and 0.0001. Vertical axes: performance in terms of the fraction of relevant clusters. Horizontal axes are logarithmic and show the training set size used. The figure is best viewed electronically.

The results in Fig. 2 further corroborate the general occurrence of counter-intuitive behavior. Especially when using $K = 20$ (top row), the best results (all statistically significant) are obtained with sample sizes considerably smaller than the complete dataset. For $K = 40$ (bottom row), only for the right two, the best performing training size is statistically significantly better (< 0.05) than using the full set; for the plot on the left we find $p = 0.059$.

3.4 Two Synthetic Datasets

In research, one way to try and understand more complex behavior is by designing toy models/examples that display some of the behavior we aim to understand [31]. In our final experiments, we consider two artificial datasets, where the true labeling follows directly from the generative process that defines the data.

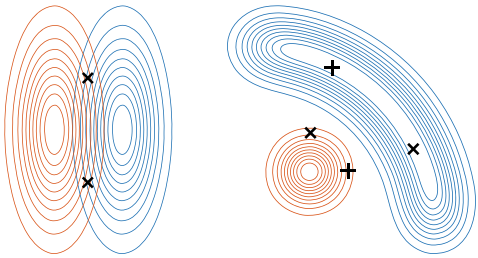

Fig. 3. Contour plots of the densities of the two artificial datasets leading to worsening learning behavior. The pairs of crosses indicate the globally optimal cluster centers. `Twogauss`, on the left, has one such optimum. For `oneeye`, on the right, there are two global optima: the two +s or the two ×s.

The first, referred to as `twogauss` and displayed on the left in Fig. 3, consists of two elongated Gaussian classes in two dimensions with means at $(-0.6, 0)$ and $(+0.6, 0)$ and equal covariance matrices $\begin{pmatrix} 0.16 & 0 \\ 0 & 1 \end{pmatrix}$. The class overlap is modest: the Bayes error is smaller than 0.07, and two well-placed means can attain this error. However, K-means will not find them because the variability in the vertical direction is too large. In the figure, the two black crosses indicate where the optimal means, in terms of the MSS objective, are located. These provide a solution that is completely at odds with the true classes. With more and more data, the K-means' solution gets closer to this suboptimum. As a result, the clustering quality becomes *worse* with increasing training set size (while the intrinsic MSS loss consistently improves). The top row in Fig. 4 provides the accompanying plots. This example shows an extreme case: the more data we train on, the worse the quality of clustering becomes.

The other dataset, referred to as `oneeye` and depicted on the right of Fig. 3, is a variation to mimic the possibility of, first, an improving quality measures,

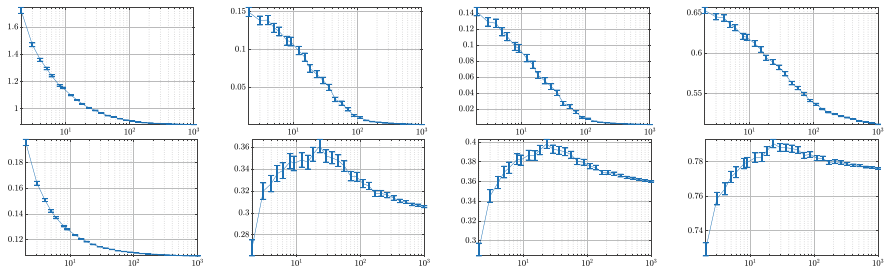

Fig. 4. Results on the `twogauss` (top) and `oneeye` (bottom) datasets. On the vertical axes is the performance in terms of MSS, ARI, NMI, and purity, from left to right respectively. The horizontal axes are logarithmic and show the training set size used, ranging from 2 to 1000.

followed by a deterioration. The one class is a Gaussian with mean $(0.3, 0, 3)$ and covariance $\frac{1}{81}I$. The other class is generated by first sampling a location uniformly at random from the arc in the top-right quadrant from the unit circle and then adding zero-mean Gaussian noise with the same covariance $\frac{1}{81}I$. The two global optima are given by the pair of pluses and the pair of crosses, respectively. With moderate amounts of training data, the two means we find are more often in between these two extremes, which leads the one mean to capture most of the red class and the other to capture most of the blue and therefore is better than the solutions that are globally optimal in terms of the MSS. The three bottom plots on the right in Fig. 4 show how the learning regime changes over increasing training set sizes. As for all other examples, the MSS shows regular monotonic behavior.

These two-dimensional examples, enable us to visualize how (initially) counterintuitive behavior may occur. However, this also makes it easy to see how K-means fails. One might then object that K-means should not be used in these settings. However, the point we want to make in the first place, is that it at all *can* happen that the performance is worse for a large training set than for a small one and that we understand how this could happen. Clearly, in real-world settings, one often cannot check any of this due to the absence of labels, or because of the high data dimensionality of the problem, etc. In those cases, one can only rely on the kind of insight such as the current work offers.

4 Discussion and Conclusion

The experiments provide the valuable insight that K-means clustering can show nontrivial behavior with, at times, deteriorating performance with an increasing amount of data used to build the clustering model. Qualitatively, the behavior seems rather similar between the different quality measures.

This information is relevant to the clustering practitioner because it is valuable to know that such behavior can at all happen and that the best performance

is not necessarily attained by using the complete dataset. The insight in itself should be of interest to the clustering researcher. Furthermore, our findings suggest new ways to broaden the applicability of K-means as such. For example, one of the major surveys on gene expression clustering, [19], claims that research has moved away from K-means as this method has relevant drawbacks. One of those drawbacks is the computational burden large dataset sizes cause. However, our work suggests that in certain settings one can have both less data and better performance.

By constructing examples using synthetic data, we have shed some light on the potential origins of this behavior. Of course, we do not think this is where it should stop. In fact, the real-world data seems to display nonmonotonic behavior that can be quite a bit more dramatic. Are we able to construct synthetic data that leads to curves similar to those in Fig. 1? Can we, more generally, identify every underlying mechanism that can lead to such behavior? Can we characterize it for some quality measures? Such a deeper understanding would also be of much use to the practitioner as it provides some control over the influence of the training set size.

A major open question is of course whether similar observations can be made for other clustering methods. It is possible to set up similar studies if the clustering method has a mechanism that permits assigning new points to the clusters identified in the training set. Even if, for many clustering approaches, there may be no natural way of accomplishing this, the class of methods for which it is possible is extensive: all K-means variants (including the so-called Generalized K-means models, in which models replace means), mixture models like Gaussian mixture models, and exemplar-based methods [21], to name a few. While we do not offer solutions, identifying nonmonotonic behavior is an important step toward a better understanding of clustering, both in theory and in practice.

Disclosure of Interests. The authors have no competing interests relevant to the content of this article.

References

1. Adolfsson, A., Ackerman, M., Brownstein, N.C.: To cluster, or not to cluster: an analysis of clusterability methods. Pattern Recogn. **88**, 13–26 (2019)
2. Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J.M., Perona, I.: An extensive comparative study of cluster validity indices. Patt. Recog. **46**(1), 243–256 (2013)
3. Arthur, D., Vassilvitskii, S.: k-means++ the advantages of careful seeding. In: Proceedings ACM-SIAM Symposium on Discrete Algorithms, pp. 1027–1035 (2007)
4. Ben-David, S., Ackerman, M.: Measures of clustering quality: a working set of axioms for clustering. NeurIPS **21** (2008)
5. Ben-David, S., Loker, D., Srebro, N., Sridharan, K.: Minimizing the misclassification error rate using a surrogate convex loss. In: ICML, pp. 83–90 (2012)
6. Berriz, G.F., King, O.D., Bryant, B., Sander, C., Roth, F.P.: Characterizing gene sets with funcassociate. Bioinformatics **19**(18), 2502–2504 (2003)
7. Buhmann, J.: Empirical risk approximation: an induction principle for unsupervised learning. Technical report AIA-TR-98-3, 3 (1998)

8. Chen, Z., Loog, M., Krijthe, J.H.: Explaining two strange learning curves. In: Benelux Conference on Artificial Intelligence, pp. 16–30. Springer (2022)
9. Crussell, J., Kegelmeyer, P.: Attacking dbscan for fun and profit. In: Proceedings SIAM International Conference on Data Mining (SDM), pp. 235–243 (2015)
10. DeRisi, J.L., Iyer, V.R., Brown, P.O.: Exploring the metabolic and genetic control of gene expression on a genomic scale. Science **278**(5338), 680–686 (1997)
11. D'haeseleer, P.: How does gene expression clustering work? Nat. Biotechnol. **23**(12), 1499–1501 (2005)
12. Eren, K., Deveci, M., Küçüktunç, O., et al.: A comparative analysis of biclustering algorithms for gene expression data. Brief. Bioinf. **14**(3), 279–292 (2013)
13. Gasch, A.P., Eisen, M.B.: Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. Genome Biol. **3**(11), 1–22 (2002)
14. Hartigan, J.: Clustering Algorithms. Wiley (1975)
15. Hennig, C.: What are the true clusters? PRL **64**, 53–62 (2015)
16. Jain, A.: Data clustering: 50 years beyond k-means. PRL **31**(8), 651–666 (2010)
17. Jain, A., Dubes, R.: Algorithms for clustering data. Prentice Hall (1988)
18. Jain, A., Murty, M., Flynn, P.: Data clustering: a review. ACM Comput. Surv. **31**(3), 264–323 (1999)
19. Jiang, D., Tang, C., Zhang, A.: Cluster analysis for gene expression data: a survey. IEEE Trans. Knowl. Data Eng. **16**(11), 1370–1386 (2004)
20. Kleinberg, J.: An impossibility theorem for clustering. NeurIPS **15** (2002)
21. Lashkari, D., Golland, P.: Convex clustering with exemplar-based models. NeuRIPS **20** (2007)
22. Lloyd, S.: Least squares quantization in PCM. Bell Laboratories (1957)
23. Loog, M., Duin, R.P.: The dipping phenomenon. In: Proceedings of S+SSPR, pp. 310–317. Springer (2012)
24. Loog, M., Krijthe, J.H., Bicego, M.: Also for k-means: more data does not imply better performance. Mach. Learn. **112**(8), 3033–3050 (2023)
25. Loog, M., Krijthe, J.H., Jensen, A.C.: On measuring and quantifying performance. In: Handbook of PRCV, pp. 53–68. World Scientific (2016)
26. Loog, M., Viering, T.: A survey of learning curves with bad behavior: or how more data need not lead to better performance. arXiv preprint arXiv:2211.14061 (2022)
27. Loog, M., Viering, T., Mey, A.: Minimizers of the empirical risk and risk monotonicity. NeurIPS **32**, 7478–7487 (2019)
28. Meek, C., Thiesson, B., Heckerman, D.: The learning-curve sampling method applied to model-based clustering. J. Mach. Learn. Res. **2**(Feb), 397–418 (2002)
29. Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., Sutskever, I.: Deep double descent: where bigger models and more data hurt. In: ICLR (2020)
30. Padilha, V.A., Campello, R.J.: A systematic comparative evaluation of biclustering techniques. BMC Bioinform. **18**(1), 1–25 (2017)
31. Páez, A.: Understanding with toy surrogate models in machine learning. Mind. Mach. **34**(4), 45 (2024)
32. Perlich, C.: Learning curves in machine learning. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning, pp. 577–488. Springer (2010)
33. Petegrosso, R., Li, Z., Kuang, R.: Machine learning and statistical methods for clustering single-cell RNA-sequencing data. Brief. Bioinf. **21**(4), 1209–1223 (2020)
34. Prelić, A., Bleuler, S., et al.: A systematic comparison and evaluation of biclustering methods for gene expression data. Bioinformatics **22**(9), 1122–1129 (2006)
35. Rakhlin, A., Caponnetto, A.: Stability of k-means clustering. NeurIPS **19** (2006)
36. Rezaei, M., Fränti, P.: Set matching measures for external cluster validity. IEEE Trans. Knowl. Data Eng. **28**(8), 2173–2186 (2016)

37. Steinhaus, H.: Sur la division des corps matériels en parties. Bulletin de l'Académie Polonaise des Sciences IV **12**, 801–804 (1956)
38. Steinley, D.: K-means clustering: a half-century synthesis. Br. J. Math. Stat. Psychol. **59**(1), 1–34 (2006)
39. Taskesen, E., Huisman, S.M., Mahfouz, A., et al.: Pan-cancer subtyping in a 2D-map shows substructures that are driven by specific combinations of molecular characteristics. Sci. Rep. **6**(1), 24949 (2016)
40. Tavazoie, S., Hughes, J.D., Campbell, M.J., Cho, R.J., Church, G.M.: Systematic determination of genetic network architecture. Nature Gen. **22**(3), 281–285 (1999)
41. Troyanskaya, O., Cantor, M., Sherlock, G., et al.: Missing value estimation methods for DNA microarrays. Bioinformatics **17**(6), 520–525 (2001)
42. Vapnik, V.: Estimation of Dependences Based on Empirical Data. Springer Series in Statistics. Springer (1982)
43. Viering, T., Loog, M.: The shape of learning curves: a review. TPAMI (2022)
44. Vinh, N.X., Epps, J., Bailey, J.: Information theoretic measures for clusterings comparison: is a correction for chance necessary? In: ICML, pp. 1073–1080 (2009)
45. Von Luxburg, U., Ben-David, S.: Towards a statistical theory of clustering. In: Pascal Workshop on Statistics and Optimization of Clustering, pp. 20–26 (2005)
46. Von Luxburg, U., Williamson, R.C., Guyon, I.: Clustering: science or art? In: Workshop on Unsupervised and Transfer Learning, pp. 65–79 (2012)
47. Wu, J., Xiong, H., Chen, J.: Adapting the right measures for k-means clustering. In: Proceedings of the ACM SIGKDD, pp. 877–886 (2009)
48. Xu, R., Wunsch, D.: Survey of clustering algorithms. TNN **16**(3), 645–678 (2005)
49. Yang, X., Deng, C., Wei, K., Yan, J., Liu, W.: Adversarial learning for robust deep clustering. In: NeurIPS, vol. 33, pp. 9098–9108 (2020)