Erschienen in: IEEE Computer Graphics and Applications (M-CG&A) ; 40 (2020), 2. - S. 98-102. - Institute of Electrical and Electronics Engineers (IEEE)
https://dx.doi.org/10.1109/MCG.2020.2970560

Teaching Clustering Algorithms With EduClust: Experience Report and Future Directions

Johannes Fuchs
University of Konstanz

Petra Isenberg
Université Paris-Saclay, CNRS, Inria, LRI

Anastasia Bezerianos
Université Paris-Saclay, CNRS, Inria, LRI

Matthias Miller
University of Konstanz

Daniel A. Keim
University of Konstanz

***Abstract*—We share our experiences teaching university students about clustering algorithms using *EduClust*, an online visualization we developed. *EduClust* supports professors in preparing teaching material and students in visually and interactively exploring cluster steps and the effects of changing clustering parameters. We used *EduClust* for two years in our computer science lectures on clustering algorithms and share our experience integrating the online application in a data science curriculum. We also point to opportunities for future development.**

We are currently seeing an immense increase in online learning platforms and sharing of teaching material.[1] We implemented *EduClust* (see Figure 2) to reduce the considerable effort in creating high-quality teaching material and to encourage learning in and outside the classroom. *EduClust* is an easily accessible online visualization application, which supports dynamic

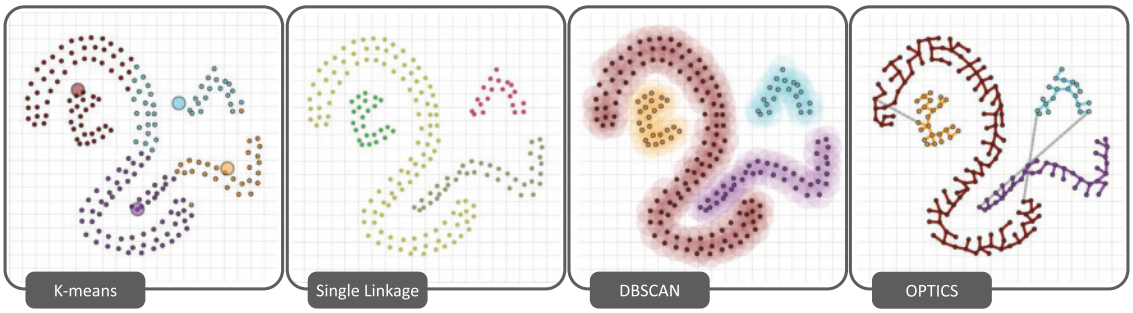

Figure 1. Application overview: Algorithms and datasets can be selected in the Selection Menu (top), parameters and animations can be adjusted in the Navigation Area and Parameter Settings (left), detailed text descriptions are displayed in the Information Space (right), and the clustering behavior is visualized in the Cluster View (center).

teaching and learning of clustering algorithms.[2] Simple two-dimensional data representations like scatterplots are used to show clustering behavior. We added animations to communicate changes between algorithmic steps. Different algorithms can be applied to various datasets and can be steered by changing input parameters or distance metrics. Additionally, further details about the algorithms are provided in a separate panel showing pseudocode, algorithmic complexity, and hyperparameters.

For two years, we used *EduClust* in our teaching routine. Based on our experiences with the software, we provide the interested reader with some guidance on preparing and organizing teaching material (e.g., slides and assignments) together with ideas about how to include the software in classroom settings (e.g., hands-on sessions with students). Given the positive feedback from our students using the software, we want to motivate similar development and research in this area.

TEACHING SCENARIOS WITH *EduClust*

EduClust is accessible online (educlust.dbvis.de/) and comes with nine different clustering implementations and an initial pool of datasets. Teachers and students can start right away using the software. In our data mining lecture at the University of Konstanz, we teach several different clustering algorithms. Our learning goals are based on Krathwohl's revised version of Bloom's educational objectives.[3] They comprise simple cognitive processes like *remembering* for which clustering algorithms exist and *understanding* the different categories the algorithms fall into, as well as the single steps of the algorithmic behavior. We want students to be able to *apply* the clustering algorithms to data in a meaningful way and *analyze* the influence of input parameters or distance measures on the clustering result. Ultimately we want students to *evaluate* (see Figure 1) the performance of clustering algorithms in certain situations and *create* scenarios, in which algorithms fail or outperform others. With *EduClust*, we were able to teach even complex cognitive processes and reduce the preparation time of the lecturer to a minimum.

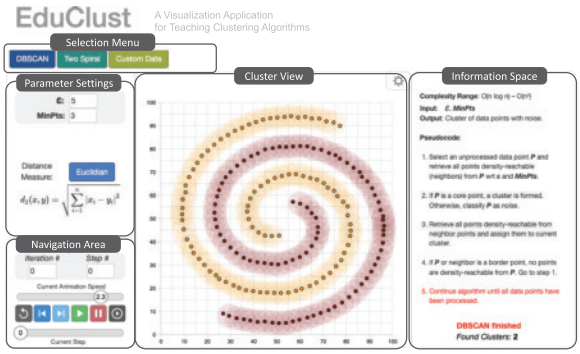

Figure 2. Four of the nine clustering algorithms and their visualization supported in EduClust: (a) *k*-means clustering centers shown as circles. Shapes in this artificial dataset are not separated well. (b) In Single Linkage, a dendrogram from the hierarchical clustering (not shown here) was used to determine the effective horizontal cut to differentiate each shape. (c) DBSCAN algorithm uses an ϵ-distance, which is represented using blurry circles. (d) Visualization shows the spanning tree of the OPTICS algorithm.

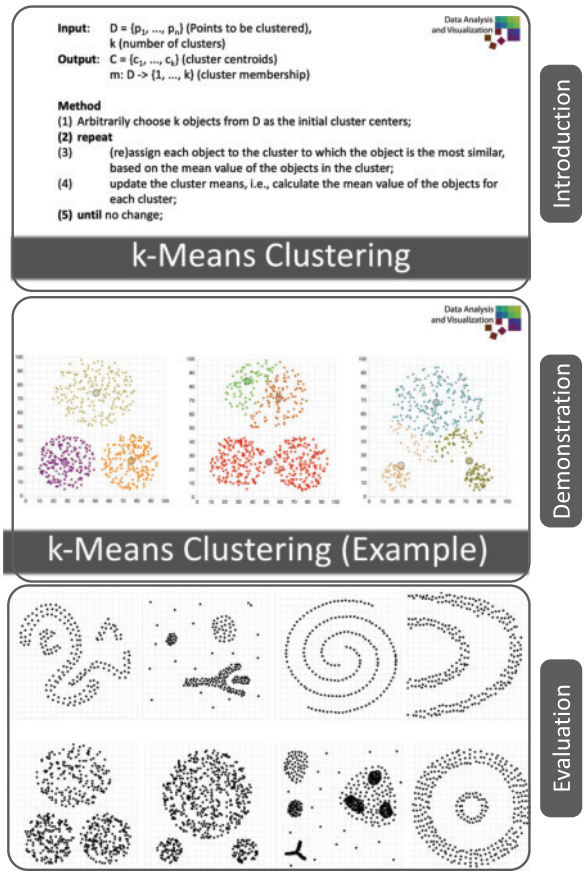

Figure 3. Our instructional material consists of three parts: first, introduction slides with pseudocode; second, a live demonstration of the clustering behavior (animations exported with EduClust); third, an evaluation of the clustering results using datasets with different characteristics (in-class exploration with EduClust).

Preparing Slides

The preparation of slides to show clustering steps can be a tedious task. To visually show changes over time, multiple intermediate steps of the algorithmic behavior need to be drawn out and displayed, ideally with animation.

With *EduClust*, one can use an export mechanism for individual clustering animations and the details about the algorithm provided in the information space. Lecturers just have to decide which algorithms, hyperparameters, and datasets they want to include in the slide deck. They run the software once and export the displayed animations in graphics interchange format to use in their slides. For details about the algorithmic behavior, complexity, or input parameters, *EduClust* provides ample text that can be copied on slides as well.

During the Lecture

To profit from *EduClust* during the lecture, we found it useful to split the session into three parts. First, a theoretical introduction to a new algorithm; next, a hands-on session; and finally a group discussion of advantages and disadvantages of different algorithms.

In our lectures, we always introduce new clustering algorithms showing slides with text information and pseudocode first, followed by a moderated animation generated by *EduClust* (see Figure 3). Students can see the algorithm in action, understand the individual clustering steps, and relate to the previously shown text descriptions. This first introduction is meant to support the cognitive processes *remember* and *understand*.

In the second part of the lecture, students use *EduClust* on their own to *apply* the algorithms to different datasets and *analyze* their peculiarity. Students, thus, experience the influence of changing input parameters and cluster characteristics. The duration of these individual hands-on sessions depends on the complexity of the algorithm.

In the third part, we put clustering algorithms into context with each other (see Figure 1). The lecturer starts a discussion by bringing up a dataset with specific characteristics. Students then discuss whether or not clustering algorithms are capable of separating data points into clusters. The lecturer and students use *EduClust* to try algorithms with various input parameters and discuss their advantages and disadvantages. Thereby, students *evaluate* the usefulness of algorithms and understand their individual application areas.

We found that this lecture structure covers nearly all cognitive processes to support student learning. However, we recommend to accompany the session with an assignment sheet to also support *create* as the another cognitive process.

Preparing an Assignment

Our assignments are designed to generate a deep engagement with specific clustering processes. We ask questions that require students to *apply* algorithms, *analyze* the consequences when changing input parameters, or *evaluate* different clustering techniques given a certain dataset. To further increase the learning rate, we also

include questions, in which students have to *create* datasets being suitable for the one algorithm but not for the others. In such scenarios, students have to understand details of the algorithms to come to a solution. Trial and error usually fails due to the complexity of the problem space with many different variables, e. g., input parameters, clustering algorithms, or distance metrics.

Student Assessment

EduClust supports the export and import of data files in the json format. This feature can facilitate the correction of submissions. When students have to *create* datasets for their assignment, they can export them and email their result to the lecturer. The lecturer can use *EduClust* to import the dataset and check for correctness.

Summary of Benefits

Although not exhaustive, *EduClust* covers the most prominent clustering algorithms and provides a visual categorization based on their clustering behavior. Lecturers can do live demonstrations of the clustering behavior of individual algorithms and use *EduClust* to prepare teaching material. *EduClust* offers datasets covering various cluster characteristics, which can be used together with all implemented clustering algorithms. The influence and importance of choosing input parameters wisely can be shown by running the same algorithm on the same dataset with varying input parameters. During the lecture, multiple clustering algorithms can be compared using the same dataset, revealing the benefits of each clustering algorithm. Both the description of algorithmic steps (in text), and a sequence of images showing these steps on a dataset, can be exported and added to traditional teaching material.

Students can apply clustering algorithms without implementation effort to various datasets and rerun the same algorithm multiple times using different input parameters. While running the algorithm, the underlying pseudocode is displayed in the information space. The selection of different algorithms and datasets help students to evaluate the performance of the respective algorithms. Finally, students can create individual datasets to be clustered with all implemented algorithms.

FUTURE RESEARCH DIRECTIONS

Qualitative evaluations showed that students are willing to use *EduClust* in their learning routine.[2] Currently, *EduClust* is limited to nine clustering algorithms, but we will extend it to include cluster quality measures and additional algorithms like DENCLUE.

Given the positive feedback from our students, we also see a lot of potential for applying what we learned to different categories of algorithms. A promising starting point could be decision trees. In addition, we would like to use *EduClust* as a motivation to establish a new research direction called teachable AI. While explainable AI gets a lot of research attention, respective applications focus on understanding the algorithmic behavior of individual architectures. We would like to argue for further research toward experiencing the entire inner workings of multiple algorithms together with the consequences of changing parameters and the possibility to evaluate different approaches on the same dataset, along the lines of explAIner[4] but with a focus on teaching ML algorithms from a professor and a student perspective.

REFERENCES

1. I. E. Allen and J. Seaman, *Changing Course: Ten Years of Tracking Online Education in the United States*. Babson Park, MA, USA: Babson Survey Research Group, 2013.
2. J. Fuchs, P. Isenberg, A. Bezerianos, M. Miller, and D. Keim, "Educlust-a visualization application for teaching clustering algorithms," in *Proc. Eurographics—Educ. Papers*, 2019. [Online]. Available: https://dx.doi.org/10.2312/eged.20191023
3. D. R. Krathwohl, "A revision of bloom's taxonomy: An overview," *Theory Into Practice*, vol. 41, no. 4, pp. 212–218, 2002.
4. T. Spinner, U. Schlegel, H. Schäfer, and M. El-Assady, "Explainer: A visual analytics framework for interactive and explainable machine learning," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 1, pp. 1064–1074, Jan. 2020.

Johannes Fuchs is a Research Scientist and Lecturer with the University of Konstanz, Konstanz, Germany. Contact him at fuchs@dbvis.inf.uni-konstanz.de.

Petra Isenberg is a Research Scientist with Inria, Rocquencourt, France in the Aviz team. Contact her at petra.isenberg@inria.fr.

Anastasia Bezerianos is an Associate Professor with University Paris-Saclay, Essonne, France, and part of the Inria ILDA team. Contact her at anastasia.bezerianos@lri.fr.

Matthias Miller is a Research Associate and is currently working toward the Ph.D. degree with the University of Konstanz, Konstanz, Germany. Contact him at miller@dbvis.inf.uni-konstanz.de.

Daniel A. Keim is a Full Professor and the Head of the Information Visualization and Data Analysis Research Group, University of Konstanz, Konstanz, Germany. Contact him at keim@uni-konstanz.de.

Contact department editors Beatriz Sousa Santos at bss@ua.pt and Ginger Alford at alfordg@smu.edu.