How can Small Data Sets be Clustered?

Anna Christina Weigand*
Faculty of Technology, University of Applied Sciences Emden/Leer
Emden, Germany
anna.ch.weigand@gmail.com

Daniel Lange*
Faculty of Technology, University of Applied Sciences Emden/Leer
Emden, Germany
daniel.lange@stud.hs-emden-leer.de

Maria Rauschenberger
Faculty of Technology, University of Applied Sciences Emden/Leer
Emden, Germany
maria.rauschenberger@hs-emden-leer.de

ABSTRACT

In many areas, only small data sets are available and big data does not play a significant role, *e.g.,* in Human-Centered Design research. In the context of machine learning analysis, results of small data sets can be biased due to single variables or missing values. Nevertheless, reliable and interpretable results are essential for determining further actions, such as, *e.g.,* treatments in a health-related use case. In this paper, we explore machine learning clustering algorithms on the basis of a small, health-related (variance) data set about early dyslexia screening. Therefore, we selected three different clustering algorithms from different clustering methods: K-Means, HAC and DBSCAN. In our case, K-Means and HAC showed promising results, while DBSCAN did not deliver distinct results. Based on our experiences, we provide first proposals on how to handle small data set clustering and describe situations in which using Human-Centered Design methods can increase interpretability of machine learning clustering results. Our work represents a starting point for discussing the topic of clustering small data sets.

CCS CONCEPTS

• **Theory of computation** → **Unsupervised learning and clustering**; • **Human-centered computing** → *Human computer interaction (HCI)*; • **Social and professional topics** → People with disabilities.

KEYWORDS

machine learning, human-centered design, interactive systems, health, small data, imbalanced data, variances, interpretable results, guidelines, data set, clustering

1 INTRODUCTION

The beginnings of machine learning (ML) focused on big data: a large amount of data that has both high velocity and high variety [4]. In web use cases, these databases usually contain, *e.g.,* millions or billions of data objects [7]. In some areas, big data does not yet exist or will never exist; therefore, investigating small data sets in the context of ML is of great value.

The interpretability of ML results is essential for understanding the outcome and for deducing appropriate further actions. It is especially important in health-related research areas, as wrongly interpreted ML results can have a significant negative impact on therapy approaches. For example, in the beginning of the COVID-19 pandemic, little data was available about the new disease, but many decisions were based on this data. Furthermore, in the case of COVID-19, the consequences do not only impact certain individuals, but instead affect the whole society. Additionally, health-related use cases deal with another complexity: multi-modal data can influence one result [9].

In connection with the *Human-Centered Design (HCD)* approach [6], small data sets often have around 200 or fewer data points, which are also called *tiny* data [16]. We use the terms *small* and *tiny* data sets as synonyms because tiny data has not yet been formally defined.

When only small data is available, results might be over-biased in comparison to big data as single variables or missing values have more impact in relation to the total number of values. This is one of many issues regarding the reliability of ML results when using small data sets. Hence, previous research already focused on ML prediction, or rather classification, of small data sets [1, 16, 17]. In one case, the authors used a data set that contained health-related data regarding early dyslexia screening [16, 17]. They derived recommendations for handling small data sets in ML prediction. The other research focused on ML prediction of student performance based on small data sets [1]. In advance, they used clustering to identify relevant features for the ML prediction. However, to the best of our knowledge, how to handle clustering with small data sets has not yet been explored.

In this paper, we investigate using ML clustering algorithms on small data sets. As a starting point, we explore one small, high variance and imbalanced data set, applying three different ML clustering algorithms. We used the open-access data set of early dyslexia screening [16] that was previously used for ML prediction. In the prediction use case, the aim was to predict whether or not a person has dyslexia. In our clustering use case, we aim to find relevant groups within the dyslexia data set, *i.e.,* preexisting conditions of persons with and without dyslexia. Our main contribution is a first proposal on how to handle small data in ML clustering and we explain as well how HCD methods might support the interpretability of results and the meaningful usage of distinct algorithms. Some of our proposals are also valid for medium or large data sets. This is a first step of exploring possible guidelines and obstacles for small data clustering and we do not claim completeness.

This paper is organized as follows: Section 2 describes the origin and types of ML clustering, while Section 3 describes related work.

[*] Both authors contributed equally to this research.

Mensch und Computer 2021, Workshopband, Workshop on User-Centered Artificial Intelligence (UCAI '21)

https://doi.org/10.18420/muc2021-mci-ws02-284

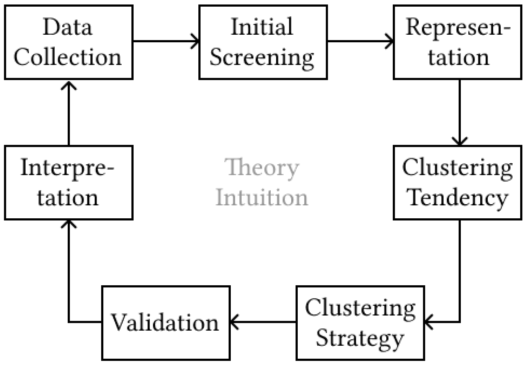

Figure 1: Iterative cluster analysis according to [11].

In Section 4, we show a small data use case in which we apply different clustering algorithms on a small data set. We then introduce first proposals for clustering small data sets in Section 5. Section 6 completes our work with the conclusion and future work.

2 BACKGROUND

ML clustering algorithms, also called *cluster analysis* or simply *clustering* [7], are used to identify relevant subsets within a data set. Furthermore, clustering relates to *unsupervised learning*, which means class labels are not available [7].

General steps and methods of ML clustering, as well as over- and under-fitting in this area, are described in the following subsections.

2.1 Clustering Steps

A well-known iterative approach describes seven steps for successfully exploring and clustering data sets [11]: data collection, initial screening, representation, clustering tendency, clustering strategy, validation, and interpretation (see Fig. 1). The authors indicate that it may be necessary to conduct some iterations to set the basis for a meaningful data set. The initial data screening is connected to data cleaning. The result of the representation step should be a proximity or pattern matrix. The representation depends, *e.g.,* on the use case, and the data. Afterwards, investigating the tendency of clustering prevents data from clustering, making it impossible to build appropriate clusters. Finding the right clustering strategy is a central point when a clustering method is chosen. The main goal is to fit the clustering algorithm and the data together, considering appropriate parameters as well as the method of data visualization. The data validation ensures robust indication of the clusters. At the end, the interpretation of the results is dependent on the investigator's knowledge of exploratory data analysis.

Still, finding the right clustering strategy can be challenging due to various factors [7]: scalability, ability to deal with different types of attributes, discovery of clusters with arbitrary shape, requirements for domain knowledge to determine input parameters, ability to deal with noisy data, incremental clustering and insensitivity to input order, capability of clustering high-dimensional data, constraint-based clustering, interpretability and usability.

These challenges directly or indirectly influence the choice of a suitable clustering method. An overview of the most common clustering methods is shown in the following section.

2.2 Clustering Methods

The fundamental clustering methods are divided into the following subgroups [7]: (1) *partitioning*, (2) *hierarchical*, (3) *density-based*, and (4) *grid-based*. The partitioning method (1) divides a data set with n objects into k partitions, an example algorithm is K-Means. Small to medium data sets can be used for partitioning methods [7]. The hierarchical methods (2) are categorized into agglomerative (bottom-up) and divisive (top-down) methods [7]. The bottom-up approach is based on having separate clusters for each object, which are then step by step grouped into one. The top-down approach starts with all objects in one cluster, which is then divided step by step into smaller clusters. The HAC (hierarchical agglomerative clustering) algorithm is an example in the context of hierarchical clustering methods. For grouping the clusters, the proximity between the clusters must be calculated. Therefore, different linkage metrics are used, *e.g.,* single linkage, complete linkage, average linkage or ward linkage [2]. The basic idea of density-based methods (3) is extending a cluster until a defined threshold of existing data points (density) around is met [7]. DBSCAN is an example algorithm for density-based clustering. With the grid-based clustering method (4), the object space is divided "*[...] into a finite number of cells that form a grid structure,*" with which the clustering is conducted [7]. Grid-based methods often have a short processing time and are used in combination with density-based or hierarchical methods.

After choosing a suitable clustering method for a specific use case, it is important to avoid issues such as over- or under-fitting and to consider the *Impossibility Theorem.* These topics are described in the following section.

2.3 Over-fitting, Under-fitting and the Impossiblity Theorem

Over-fitting and under-fitting are known concerns in the ML domain. Therefore, one approach for finding the optimal number of clusters k is, *e.g.,* the "*elbow method*" [20]. The "*elbow method*" is the optimal number of clusters in the *elbow* of the graph. This is measured by the total within-cluster sum of squares (WSS), which minimizes the intra-cluster variation [12]. Over-fitting occurs when one chooses a higher number of clusters than the *elbow*, while under-fitting occurs when one chooses a lower number of clusters.

Another phenomenon in ML clustering is the *Impossibility Theorem* [14]. This says that no more than two of the following points can be reached: richness, scale invariance and consistency. Richness, in this context, means that "*a clustering algorithm has the ability to create all types of mappings from data points to cluster assignments*" [13]. Scale invariance means that the algorithm is not sensitive to measurement changes [14]. The consistency of a clustering algorithm is explained as follows: after scaling down distances of data points inside one cluster and scaling up distances of data points between different clusters, the outcome is the same [14].

3 RELATED WORK

As described above, clustering has a long history but lacks distinct recommendations for clustering small data sets. Several clustering methods exist, along with their accompanying algorithms.

Recently, recommendations for small (high variance) data sets for predictive machine learning algorithms have been proposed [1, 17], but are not explicitly for clustering small (imbalanced/high variance) data.

One case used data gathered through a user experiment with an interactive system [17]. In this context, three different perspectives were combined: *Design Science Research Methodology (DSRM)* for setting up the corresponding experiment, *HCD* for designing the interactive system, and *data science* for applying the ML algorithms. The authors also explained that HCD can be an approach for resolving ML challenges.

Furthermore, prediction of student performance based on small data sets has been explored [1]. The authors found the key features using a hierarchical clustering algorithm. Afterwards, different ML algorithms were trained by these key features to evaluate the accuracy of their outcomes. The most promising results show the *support vector machine* as well as the *learning discriminant analysis* algorithms to be most effective.

Another relevant topic regarding machine learning algorithms and small health-related data sets is the *human-in-the-loop* approach [8]. The authors remark that the human-in-the-loop can be beneficial for solving ML (clustering) problems, *e.g.,* for health-related use cases a doctor-in-the-loop can provide relevant information for the ML algorithm to improve the outcome.

Based on current research, the following research gap has emerged: application of ML clustering algorithms to small data sets is missing because of the lack of information about clustering behavior in the context of small data sets. Hence, we consider the following open questions: *"What should be considered when clustering small data sets?" "Which clustering methods are useful?" "Are there any specific requirements regarding the data set?"*

Thus, our aim is to explore the application of clustering algorithms to a first small data set. Afterwards, we generate first proposals for handling small data set clustering.

4 SMALL DATA USE CASE

Among publicly available small data sets there are simulated data sets [10], which are not appropriate for our research approach. Hence, we explored one existing data set which contains health-related data collected with an interactive system in an online experiment. The aim of this already-collected data was to distinguish between children with and without dyslexia. The interactive system was designed with the HCD approach [6] and is well described in research papers [16–19].

4.1 Data Set Description

We used the data set consisting of 302 data points and 39 features [18] along with the provided description and the statistical analysis of the dependent variables that were collected with visual and auditory parts. The interactive system includes content related to visual and auditory indicators that have been successfully used in lab studies to distinguish between children with and without dyslexia.

An example for visual content is finding similar sketches and for auditory content is finding similar frequencies .

The data set has a high variance and is an imbalanced data set. For example, it includes about 37% dyslexic children and 63% non-dyslexic children for all languages. The data set has visual and auditory dependent variables.

4.2 Our Approach

We used some dependent variables from the visual part (*Total Clicks, First Click, Hits, Efficiency, Misses*) and some from the auditory part (*4th Click, Duration and Average*) as features for the different clustering algorithms. We selected these most-promising variables because they had been used in the statistical analysis and were found to be partly significant [18].

We aim to explore clustering algorithms for small data sets and discover obstacles in order to derive possible recommendations. We used the following clustering algorithms because they are among the most well-known, are easy to apply, and provide different clustering approaches: K-Means, DBSCAN and HAC.

First, we created a correlation matrix for the data set, but no surprising correlations were found. For example, there is a negative correlation of r = -.71 between *Accuracy* and *Misses*, but that is because the calculation of *Accuracy* depends on *Misses*. We then paired each of our features. Subsequently, the clustering algorithms were used on these two-dimensional feature pairs. Using more features would impair further analysis because the results would be expanded by additional complexities, resulting in lower interpretability.

4.3 Results and Discussion

Generally, we show the clustering results for the variables *First Click* (time taken until the first click happens) and *Total Clicks* (total number of clicks needed) in more detail. We chose these two features because one can understand them without further knowledge of the domain.

The StandardScaler from the Python module scikit-learn is used on the data (if not stated otherwise, we refer to the implementation of the scikit-learn library version 0.23.2 [15]). It scales the data to unit variance and thereby changes the x- and y-axes' units. Because of this, the range for *Total Clicks* is between *-3* and *6* and the range for *First Click* is between *-2* and *6*. The higher the number, the higher the time taken until the *First Click* happened or the higher the number of *Total Clicks* needed.

The results of K-Means and HAC are promising. DBSCAN shows poor results due to high variances in the data set.

First, we explore the K-Means algorithm (see Fig. 3). We use $k = 5$ as the number of clusters because the output of the *elbow* method was this optimal number of k (see Fig. 2). Besides the four distinct clusters, there is the light brown cluster that includes three data points (see Fig. 3) which seem to be outliers. The blue cluster has, on average, a number of *Total Clicks* under *0* and a long time before the *First Click* is made (over *1*). The light green cluster ranges from about *-2* to *1* for *Total Clicks* and from about *-1* to *2* for *First Click*. The fourth cluster (teal) ranges from about *-1* to *1* for *Total Clicks* and from about *-1* to *1* for *First Click*. The last cluster (grey) ranges from about *0.5* to *3* for *Total Clicks* and from about *-2* to *0* for *First Click*.

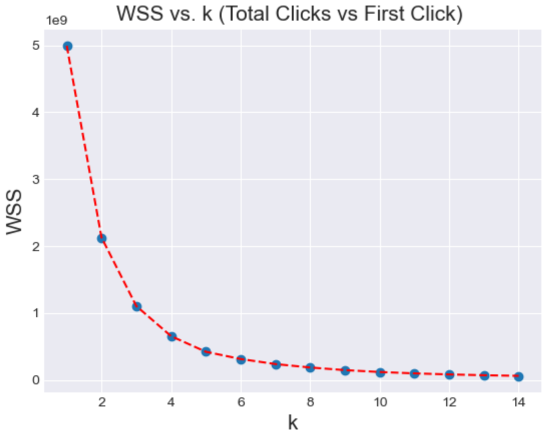

Figure 2: WSS vs. k (Total Clicks vs. First Click).

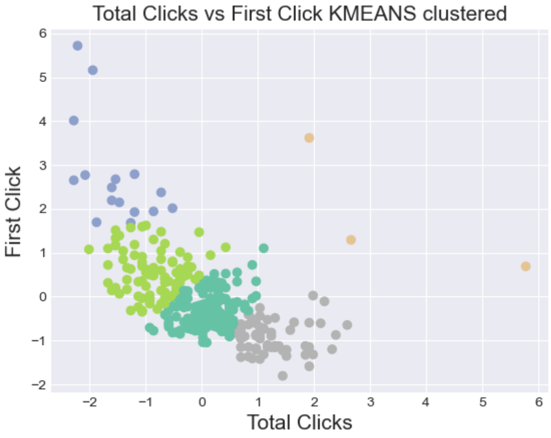

Figure 3: Total Clicks vs. First Click (K-Means clustered).

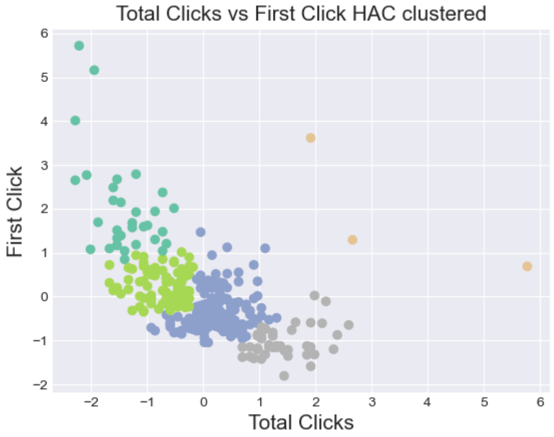

Figure 4: Total Clicks vs. First Click (HAC clustered).

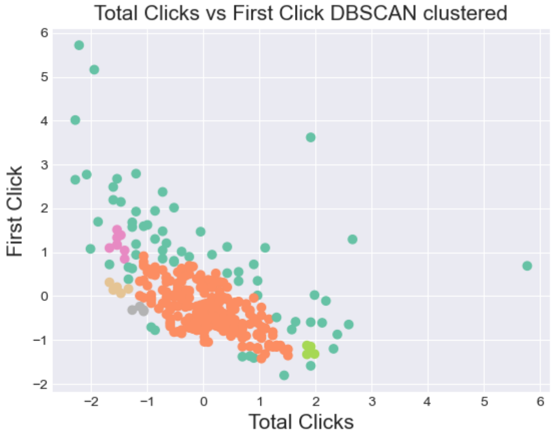

Figure 5: Total Clicks vs. First Click (DBSCAN clustered).

Next, we examine the HAC algorithm (see Fig. 4). To ensure comparability between HAC and K-Means, we again use $k = 5$ as the number of clusters. In the case of the HAC algorithm, a linkage metric can be set. We appropriate the *ward* metric for a similar approach as K-Means. In Figure 4, the light brown cluster of three data points seems to be entirely made up of outliers. In addition, another four clusters are shown. For the teal cluster, the number of *Total Clicks* is under *0* and the time to *First Click* is over *1*. The green cluster in Figure 4 ranges from about *-2* to *0* for *Total Clicks* and from about *-1* to *1* for *First Click*. The fourth cluster is the blue cluster. It ranges from about *-1* to *1* for *Total Clicks* and from about *-1* to *1.5* for *First Click*. The cluster in grey ranges for *Total Click* from about *0.5* to *3* and for *First Click* from about *-2* to *0*.

Third, we analyze the DBSCAN algorithm (see Fig. 5). The number of clusters is defined automatically by the algorithm ($k = 6$). As shown in Figure 5, the clusters are not clearly distinguishable.

As depicted in Figures 3 and 4, the results, the clustering, and the outlier partitioning for the K-Means and the HAC algorithms are very similar. For us, the outcome seems to be plausible in these cases, as the algorithms can deal with this small, high variance and imbalanced data set in a proper way. The five clusters are clearly identifiable and obviously both algorithms deliver similar results.

In contrast, the DBSCAN algorithm shown in Figure 5 does not deliver a meaningful outcome for this small data set. Here, the clusters are not clearly distinguishable from one another. The variances of this small data set influence the clustering outcome so that the results are not beneficial.

5 PROPOSALS FOR SMALL DATA SETS

Based on both our experience during the use case exploration and the existing literature, we aim to find answers to our initial questions: *“What should be considered when clustering small data sets?”*

"Which clustering methods are useful?" "Are there any specific requirements regarding the data set?" Therefore, we reveal the following first proposals for handling small data set clustering:

Data set description Domain knowledge is important for understanding the data set, especially in the HCD context because HCD experiment data is rarely self-explanatory and misinterpretations can have unintended consequences. Current data sets are often not well described; therefore, we recommend offering a specific and thorough description in an extra file. Some important descriptive aspects are: *"How was the data collected?" "What is the meaning of each attribute?" "Is the data already normalized or encoded?" "Is missing data already marked or predicted?", "Were data points with missing data deleted?" "Could there be some kind of bias in the selection of the data points?"* For HCD-related small data sets, it is highly important that one use the existing data as efficiently as possible, since the data set cannot easily be expanded. Repeating an HCD experiment or study appropriately means conducting it under the same test conditions and with the same investigators.

Clustering results Even for interpreting the clustering results, domain knowledge is often necessary. This is particularly true for small data in the case of HCD experiments, as the use cases can be very specific. In this case, HCD might be a supporting tool for evaluating the clustering outcomes together with domain experts. We recommend bringing ML and domain experts together to (iteratively) review the clustering algorithms' results, as was proposed by [21]. Adapting the whole clustering strategy can also be part of these reviews if necessary.

Feature selection Setting up hypotheses can help in selecting the features for useful results, as small data sets often have dependencies by chance [5]. For prediction (supervised learning), the entropy of the feature has an impact on the quality of the prediction results [16]. For our use-case clustering (unsupervised learning), we choose a few features to reduce the resource cost, starting with the most promising.

Variances If the data shows high variances in the small data set, we recommend performing a plausibility check to ensure the results are interpretable. For our use case, the results of the DBSCAN clustering were not distinct. In this case, it might be helpful to utilize an iterative HCD approach for result evaluation with domain experts. The goal is to check the algorithms' results for plausibility.

During our work, we searched for further open-source small data sets to validate the results of the first use case with additional data. Finding a suitable data set was quite difficult, which brings us to further proposals regarding small data sets:

Data types and algorithms As, *e.g.,* K-Means clustering is only suited for numerical data, the algorithm is not applicable for categorical data. When we have small data sets we recommend choosing an algorithm that is suitable for as many features as possible. The aim is to avoid losing information and to make the clustering results as comparable as possible. In the case of K-Means, *e.g.,* K-Modes could be a more suitable algorithm for categorical data. Moreover, HAC can be used for categorical data with the right distance metric, which is also valid for DBSCAN. However, as described in the previous section, in cases of high variances, the DBSCAN output implicitly needs to be checked for plausibilty.

Missing data Handling missing data depends on the amount of missing data and the total number of data points. If missing data occurs only a few times for a feature, we recommend removing these data points from the data set. If the data is frequently missing for a feature, we recommend either removing the whole feature or applying a ML regression algorithm to predict the missing data. As for prediction, missing or incorrect data can be imputed [16] if the data does not have high variances. Otherwise, noise would be added to the data. In the context of small data sets, missing data should be handled especially carefully because data is rare and reducing or imputing data can have a strong influence on the results.

Multiple subjects If the data set consists of multiple subjects with multiple measurement points (*e.g.,* interview data), we recommend representing each subject with one data point and having separate attributes for each of the different measurement points. This helps one to distinguish between small data sets and big data sets. Furthermore, analysis is then much more simple.

Verification Real small data sets are rare and often not distinguishable at first glance from an imitated test data set. To ensure the validity of publicly available data sets, we recommend providing an up-to-date verification reference to the related data sets. This is also valid for medium and large data sets.

License When using publicly available data sets, the following questions arise: *"Is the data free to use?" "Should credits be given when publishing work based on the data?" and "Is one allowed to publish work based on the data?"* We recommend verifying the license, such as *Creative Commons* [3], in a very early project stage so that regulations are observed accordingly.

6 CONCLUSION AND FUTURE WORK

In this work, we present first proposals for small data clustering. We also point out that HCD supports better interpretability of small data clustering results. Our proposals for clustering small data sets are a first proposition in this research area.

Future work is necessary to validate our first proposals for small data clustering. Furthermore, investigation using other small data sets is needed to develop general recommendations on this topic. In addition, more clustering algorithms and approaches need to be explored for small data sets (*e.g.,* OPTICS, Affinity Propagation, K-Modes).

REFERENCES

[1] Lubna Mahmoud Abu Zohair. 2019. Prediction of Student's performance by modelling small dataset size. *International Journal of Educational Technology in Higher Education* 16, 1 (dec 2019), 27. https://doi.org/10.1186/s41239-019-0160-3

[2] Lahbib Ajallouda, Fatima Zahra Fagroud, Ahmed Zellou, and El Habib Benlahmar. 2020. K-means, HAC and FCM Which Clustering Approach for Arabic Text?. In *Proceedings of the 13th International Conference on Intelligent Systems: Theories and*

Applications. ACM, New York, NY, USA, 1–8. https://doi.org/10.1145/3419604.3419779

[3] Creative Commons Corporation. 2021. Creative Commons - About the Licences. https://creativecommons.org/licenses/?lang=en. [Online; accessed 30-May-2021].

[4] Andrea De Mauro, Marco Greco, and Michele Grimaldi. 2015. What is big data? A consensual definition and a review of key research topics. *AIP Conference Proceedings* 1644, 1 (2015), 97–104. https://doi.org/10.1063/1.4907823 arXiv:https://aip.scitation.org/doi/pdf/10.1063/1.4907823

[5] Andy P. Field and Graham Hole. 2003. *How to design and report experiments*. SAGE Publications, London. 384 pages.

[6] DIN Deutsches Institut für Normung e. V. 2020. *DIN EN ISO 9241-210:2020-03, Ergonomie der Mensch-System-Interaktion - Teil 210: Menschzentrierte Gestaltung interaktiver Systeme; Deutsche Fassung*. Technical Report. Beuth Verlag GmbH. https://doi.org/10.31030/3104744

[7] Jiawei Han, Micheline Kamber, and Jian Pei. 2012. 10 - Cluster Analysis: Basic Concepts and Methods. In *Data Mining (Third Edition)* (third edition ed.), Jiawei Han, Micheline Kamber, and Jian Pei (Eds.). Morgan Kaufmann, Boston, 443–495. https://doi.org/10.1016/B978-0-12-381479-1.00010-1

[8] Andreas Holzinger. 2016. Interactive machine learning for health informatics: when do we need the human-in-the-loop? *Brain Informatics* 3, 2 (jun 2016), 119–131. https://doi.org/10.1007/s40708-016-0042-6

[9] Andreas Holzinger, Bernd Malle, Anna Saranti, and Bastian Pfeifer. 2021. Towards multi-modal causability with Graph Neural Networks enabling information fusion for explainable AI. *Information Fusion* 71, October 2020 (jul 2021), 28–37. https://doi.org/10.1016/j.inffus.2021.01.008

[10] Kandi Jagadish. 2019. Mall customers dataset. https://www.kaggle.com/kandij/mall-customers. [Online; accessed 30-May-2021].

[11] Anil K. Jain and Richard C. Dubes. 1988. Algorithms for Clustering Data. , 320 pages. https://homepages.inf.ed.ac.uk/rbf/BOOKS/JAIN/Clustering_Jain_Dubes.pdf

[12] Alboukadel Kassambara. 2017. *Practical guide to cluster analysis in R: Unsupervised machine learning*. Vol. 1. Sthda, .

[13] Matthew Kirk. 2017. *Thoughtful Machine Learning in Python*. O'Reilly Media, Sebastopol. 1–206 pages.

[14] Jon Kleinberg. 2003. An Impossibility Theorem for Clustering. *Advances in neural information processing systems* 15 (2003), 463–470. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.62.3877&rep=rep1&type=pdf

[15] Scikit learn developers. 2020. scikit-learn Release 0.23. https://scikit-learn.org/0.23/. [Online; accessed 30-May-2021].

[16] Maria Rauschenberger and Ricardo Baeza-Yates. 2020. How to Handle Health-Related Small Imbalanced Data in Machine Learning? *i-com* 19, 3 (2020), 215–226. https://doi.org/10.1515/icom-2020-0018

[17] Maria Rauschenberger and Ricardo Baeza-Yates. 2020. Recommendations to Handle Health-related Small Imbalanced Data in Machine Learning. In *Mensch und Computer 2020 - Workshopband (Human and Computer 2020 - Workshop proceedings)*, Bernhard Hansen, Christian AND Nürnberger, Andreas AND Preim (Ed.). Gesellschaft für Informatik e.V., Bonn, 1–7. https://doi.org/10.18420/muc2020-ws111-333

[18] Maria Rauschenberger, Ricardo Baeza-Yates, and Luz Rello. 2020. Screening Risk of Dyslexia through a Web-Game using Language-Independent Content and Machine Learning. In *W4a'2020*. ACM Press, Taipei, 1–12. https://doi.org/10.1145/3371300.3383342

[19] Maria Rauschenberger, Luz Rello, Ricardo Baeza-Yates, and Jeffrey P. Bigham. 2018. Towards language independent detection of dyslexia with a web-based game. In *W4A '18: The Internet of Accessible Things*. ACM, Lyon, France, 4–6. https://doi.org/10.1145/3192714.3192816

[20] Robert L Thorndike. 1953. Who belongs in the family? *Psychometrika* 18, 4 (1953), 267–276. https://doi.org/10.1007/BF02289263

[21] Anna Christina Weigand and Martin Christof Kindsmüller. 2021. HCD3A: An HCD Model to Design Data-Driven Apps. In *Artificial Intelligence in HCI*, Helmut Degen and Stavroula Ntoa (Eds.). Springer International Publishing, Cham, 285–297. https://doi.org/10.1007/978-3-030-77772-2_19