Interactive Labelling of a Multivariate Dataset for Supervised Machine Learning using Linked Visualisations, Clustering, and Active Learning

Mohammad Chegini[a,b,*], Jürgen Bernard[c], Philip Berger[d], Alexei Sourin[b], Keith Andrews[a], Tobias Schreck[a]

[a] *Graz University of Technology, Austria*
[b] *School of Computer Science and Engineering, Nanyang Technological University, Singapore*
[c] *TU Darmstadt, Germany*
[d] *University of Rostock, Germany*

Abstract

Supervised machine learning techniques require labelled multivariate training datasets. Many approaches address the issue of unlabelled datasets by tightly coupling machine learning algorithms with interactive visualisations. Using appropriate techniques, analysts can play an active role in a highly interactive and iterative machine learning process to label the dataset and create meaningful partitions. While this principle has been implemented either for unsupervised, semi-supervised, or supervised machine learning tasks, the combination of all three methodologies remains challenging.

In this paper, a visual analytics approach is presented, combining a variety of machine learning capabilities with four linked visualisation views, all integrated within the mVis (**m**ultivariate **Vis**ualiser) system. The available palette of techniques allows an analyst to perform exploratory data analysis on a multivariate dataset and divide it into meaningful labelled partitions, from which a classifier can be built. In the workflow, the analyst can label interesting patterns or outliers in a semi-supervised process supported by active learning. Once a dataset has been interactively labelled, the analyst can continue the workflow with supervised machine learning to assess to what degree the subsequent classifier has

*Corresponding author
Email address: m.chegini@cgv.tugraz.at (Mohammad Chegini)

effectively learned the concepts expressed in the labelled training dataset. Using a novel technique called automatic dimension selection, interactions the analyst had with dimensions of the multivariate dataset are used to steer the machine learning algorithms.

A real-world football dataset is used to show the utility of mVis for a series of analysis and labelling tasks, from initial labelling through iterations of data exploration, clustering, classification, and active learning to refine the named partitions, to finally producing a high-quality labelled training dataset suitable for training a classifier. The tool empowers the analyst with interactive visualisations including scatterplots, parallel coordinates, similarity maps for records, and a new similarity map for partitions.
Keywords: labelling, clustering, classification, active learning, multivariate data, visualisation

---

1. Introduction

A multivariate dataset is a dataset with more than one dimension. Partitioning a multivariate dataset into labelled classes (partitions) is one of the most prominent supervised machine learning (ML) tasks. Every record in a partitioned dataset must belong to exactly one of the partitions: records cannot belong to multiple partitions, nor can they be left belonging to no partition.

Once a classifier has learned the characteristics of a given multivariate dataset in the training process, the ML model can thereafter be used to automatically partition other, similar datasets. The state of the art in ML demonstrates the effectiveness of today's classifiers in many domains, from the detection of attacks in computer networks [1] to facial image data analysis [2].

Two prerequisites for effective ML techniques are the availability of (1) sufficiently large training datasets and (2) labels provided with those datasets. Without labels, a supervised ML model cannot be trained. Without sufficient numbers of labelled records for training, the supervised ML model will not perform effectively.

However, the unavailability of labels for many real-world datasets is often the bottleneck in supervised ML applications. Today's scientists are often overwhelmed by thousands or even millions of unlabelled records in datasets, all of which are thus unavailable for supervised ML. Given a means to more effectively support analysts in the labelling process, a plethora of unsolved real-world data-centered challenges could be addressed with ML techniques.

The particular challenge addressed by the approach can be exemplified by a domain expert wanting to use a previously unknown multivariate dataset for supervised ML, where neither the characteristics of the dataset are known, nor are there any labels or labelled records.

Sometimes, the cost of labelling a dataset is significantly higher than the cost of creating it [3] and effective labelling solutions are still scarce. Analysts are confronted with the problem of making sense of a dataset, for example by identifying data characteristics such as frequent patterns or outliers. Active learning (AL) techniques, where the system periodically asks the user to label chosen records, can assist in the labelling process. However, since no labels exist at the beginning, AL techniques often suffer from bootstrap problems [4].

Adding to the challenge is that an appropriate *label alphabet*, the vocabulary of labels, is generally unknown at the start of such a process, given an unknown dataset and/or users with ill-defined information needs. In some situations, different label alphabets might be appropriate, depending on the task at hand or a user's individual preferences. Analysts often derive the labels appropriate for a specific dataset and task from the data itself, exploiting the characteristics encoded in the multivariate data records and dimensions. In other situations, analysts rely on special domain knowledge to come up with initial labels. In any of these cases, neither AL tools nor the results of classifiers are particularly helpful for the determination of a label alphabet. Furthermore, the label alphabet is often subject to change during the labelling process itself.

Combining the strengths of humans and computers has been shown to be highly beneficial for the ML process [5] as well as for information visualisation and visual analytics (VA) [6]. The visual interactive labelling (VIAL) technique

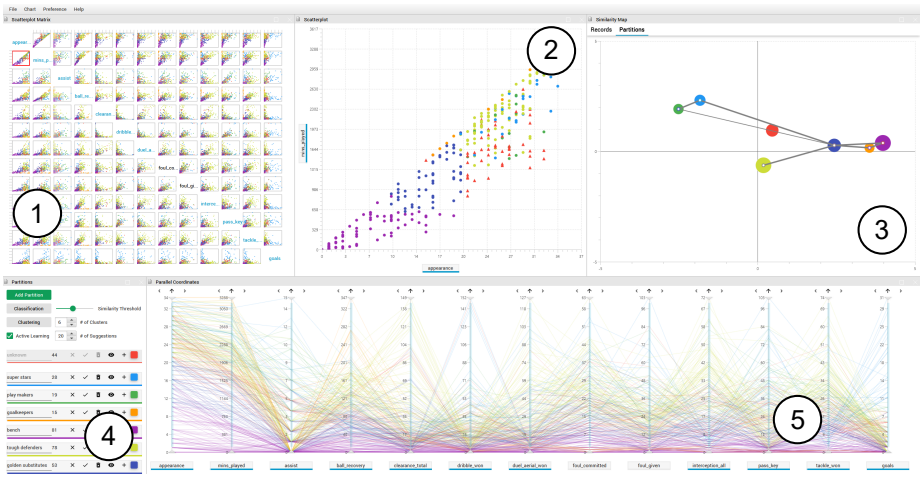


Figure 1: *The scatterplot matrix (SPLOM) view ① shows the bivariate relationships between dimensions. The analyst can select a scatterplot from the SPLOM to show it in detail ②. The partition similarity map ③ shows partitions grouped by similarity and colour-coded as indicated in the partitions panel ④. If two partitions have associated dimensions (through user interaction), they are connected by a line. The parallel coordinates view ⑤ shows the dimensions of the dataset. Dimensions participating in the machine learning algorithms are indicated with a blue ribbon.*

[7] combines ML principles with interactive visual interfaces for the effective selection of records for labelling. This principle has been adopted here. With the highly iterative VIAL process, a classifier can be continuously updated according to new label information provided by the user. Embedded AL strategies guide the user towards records which, once labelled, are likely to improve the underlying ML model. In mVis (**m**ultivariate **Vis**ualiser), this principle is complemented with interactive visual interfaces for data exploration, allowing the meaningful selection and labelling of records based on insights gained by the user, in addition to those suggested by AL. Figure 1 shows the user interface of mVis.

The interactive visual approach described in this paper enables analysts to label records and create partitions of a previously unknown dataset in an effec-

tive and efficient way. While analysts may start without any knowledge about the dataset and the label alphabet, the output of the implemented approach is a labelled training dataset which can be used for supervised ML. The labelling process represents a pathway from unsupervised ML, through semi-supervised ML, to supervised ML. This pathway is guided by algorithmic models built upon both unsupervised and supervised ML principles. The approach presented here has three main components: (a) visual exploration, (b) interactive visual labelling, and (c) automatic guidance.

Firstly, the dataset can be explored interactively using a palette of linked visualisations, including scatterplots, a SPLOM, similarity maps, and parallel coordinates. These tools allow interactive visual exploration of a dataset's records and dimensions to both discover and then interactively label groupings, patterns, and outliers. Moreover, a novel view called the partition similarity map shows the similarity of partitions (each represented by a coloured node), based on the centroid of each partition. A link is drawn between two partitions if both partitions are associated with at least one common dimension. A dimension is associated with a partition, if the user interacted with that dimension while adding records to the partition.

Secondly, records can be selected and labelled in any of the interactive views, leading to labelled datasets which can be used for supervised ML. During the labelling process, dimensions that the user interacted with to perform labelling are added to the label as metadata. This solution facilitates labelling without the need for domain-specific visual representations by leveraging the structural information provided within a multivariate dataset, such as patterns and relations between records and dimensions. The original VIAL process is extended by incorporating classic k-means and hierarchical clustering to the supervised ML techniques.

Thirdly, clustering, active learning, and classifier algorithms are all available to support the effective and efficient selection of candidate records for labelling. In addition, using a new *automatic dimension selection* technique, interactions of the user with specific data dimensions are remembered and fed into the semi-

supervised and supervised ML techniques. For example, if the user selected records in a scatterplot of dimensions A and B, and added these records to a partition, then dimensions A and B are associated with that partition. Initially, dimensions which are not interacted with play no role in the ML algorithms, but the user has final control over which dimensions should be included in or excluded from the ML algorithms.

The primary contribution of this paper is to elaborate how linked interactive visualisations can be effectively integrated with classic ML algorithms to provide guidance during the labelling process without overwhelming the user. This work adds to explorations of the potentially large design space of visual analytics methods facilitated by active learning, and sets examples upon which to build future work. To demonstrate the effectiveness of the approach, it has been incorporated into the mVis system and tested with a real-world football dataset.

2. Related Work

VA applications benefit from both unsupervised and supervised ML algorithms to support data exploration and analytical reasoning [8]. Table 1 gives an overview of some of the techniques which support interactive labelling. Unsupervised machine learning techniques can be applied to unlabelled datasets, since they do not require any training data. For example, clustering techniques [9] can be used to find groupings of similar records within a dataset. Exploratory information visualisations can be used to visually cluster (and then select) records according to their similarity or dissimilarity, since similar records are typically closer together in the visualisation. Semi-supervised ML techniques [10] require at least some labelled data records before they can be used. In active learning, some labelled data records are provided, and the system interactively collects new examples through additional input from the user. Supervised ML techniques such as classification [2] require a proper training set of labelled records.

Visual Clustering. Exploratory information visualisations can be used as interactive interfaces to select (groups of) similar records or to identify and select

	Visual Clustering	Clustering	Classification	Active Learning
ML Type	Unsupervised	Unsupervised	Supervised	Semi-Supervised
Existing Labels	Not Required	Not Required	Required	Required
Records to Label	Chosen by user.	All unlabelled records.	Unlabelled records closer than a threshold to a label.	Specific number of records chosen strategically.
Creates Partitions	By User	Yes	No	No
Algorithms	PCA, MDS, t-SNE	K-means, Hierarchical	Random Forest	Random Forest
Triggered By	User	User	User	System

Table 1: *Techniques which support interactive labelling of records.*

outliers. Scatterplots visualise records along two chosen dimensions. Records which are similar (in those two dimensions) are plotted close together. Dimensionality reduction and projection methods can be used to generate a *similarity map*, which visually infers a clustering by spatial proximity. Records closer together in the projected similarity map are more similar to one another in the high-dimensional space [11, 12]. In parallel coordinates [13], similar records are represented by polylines which follow similar paths. It is also possible to filter records by ranges on each dimension.

Cluster Sculptor [14] is an interactive clustering system which allows the user to update the cluster labels of a dataset iteratively. The system relies on a t-SNE projection view, label diffusion, and dissimilarity transform techniques. Lee et al. [15] built a system called iVisClustering based on latent Dirichlet allocation (LDA), which helps the user to perform clustering with interactive visualisation, including parallel coordinates and scatterplots. RCLens [1] supports the identification and exploration of rare categories (minority classes), utilising an active learning algorithm to help the analyst iteratively finds rare categories within the dataset. In mVis, interactive clustering is used to guide the analyst in finding some preliminary structure in the dataset.

Clustering. Classic clustering techniques such as k-means [16] and hierarchical clustering [17] are used to form groups (partitions) of records according to their

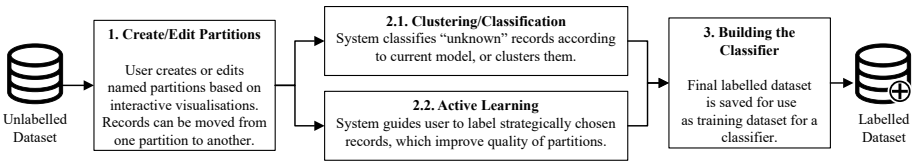


Figure 2: *The workflow for interactive labelling. First, the analyst creates and names (labels) partitions in the dataset and assigns records to them. In the second and the third step, with guidance from the system, partitions are refined, and more records are added (labelled). After sufficient iterations, based on the quality of the result, the analyst saves the labelled dataset to be used as a training dataset for a classifier.*

similarity. The result of these clustering algorithms can be visually inspected. In early work, gCluto [18] allowed an analyst to visually inspect clusters created by running multiple clustering techniques while tuning the parameters. Nam et al. [19] proposed a technique allowing analysts to tune the parameters of clustering algorithms interactively to find suitable clusters based on the user's needs. The technique was proposed and tested on high-dimensional datasets. Later, Andrienko et al. [20] suggested a general approach to find clusters in large sets of spatial data objects and demonstrated the approach on a dataset of trajectories. Kwon et al. [21] developed Clustervision, which clusters a dataset with various clustering algorithms, and ranks and visualises clustering results based on quality metrics, allowing analysts to choose the most suitable for their purpose.

Classification. Classification is a supervised ML technique which can identify to which class a record belongs, given a sufficiently large training set of labelled records. VA can help classification algorithms by adding the knowledge of the user in an iterative manner [22]. For example, iVisClassifier [2] supports a user-driven classification process, where the analyst explores multi-dimensional data through a supervised dimensionality reduction and performs classification.

Active Learning. The process of labelling records to create training data usually requires tedious amounts of repetitive work by human analysts. Active Learning

(AL) strategies interactively collect new labelled records by judiciously asking for additional input from the user [10]. To make the process more effective and efficient, it is crucial for the system to propose records for interactive labelling wisely, choosing those records which are most likely to improve the underlying ML model.

Known strategies include looking for helpful records a) near decision boundaries of margin-based classifiers [23, 24]), b) with high entropy of class probabilities [25], c) with high uncertainty of a committee of classifiers [26, 27], or d) to reduce risk [28] or variance [29].

Only a few existing techniques work independently of the learning model, by choosing to focus on data characteristics. Some approaches explicitly allow users to select records in the kind of interactive visualisations typically used for data exploration or analysis [30–32]. The visual interactive-labelling (VIAL) process [7] combines both model-based active learning and interactive visual interfaces to support the human-centered selection and labelling of records. Recent experiments have shown that individual strategies have different complementary strengths [3, 33].

mVis extends the approach of VIAL: analysts can use linked interactive visualisations to help mitigate the cold start problems associated with active learning. In addition, clustering and classification are provided to better guide the user in the labelling task.

3. Interactive Visual Labelling

It is often the case that an analyst is confronted by an exploratory scenario in which the records in the dataset are unknown, and no labels are assigned to them. For ML applications, similar records must be grouped together and manually labelled in order to use the dataset as a training dataset. Since the definition of similarity varies from dataset to dataset, it is necessary to offer support to analysts to interactively group and label records and iteratively construct the label alphabet (L).

In an exploratory scenario, there is no single absolute L for a dataset. Based on the knowledge of the expert, L and the records assigned to each partition may vary significantly. Thus, a dynamic L is necessary to empower the analyst to build an appropriately labelled dataset fitting the purpose of the desired classifier. This includes allowing the analyst to (1) add new labels to L, (2) delete labels from L, (3) add or remove records to a label in L and (4) rename a label in L.

A partition, identified by P_i, is a set of records from the dataset, whereby each record must belong to one and only one partition. The union of all partitions P contains all records in the dataset. Each partition also has a label, l_i, which is a text string belonging to the label alphabet L, and a set of related dimensions Dim_i:

$$P_i = (l_i, Rec_i, Dim_i) \tag{1}$$

where:

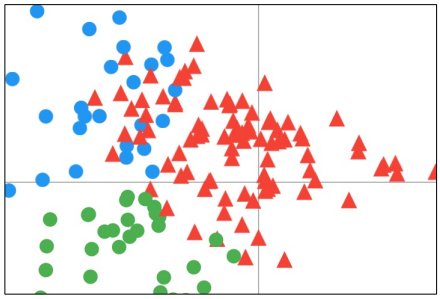
l_i is one of the labels in the alphabet L. One label exists for each partition, one partition exists for each label.

Rec_i is the set of all records labelled as l_i. There is a non-injective non-surjective function which maps records to partitions. In other words, every record is mapped to one and only one label at a time; $f : P \rightarrow L$, where f is the function which maps records to labels. The mapping is guided by the system, but is the analyst's task.

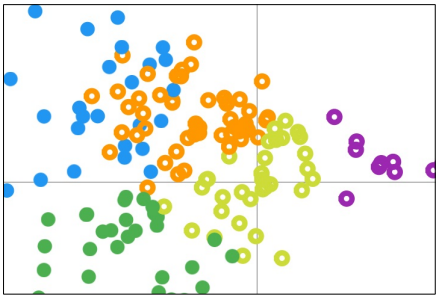
Dim_i is a set of dimensions that the user interacted with while adding records to P_i. It is possible for a dimension to be associated with more than one partition, and there could be dimensions which are not associated with any partition.

3.1. Analyst Role: Selection and Labelling

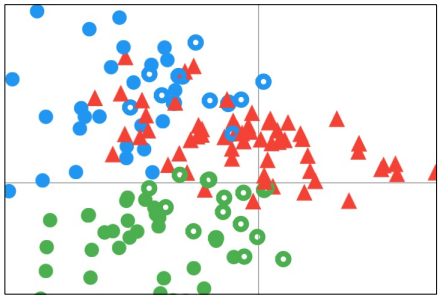
Figure 2 illustrates the workflow in which an analyst creates and edits partitions and labels records interactively. Initially, all records are assigned to a



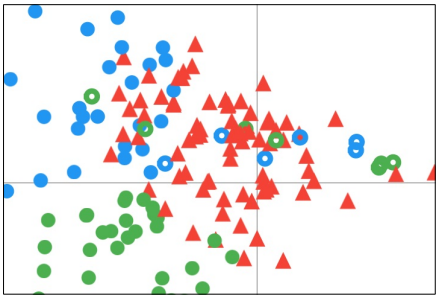
(a) *The initial state with three partitions: green circles, blue circles, and red triangles (unknown).*



(b) *After clustering from (a), with k-means (k=3). The newly suggested clusters (partitions) are orange, yellow, and purple.*



(c) *After classification from (a) with a similarity threshold of 70%. Hollow circles are the system suggestions.*



(d) *After active learning from (a) in which the system suggested 30 records for labelling by the user.*

Figure 3: *The results of clustering, classification, and active learning in mVis, each applied to the initial state shown in (a). In each case, hollow circles indicate records with labels suggested by the system. Solid circles indicate previously approved labels. Solid red triangles indicate currently unlabelled records belonging to the unknown partition.*

special partition labelled as *unknown*. In the first step, the analyst creates at least one partition, assigns records to it, and gives it a label. Later, the analyst can perform clustering and classification to label further records currently labelled as *unknown*. In the case of clustering, the system creates new partitions of *unknown* records and assigns temporary labels to them. In the case of classification, currently labelled records are used as a training set to label other *unknown* records based on existing partitions, which then potentially expands them. In either case, the system provides guidance by suggesting new labelled records, which the analyst can then approve or reject.

Periodically, the system suggests that the analyst should manually label a specific number of records by running active learning techniques. These records are wisely chosen to further resolve ambiguity in the dataset. The analyst investigates the result and decides if the alphabet and labels on records need further improvement. The process finishes when the analyst is satisfied with the quality of the result. The result of this process is a label alphabet (L) and a set of labelled partitions (P_i), in other words a labelled training dataset for a classifier. Records still labelled *unknown* may or may not be included in the output.

3.2. System Role: Guidance

The system's role is to suggest records for labelling to the analyst by visual clustering, classic clustering, classification, and active learning. Table 1 differentiates between these four kinds of technique.

In terms of visual clustering, the system provides similarity maps using one of three different projections: PCA, MDS, and t-SNE. Similar records are grouped by proximity and the analyst can efficiently create and modify partitions by visually inspecting these views.

In terms of classic clustering, the user can ask the system to cluster currently unlabelled records, using either k-means or hierarchical clustering. This results in a number of newly created partitions (i.e. clusters) with temporary labels, which the analyst can then either rename, approve, or reject.

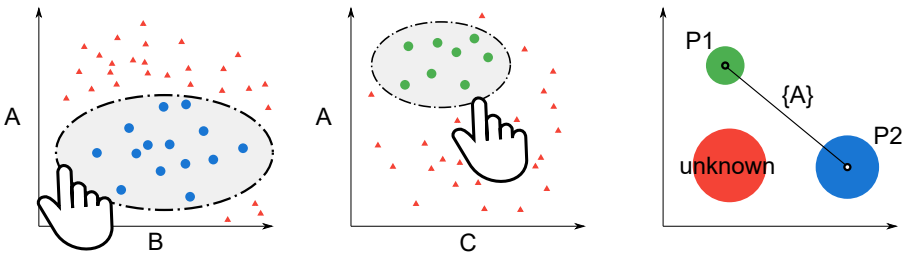
Once sufficient numbers of records have been labelled, the analyst can use classification to help label further records. After performing the classification, the system calculates the similarity of each record (r_j) to each partition (P_i). The sum of all these scores for each record is always 100. The user can then define a *similarity threshold*. The system will suggest adding records with a similarity score higher than the threshold to the corresponding partition. If multiple partitions have a higher similarity score than the threshold, the system will choose the partition with the highest score. The user can either approve or reject the new suggestions. In classification, no new partitions or labels are created, but records may be added to the existing partitions P_i.

For active learning (AL), the system also requires a sufficient number of labelled records. It then chooses those unlabelled records which are most likely to further resolve ambiguity in the dataset, and asks the analyst to manually label them. Unlike clustering and classification, AL is not triggered by the user, but periodically by the system. Figure 3 shows the differing results of clustering, classification, and active learning in mVis.

The set of all dimensions associated (by user interaction) to at least one partition, Dim, is the union of all Dim_i. The above techniques do not always incorporate all of the dataset's dimensions in their various calculations. Instead, a set of *participating* dimensions is maintained by the system. Initially, the set of participating dimensions is set to be Dim, a feature called *automatic dimension selection*. However, the analyst has final control, and can include or exclude any dimensions from the set of participating dimensions. The final result of the workflow is a labelled dataset which includes P_i, L, and a set of related dimensions.

4. mVis System Overview

The mVis system consists of four data visualisation views and a panel to control partitions. mVis is written in Java and uses JavaFX for its user interface. It supports traditional mouse and keyboard as well as multi-touch user input.



(a) *Creating P_1.* **(b)** *Creating P_2.* **(c)** *Similarity map.*

Figure 4: *Records are added to partition P_1 (blue) from AB, then to partition P_2 (green) from AC. The partition similarity map shows a link between P_1 and P_2 because they are both associated with dimension A.*

The system has been tested on a PC with a 3.4 GHz Intel i7-6700 CPU and 64 GB of RAM, running 64-bit Windows 10.

4.1. Visualisations and Partitions Panel

The four linked exploratory data visualisations built into mVis are: SPLOM, scatterplot, similarity map (projection by PCA, MDS, and t-SNE), and parallel coordinates plot. All the visualisations are connected through standard brushing and linking, so selections and changes in one view are reflected in all other views. Moreover, the user can close, rearrange, or enlarge any view. Axis tick labels in the scatterplot and parallel coordinates views reflect the original values in the dataset. Coordinates in the SPLOM view are normalised, so axis tick labels are omitted.

The SPLOM provides an overview of the entire dataset by showing all bivariate projections of n dimensions. The result is a matrix of n^2 scatterplots [34]. The SPLOM can indicate both patterns of records in two dimensions and correlations between pairs of dimensions, which can then be examined in individual scatterplots.

Individual scatterplots are widely used for regression analysis [35] or exploration of local patterns [36]. In mVis, the user can select a scatterplot in the SPLOM, which is then shown enlarged in the scatterplot view.

The parallel coordinates visualisation shows the dimensions of a dataset as parallel vertical axes and its records as horizontal polylines [13]. Parallel coordinates provide a concise overview of the entire dataset and are suitable for exploring correlations between neighbouring dimensions. A parallel coordinates plot has been shown to outperform individual scatterplots when the task requires interaction with more than two dimensions [37]. In mVis, the parallel coordinates view supports several interactions, including brushing and selection of records, filtering of records by dragging sliders at the top and bottom of each axis, reordering axes, and inverting axes.

The similarity map view provides two kinds of similarity map: a similarity map of records and a similarity map of partitions. The record similarity map shows all the records in the dataset visually clustered by similarity, using one of three projection techniques: PCA, MDS, or t-SNE. More similar records are closer together in the similarity map. The default projection technique is t-SNE, but the user can choose a different technique in the preference menu.

The partition similarity map shows all currently defined partitions, grouped by similarity in the form of a node-link diagram. Each partition is represented as a circular node, whose size corresponds to the number of records in the partition. If two partitions share associated dimensions, then a line (link) is drawn to connect them, whose width corresponds to the number of shared associated dimensions. Figure 4 illustrates how such a diagram is created. First, in Figure 4a, the analyst creates a partition P_1 containing records selected in the scatterplot of dimension A against dimension B (AB). Later, in Figure 4b, the analyst assigns records to P_2 from the scatterplot AC. Since both partitions are associated with dimension A, there a link is drawn between P_1 and P_2, as shown in Figure 4c.

The partitions panel shown in Figure 5 gives the analyst the possibility to create new partitions, assign records to partitions, and delete partitions. The name (label) of a partition can be edited and the colour assigned to it can be changed. A special partition labelled *unknown* contains all currently unlabelled records and is initially coloured red. If a partition is deleted, all

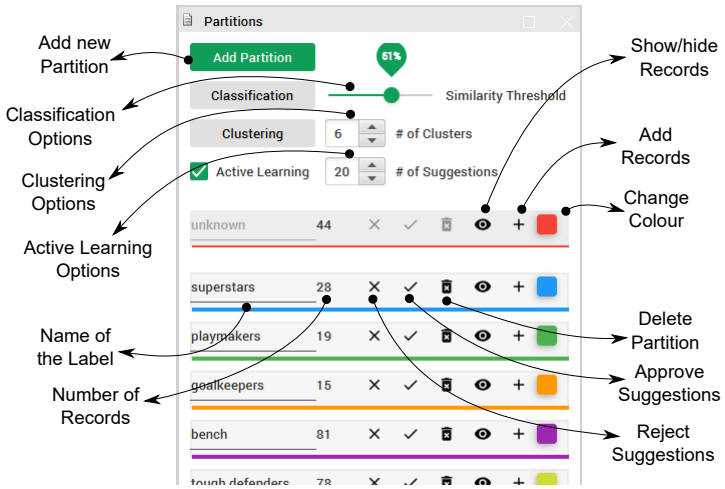


Figure 5: *The partitions panel. In the upper part of the panel, the analyst can create partitions and obtain suggestions for records to add to them. The lower part of the panel is for manipulating existing partitions.*

records contained within it are returned to the *unknown* partition. The analyst can temporarily hide the records in a given partition. Clicking the "+" button next to a partition adds currently selected records to it.

Records which have been manually assigned to a partition or approved by the analyst are considered to be "ground truth" and are represented by solid circles in the SPLOM, scatterplot, and record similarity map. Hollow circles represent records with a suggested partition, colour-coded according to the partition. Unlabelled records belong to the *unknown* partition and are represented by solid triangles, in the colour assigned to the *unknown* partition (initially red, but the colour can be changed by the analyst).

In the upper part of the partitions panel, the analyst can initiate ML techniques such as clustering and classification to obtain suggestions for records to assign to partitions. Such records become hollow circles and are recoloured to the suggested partition's colour until either approved or rejected by the analyst by clicking the Reject or Approve buttons next to each partition in the panel. Suggested records which are rejected become solid (red) triangles again and are

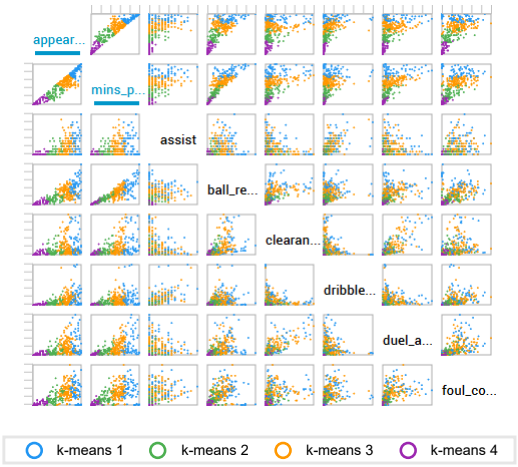


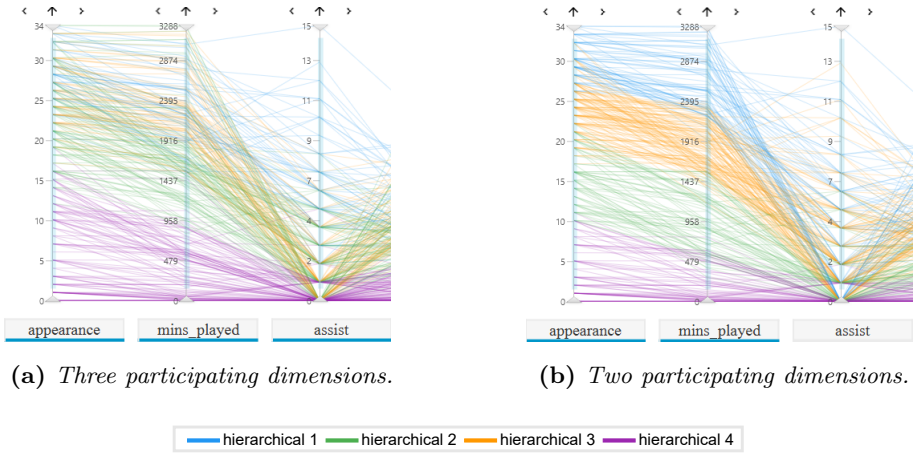
Figure 6: *The SPLOM after k-means clustering (k=4) with automatic dimension selection. A blue ribbon beneath a dimension name indicates its participation in the ML technique. The first two dimensions* appearances *and* mins_played *from the football dataset have participated in the clustering, which is reflected in the better results in their rows and columns.*

moved back into the *unknown* partition. Approved records become part of the partition and are henceforth represented by solid circles.

4.2. Machine Learning Modules

Various ML algorithms are implemented to support the interactive labelling process, including dimensionality reduction, clustering, classification, and active learning. All of these algorithms are implemented using the Java library called DMandML [38]. Interactions with a ML algorithm can be unintuitive and overwhelming to use at times. mVis uses simple widgets and a minimal number of exposed parameters to keep interactions intuitive.

While assigning records to partitions, the system keeps track of the dimensions the user interacted with, maintaining a set of associated dimensions for each partition. By default, only those dimensions associated with at least one partition participate in the ML algorithms. The user can toggle participation of a dimension by clicking on the dimension name in the SPLOM or parallel coor-



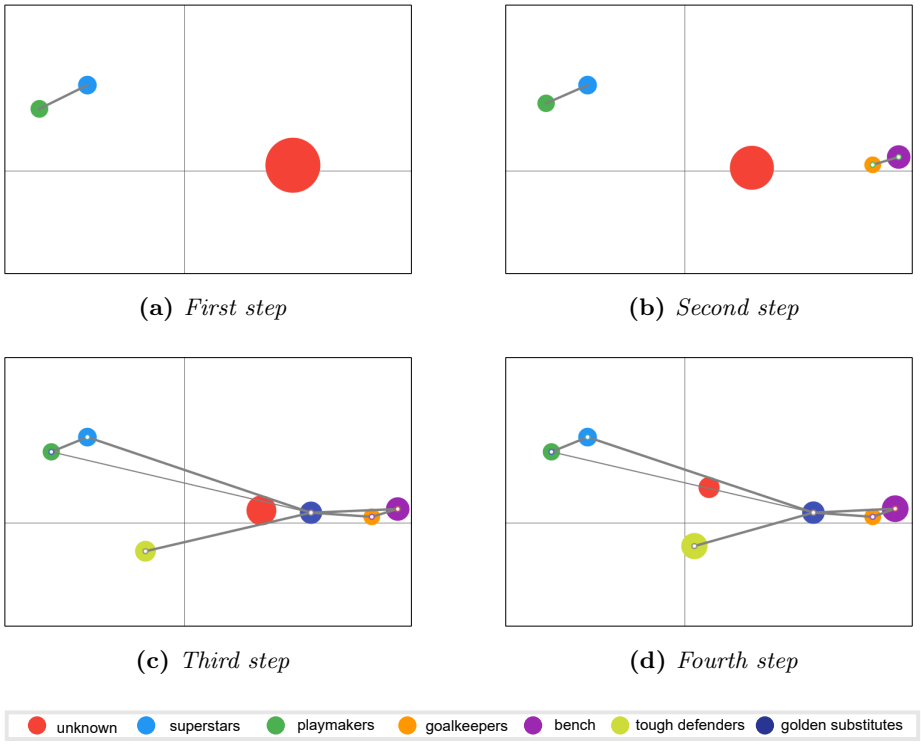
(a) *Three participating dimensions.* **(b)** *Two participating dimensions.*

Figure 7: *Part of the parallel coordinates plot after hierarchical clustering (k=4). The clusters are more visually appealing in (b).*

dinates view. Participating dimensions are indicated by a blue ribbon beneath the dimension name. Figure 6 shows k-means clustering (k=4) utilising only two of the eight available dimensions. Figure 7 demonstrates the effectiveness of automatic dimension selection when hierarchical clustering is performed on the dataset.

At any stage, the analyst can perform clustering by clicking on the clustering button in the partitions panel. The system will then cluster all currently unlabelled (*unknown*) or unapproved records using k-means or hierarchical clustering. By default, mVis uses k-means, but the user can change the algorithm by selecting hierarchical in the menu. For each cluster, a new partition is created and given a temporary name (label) of the form `k-means #cn` or `hierarchical #cn`, where `#cn` is the number of the cluster. Records assigned to a cluster are simply suggestions by the system and require subsequent user approval.

Alternatively, once sufficient records have been assigned labels, the analyst can run a classifier to classify those records which are currently either *unknown* or unapproved. The system then runs a *Random Forest* classifier using the already labelled (approved) records as a training set. The user can control the



(a) *First step* **(b)** *Second step*

(c) *Third step* **(d)** *Fourth step*

Figure 8: *Four steps of labelling the football dataset, shown in the partition similarity map. (a) The user manually creates superstars and playmakers partitions. (b) After a clustering step using k-means, two partitions called goalkeepers and bench are approved by the user. (c) The user creates tough defenders and golden substitutes partitions and assigns records to them. (d) The user performs active learning to label more records. The final result is a label alphabet with seven members.*

number of suggestions by adjusting the similarity threshold with the slider next to the Classification button. While the slider is adjusted, a number indicates its precise value. With a higher threshold, only those records more similar to a specific partition will be suggested. Similar to clustering, the analyst can then approve or reject the classification result.

Periodically, the system actively guides the user to manually label a number of records using active learning. The suggested labels can either be approved or rejected. The number of suggested records can be fine-tuned and active learning can be turned off completely with the checkbox in the partitions panel.

The current design of mVis has visualisation and algorithmic limitations. Regarding the visual scalability of the label alphabet (number of partitions), upto around twelve distinct colours can be comfortably distinguished [39]. The SPLOM and parallel coordinates views are limited by the amount of available screen space. mVis runs in real-time with a football dataset comprising 42 dimensions and 318 records on a 25-inch desktop display at a resolution of 2560×1440. One possibility to increase scalability would be to apply subspace clustering to provide an initial set of records and dimensions to explore [40]. The currently implemented ML algorithms run in real-time for the aforementioned number of partitions and dimensions.

5. Use Case

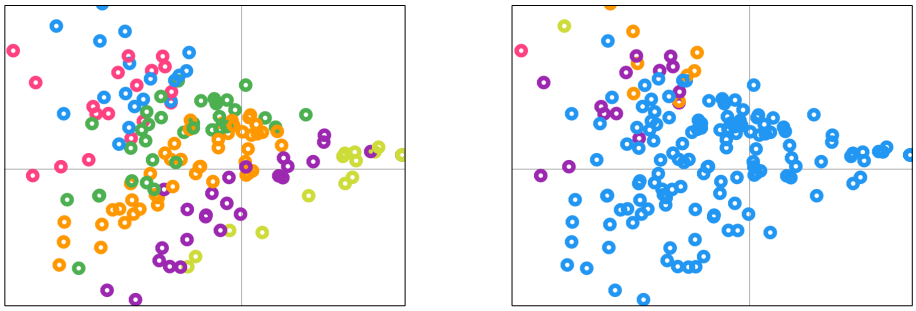
The following use case utilises a football dataset of players from 16 clubs participating in five top European leagues in the 2017/18 season [41]. The records are individual players, the dimensions are players' attributes such as the number of match appearances, committed fouls, assists, pass accuracy, and so forth. The dataset comprises 318 records and 13 dimensions.

The goal of the analyst exploring this dataset is (1) to group the players into labelled partitions based on their characteristics, and (2) to use the dataset to train a classifier for other seasons of the same or even entirely different football leagues.

For an initial grouping, the analyst wants to identify match-winning players and label them as `superstars`. The analyst proceeds by selecting the scatterplot of goals against assists in the SPLOM. The analyst creates a partition, labels it `superstars`, and includes all data records with high numbers of goals and assists.

Another important category of players are the so-called `playmakers`, having a high number of assists and key_passes. By filtering players with a high number of assists and key_passes in the parallel coordinates view, the analyst can find records to add to the `playmakers` partition. To expand the label content so that not only top players are included, the analyst searches for players similar to those selected. To this end, the analyst sets the *Classification Threshold* slider in the partitions panel (see Figure 5) to 60% and clicks the Classification button. As a result, the system suggests 100 records be labelled as `playmakers` and 20 as `superstars`. The analyst realises that this is a large number of players to be added to each partition and decides to reject the suggestion. Later, the analyst performs another classification with the slider at 80%. This time, 15 records are suggested to be added to `playmakers` and 5 to `superstars`. The analyst accepts the suggestion by clicking the Approve button of both partitions. The partition similarity map in Figure 8a shows the state of the dataset after creating the partitions `superstars` and `playmakers`.

Apart from these two obvious choices, the relationships between other dimensions are unfamiliar to the analyst. The analyst turns off the *automatic dimension selection* feature, chooses 4 as the value in the *# of Clusters* field, and performs a k-means clustering. By making all partitions except one invisible, the analyst inspects the newly suggested partitions one by one. The first suggested partition is `k-means 1`, containing 16 records. The analyst realises all the dimensions for these records are zero except appearance, mins_played, and ball_recovery. Therefore, the analyst renames the `k-means 1` partition to `goalkeepers`. Similarly, the analyst renames `k-means 2` with 88 records to `offensive players`. This partition is associated with the dimensions key_passes, dribbles_won, and goals. Next, the analyst renames `k-means 3` with 71 players to `defensive players`, since it is associated with ball_recovery, clearances, aerial_duels_won, fouls_committed, and interceptions.



(a) *K-means.* **(b)** *Hiearchical.*

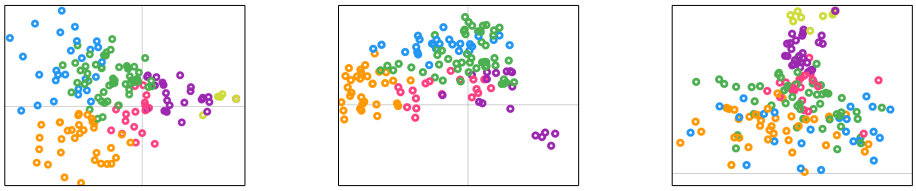
Figure 9: *The results of k-means and hierarchical clustering for k=6, using offensive attributes of football players.*

Finally, the partition `k-means 4` with 116 records is renamed `bench`. This partition is associated with a low number of appearances and mins_played.

The goal is not to create partitions based solely on a player's role on the field, so the analyst decides to delete the partitions `offensive players` and `defensive players` by clicking their Delete buttons, but to retain the partitions `goalkeepers` and `bench` by clicking their Approve buttons. Figure 8b shows the state of the dataset after this step.

Similar to the group of match-winning `superstars`, the analyst wants a label for defensive players having a high impact on the team. From the previous exploration, the analyst already knows which dimensions are associated with defensive characteristics. Therefore, the analyst creates the `tough defenders` partition characterised by their performance in the dimensions aerial_duels_won, interceptions, and tackles_won.

Exploring further, the analyst selects all records which (1) belong to the `bench` partition and (2) have either a high number of goals, key_passes, clearances, dribbles_ won, assists, or aerial_duels_won and calls the new partition `golden substitutes`. To further support the analyst, the remaining unlabelled records (belonging to the *unknown* partition) can be suggested to existing partitions via active learning. This helps refine existing labels and increasing the overall quality, an



(a) *MDS* **(b)** *PCA* **(c)** *t-SNE*

Figure 10: *The three projection techniques provided by the record similarity map. The colours were assigned by an initial k-means clustering with k=6.*

option which is not possible in traditional ML techniques.

The analyst investigates the result shown in the partition similarity map of Figure 8d. The `tough defenders` partition is linked to `golden substitutes` partition, since they are both associated with the `clearances` dimension. Also, `playmakers` and `superstars` are relatively close to one other in the partition similarity map, possibly because `playmakers` and `superstars` share similar offensive characteristics. Since the user interacted with eleven dimensions, only two dimensions are not highlighted with a blue ribbon.

The result of the session is a labelled football players dataset with meaningful partitions, which can be used as a training dataset for a classifier for other seasons or different leagues.

6. Discussion and Future Work

Characterising, comparing, and grouping (partitioning) the records in a dataset are among the most essential tasks in data analysis. The implemented approach supports these tasks with an interactive visual labelling tool. Using interactive visualisations, an analyst can identify and label groups of records in a dataset initially containing no pre-labelled records. Once the analyst has provided an initial labelling, the system supports labelling more records via clustering, classification, and active learning. With the help of clustering, the analyst can find structures in the dataset which may not be visible by manual

exploration. Using classification, the labelled data will be used as a training set for records which are not yet labelled. Moreover, the active learning module regularly makes strategic suggestions to improve the quality of partitions. The user is always responsible for approving or rejecting suggestions, which increases overall trust in the result. As the presented use case shows, algorithmic support helps efficiently propagate current labelling to more records. The approach supports both the creation of a new label alphabet and the refinement of an existing label alphabet.

Currently, mVis supports both k-means and hierarchical clustering. Although k-means is more scalable and hierarchical is more flexible, neither is superior to the other. It is the responsibility of the domain expert to choose the most suitable algorithm in a specific situation. Figure 9 shows the results of k-means and hierarchical clustering in the football dataset.

Three projection algorithms (MDS, PCA, and t-SNE) are supported for the record similarity map. Research by Bernard et al. [3] shows that users prefer t-SNE as a dimensionality reduction technique for labelling tasks and later switch to PCA and MDS for validation. Therefore, the default algorithm in mVis is t-SNE. Figure 10 shows the differences between these algorithms, performed on the football dataset.

Formative usability evaluation would provide valuable insights into how to improve the system and its user interface. A user study could help evaluate the implemented approach. For example, an experiment could measure classification accuracy as analyst interactions (number of clicks, number of created labels, etc.) with the system increase.

Since the labelling process is performed iteratively, it might be beneficial to keep a history of all user interactions and operations. The user may wish to revisit earlier labelling decisions, and possibly update the alphabet and partitions. Providing a visual history of labelling provenance, and how to propagate changes to earlier labelling decisions is an interesting research topic for future work. This also raises the need for appropriate comparative visualisation techniques [42], to contrast the different selections. Finally, one possibility to provide an analyst

with interesting initial views to start labelling would be to use Scagnostics or Pargnostics features [43] to guide the user to relevant views.

7. Concluding Remarks

This paper presented an approach to make partitions on a multivariate dataset that does not contain any labelled record. Using appropriate views including partition similarity map, the analyst can manually label records with the help of classification, clustering and active learning algorithms. The result of the process is a properly labelled and partitioned dataset. An implementation of the approach called mVis had shown its usefulness for a real-world football dataset.

References

[1] Lin H, Gao S, Gotz D, Du F, He J, Cao N. Rclens: Interactive rare category exploration and identification. Computer Graphics Forum (CGF) 2017;36(8):458–86. doi:10.1111/cgf.13092.

[2] Choo J, Lee H, Kihm J, Park H. iVisClassifier: An interactive visual analytics system for classification based on supervised dimension reduction. In: Proc. 2010 IEEE Conference on Visual Analytics Science and Technology (VAST 2010). IEEE; 2010, p. 27–34. doi:10.1109/VAST.2010.5652443.

[3] Bernard J, Hutter M, Zeppelzauer M, Fellner D, Sedlmair M. Comparing visual-interactive labeling with active learning: An experimental study. IEEE Transactions on Visualization and Computer Graphics (TVCG) 2018;24(1):298–308. doi:10.1109/TVCG.2017.2744818.

[4] Attenberg J, Provost F. Inactive learning?: Difficulties employing active learning in practice. SIGKDD Explor Newsl 2011;12(2):36–41. doi:10.1145/1964897.1964906.

[5] Amershi S, Cakmak M, Knox WB, Kulesza T. Power to the people: The role of humans in interactive machine learning. AI Magazine 2014;35(4):105–20. doi:10.1609/aimag.v35i4.2513.

[6] Sacha D, Sedlmair M, Zhang L, Lee JA, Peltonen J, Weiskopf D, et al. What you see is what you can change: Human-centered machine learning by interactive visualization. Neurocomputing 2017;268:164–75. doi:10.1016/j.neucom.2017.01.105.

[7] Bernard J, Zeppelzauer M, Sedlmair M, Aigner W. Vial: a unified process for visual interactive labeling. The Visual Computer 2018;34(9):1189–207. doi:10.1007/s00371-018-1500-3.

[8] Endert A, Ribarsky W, Turkay C, Wong BW, Nabney I, Blanco ID, et al. The state of the art in integrating machine learning into visual analytics. IEEE Transactions on Visualization and Computer Graphics (TVCG) 2018;24(7):2223–37. doi:10.1109/TVCG.2017.2711030.

[9] Wenskovitch J, Crandell I, Ramakrishnan N, House L, North C. Towards a systematic combination of dimension reduction and clustering in visual analytics. IEEE Transactions on Visualization and Computer Graphics (TVCG) 2018;24(1):131–41. doi:10.1109/TVCG.2017.2745258.

[10] Settles B. Active learning. Synthesis Lectures on Artificial Intelligence and Machine Learning 2012;6(1):1–114. doi:10.2200/S00429ED1V01Y201207AIM018.

[11] Sacha D, Zhang L, Sedlmair M, Lee JA, Peltonen J, Weiskopf D, et al. Visual interaction with dimensionality reduction: A structured literature analysis. IEEE Transactions on Visualization and Computer Graphics (TVCG) 2016;23(1):241–50. doi:10.1109/TVCG.2016.2598495.

[12] Sacha D, Zhang L, Sedlmair M, Lee JA, Peltonen J, Weiskopf D, et al. Visual interaction with dimensionality reduction: A structured literature

analysis. IEEE Transactions on Visualization and Computer Graphics 2017;23(1):241–50. doi:10.1109/TVCG.2016.2598495.

[13] Inselberg A. The plane with parallel coordinates. The Visual Computer 1985;1(2):69–91. doi:10.1007/BF01898350.

[14] Bruneau P, Pinheiro P, Broeksema B, Otjacques B. Cluster sculptor, an interactive visual clustering system. Neurocomputing 2015;150:627–44. doi:10.1016/j.neucom.2014.09.062.

[15] Lee H, Kihm J, Choo J, Stasko J, Park H. iVisClustering: An interactive visual document clustering via topic modeling. Computer Graphics Forum (CGF) 2012;31(3pt3):1155–64. doi:10.1111/j.1467-8659.2012.03108.x.

[16] Lloyd S. Least squares quantization in pcm. IEEE Transactions on Information Theory 1982;28(2):129–37. doi:10.1109/TIT.1982.1056489.

[17] Karypis G, Han EH, Kumar V. Chameleon: Hierarchical clustering using dynamic modeling. IEEE Computer 1999;32(8):68–75. doi:10.1109/2.781637.

[18] Rasmussen M, Karypis G. gcluto: An interactive clustering, visualization, and analysis system. Tech. Report CSE/UMN TR 04-021; Univ. of Minnesota, Department of Computer Science and Engineering, CSE; 2004.

[19] Nam EJ, Han Y, Mueller K, Zelenyuk A, Imre D. Clustersculptor: A visual analytics tool for high-dimensional data. In: Proc. 2007 IEEE Symposium on Visual Analytics Science and Technology. IEEE; 2007, p. 75–82. doi:10.1109/VAST.2007.4388999.

[20] Andrienko G, Andrienko N, Rinzivillo S, Nanni M, Pedreschi D, Giannotti F. Interactive visual clustering of large collections of trajectories. In: Proc. 2009 IEEE Symposium on Visual Analytics Science and Technology. IEEE; 2009, p. 3–10. doi:10.1109/VAST.2009.5332584.

[21] Kwon BC, Eysenbach B, Verma J, Ng K, De Filippi C, Stewart WF, et al. Clustervision: Visual supervision of unsupervised clustering. IEEE Transactions on Visualization and Computer Graphics (TVCG) 2018;24(1):142–51. doi:10.1109/TVCG.2017.2745085.

[22] Paiva JG, Schwartz WR, Pedrini H, Minghim R. An approach to supporting incremental visual data classification. IEEE Transactions on Visualization and Computer Graphics (TVCG) 2015;21(1):4–17. doi:10.1109/TVCG.2014.2331979.

[23] Wu Y, Kozintsev I, Bouguet JY, Dulong C. Sampling strategies for active learning in personal photo retrieval. In: Proc. 2006 IEEE International Conference on Multimedia and Expo. IEEE; 2006, p. 529–32. doi:10.1109/ICME.2006.262442.

[24] Tuia D, Volpi M, Copa L, Kanevski M, Munoz-Mari J. A survey of active learning algorithms for supervised remote sensing image classification. IEEE Journal of Selected Topics in Signal Processing 2011;5(3):606–17. doi:10.1109/JSTSP.2011.2139193.

[25] Settles B, Craven M. An analysis of active learning strategies for sequence labeling tasks. In: Proc. 2008 Conference on Empirical Methods in Natural Language Processing. Computational Linguistics; 2008, p. 1070–9.

[26] Seung HS, Opper M, Sompolinsky H. Query by committee. In: Proc. 1992 Workshop on Computer Learning Theory (COLT). ACM; 1992, p. 287–94. doi:10.1145/130385.130417.

[27] Mamitsuka NAH. Query learning strategies using boosting and bagging. In: Proc. 1998 International Conference on Machine Learning (ICML); vol. 1. Morgan Kaufmann; 1998, p. 1–9.

[28] Qi GJ, Hua XS, Rui Y, Tang J, Zhang HJ. Two-dimensional multilabel active learning with an efficient online adaptation model for image classi-

fication. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 2009;31(10):1880–97. doi:10.1109/TPAMI.2008.218.

[29] Hoi SC, Jin R, Lyu MR. Large-scale text categorization by batch mode active learning. In: Proc. 2006 International Conference on World Wide Web. ACM; 2006, p. 633–42. doi:10.1145/1135777.1135870.

[30] Höferlin B, Netzel R, Höferlin M, Weiskopf D, Heidemann G. Inter-active learning of ad-hoc classifiers for video visual analytics. In: Proc. 2012 IEEE Conference on Visual Analytics Science and Technology (VAST). IEEE; 2012, p. 23–32. doi:10.1109/VAST.2012.6400492.

[31] Bernard J, Sessler D, Ruppert T, Davey J, Kuijper A, Kohlhammer J. User-based visual-interactive similarity definition for mixed data objects-concept and first implementation. Journal of WSCG 2014;22:329–38.

[32] Ritter C, Altenhofen C, Zeppelzauer M, Kuijper A, Schreck T, Bernard J. Personalized visual-interactive music classification. In: Proc. 2018 EuroVis Workshop on Visual Analytics (EuroVA). Wiley; 2018,doi:10.2312/eurova.20181109.

[33] Bernard J, Zeppelzauer M, Lehmann M, Müller M, Sedlmair M. Towards user-centered active learning algorithms. Computer Graphics Forum (CGF) 2018;37(3):121–32. doi:10.1111/cgf.13406.

[34] Cox MAA, Cox TF. Multidimensional Scaling. Springer. ISBN 3540330372; 2008, p. 315–47. doi:10.1007/978-3-540-33037-0_14.

[35] Shao L, Mahajan A, Schreck T, Lehmann DJ. Interactive regression lens for exploring scatter plots. Computer Graphics Forum (CGF) 2017;36(3):157–66. doi:10.1111/cgf.13176.

[36] Chegini M, Shao L, Gregor R, Lehmann DJ, Andrews K, Schreck T. Interactive visual exploration of local patterns in large scatterplot spaces. Computer Graphics Forum (CGF) 2018;37(3):99–109. doi:10.1111/cgf.13404.

[37] Netzel U, Vuong J, Engelke U, O'Donoghue S, Weiskopf D, Heinrich J. Comparative eye-tracking evaluation of scatterplots and parallel coordinates. Visual Informatics 2017;1(2):118–31. doi:10.1016/j.visinf.2017.11.001.

[38] DMandML. Github Repository; 2018. URL: https://github.com/TKnudsen/DMandML.

[39] Harrower M, Brewer CA. Colorbrewer. org: an online tool for selecting colour schemes for maps. The Cartographic Journal 2003;40(1):27–37. doi:10.1179/000870403235002042.

[40] Hund M, Böhm D, Sturm W, Sedlmair M, Schreck T, Ullrich T, et al. Visual analytics for concept exploration in subspaces of patient groups. Brain Informatics 2016;3(4):233–47. doi:10.1007/s40708-016-0043-5.

[41] Berger P, Chegini M, Schumann H, Tominski C. Integrated visualization of structure and attribute similarity of multivariate graphs. Poster at IEEE Conference on Information Visualization (InfoVis); 2018.

[42] Gleicher M, Albers D, Walker R, Jusufi I, Hansen CD, Roberts JC. Visual comparison for information visualization. Information Visualization 2011;10(4):289–309. doi:10.1177/1473871611416549.

[43] Behrisch M, Blumenschein M, Kim NW, Shao L, El-Assady M, Fuchs J, et al. Quality metrics for information visualization. Computer Graphics Forum (EuroVis State of The Art Report) 2018;37(3):625–62. doi:10.1111/cgf.13446.