EuroVis Workshop on Visual Analytics (2013)
M. Pohl and H. Schumann (Editors)

Visual Grouping - Follow the Leader!

Thomas Lammers,[1] Roel Vliegen,[1] Erik-Jan van der Linden,[1] and Huub van de Wetering[2]

[1]MagnaView, Eindhoven, The Netherlands
[2] Technische Universiteit Eindhoven, Eindhoven, The Netherlands

Abstract

We present an interactive system that provides users with automated techniques for grouping data, while shielding them from the technical aspects of these techniques. In our system users create visual data representations, called views, and choose a dataset for visualization in such views. From the resulting visualization and user actions on this visualization, the system derives the information that is used to automatically steer a grouping engine. Knowledge of data mining is not necessary; parameters and distance functions are automatically derived. In this easy to use system the user can efficiently create any grouping by incrementally manipulating groups with intuitive user actions. These actions allow the user to create, remove, and manipulate groups using a leader and follower metaphor. An implementation of the system has been created in a commercial data visualization tool.

1. Introduction

Systems that help analyze data are increasingly user-friendly, and have successfully found their way to users that have limited knowledge of visualization techniques and data analysis [Tab12, Qli12, bfrN12]. These users, typically domain experts, interactively design 'mini-applications' that support them in their domain tasks. To create visual representations suitable for these tasks, they select tables from databases and files, record sets from these tables, and attributes of these records. Fulfillment of a task may require grouping of data, which typically requires domain knowledge unavailable in the data. In this paper, we consider grouping as the addition of a new attribute to the dataset with values corresponding to the actual groups.

Users have easy access to visualization techniques in commercially available systems and are at the same time shielded from their technical aspects. A similar support level is not offered for grouping, whereas users could benefit a lot from readily available grouping operations, if they would not be bothered by technical aspects. Automated classification or clustering techniques alone may be insufficient for grouping; the domain knowledge of a user may be required. When grouping data, the eventual evaluation thus has to be made by the domain expert. The central question in this paper is: *Is it possible to develop an easy to use system that supports users in efficiently creating any grouping using automated grouping techniques, while completely shielding them from the details of these techniques?* We answer this question by building a grouping system in an existing data analysis tool, called MagnaView [bfrN12]. MagnaView allows the user to create interactive visual representations with two important properties. Firstly, these representations are generalized n-dimensional treemaps [VvWvdL06]. For instance, a matrix of stacked bar charts has three dimensions, matrix, bar chart, and stacks (see Figure 1). Each dimension is linked with one of the attributes in the data, e.g. the grouping attribute. Secondly, on the lowest level of representation, the records that are found in the dataset are always available to the users. *Entities* are the objects the user intends to group and each entity is a set of records (see Figure 5).

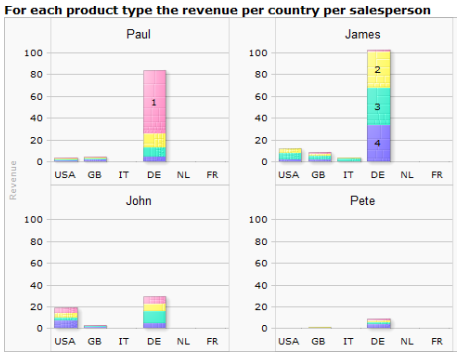


Figure 1: *Visualization with a 2x2 matrix of stacked bar charts with all involved records in the stacked parts.*

DOI: 10.2312/PE.EuroVAST.EuroVA13.001-005

2. Related Work

Clustering and classification are automated techniques in data mining that require users to input their domain knowledge by setting parameters in an unintuitive way. Interactive data mining does allow for interactive parameter tweaking. We discuss some of its variants below.

Interactive Clustering The most popular approach to interactive clustering is *constraint-based clustering*, e.g. with constraints stating that certain entities may or may not be in the same group [WCRS01]. In this context Cohn et al. [CCM03] use k-means [KR90] to group entities, and users can critique on the groups by adding constraints, after which k-means is re-run. The unstability of this grouping is reduced by Davidson et al. [DRE07]. DesJardins et al. [dMF07] implemented a system with constraints defined by dragging and dropping between groups. Basu et al. [BBM04] used active learning to reduce the number of constraints. Disadvantages of these methods are that they mostly rely on k-means, which cannot find complex structures in data, and that the grouping process is unstable. Contrarily, in our system any grouping can be created in an incremental and stable process.

Interactive Classification Interactive classification systems allow users to adapt the classification model by interacting with this model or with the data. There are two popular approaches. The first one [AEEK99, PD08, LS07, vdEvW11] visualizes the training data using decision trees. The other one lets users draw the data splits in scatter plots of the training data [TM03, LSG04]. Our system shields the user from such technical details of data mining algorithms and lets users freely choose visualizations of the grouping.

Interactive Grouping The target of interactive grouping systems is not to create a grouping that minimizes distances between groups, but to create any grouping the user desires. Usually, they create a classification model based on user interaction. In contrast with interactive classification, interactive grouping starts without training data. Basu et al. [BFDL10] propose a system that visualizes entities in a plane and allows users to create groups by dragging entities. Groups can be created and entities can be put into those groups together with similar entities. Entities which do not belong in a user defined group are put into a rest group. The system of Seifert et al. [SSG10] visualizes all entities using an "information landscape" visualization that shows the relation between entities and groups. The user can select entities and create a group from selected entities. Again, a classifier is trained in the background based on the user interaction. Chen and Lui [CL03] first use interactive clustering to create groups, which can then be manipulated by merging or splitting groups and dragging entities between them. Our system supports similar actions to interactively manipulate entities and groups, and does so in a user chosen visualization.

The visually controllable data mining method of Puolamaki at al [PPL10] requires that the involved parameters and models are visually representable. Our system hides both the data mining method and its parameters from the user.

Andrienko et al. [AAR*09] describe a system for visual clustering of trajectories. They require a distance function and a visual representation that are compatible. Our system computes a distance from the visual representation designed by the user and thus guarantees compatibility.

Finally, Hossain et al. [HOG*12] build a system with, as they say, a natural interface, in which users can critique on clustering results and incrementally build a clustering. Compared to our system their interface is less intuitive, since they do not, as we do, support direct manipulation.

Our system shares some basic techniques and interaction styles with existing work, but differs in at least two aspects: Firstly, it provides users with a visual analytics tool in which they can define their own visualizations. Secondly, the user is completely shielded from any of the technical aspects for grouping. Additionally, the system is easy-to-use: It uses a suitable metaphor, offers intuitive and direct actions, and allows the user to incrementally create any grouping.

3. Approach

In preparation of an interactive grouping process, the user loads data tables and selects records and attributes of interest, adds a grouping attribute, and creates a visual representation in which the grouping attribute is used for coloring (See Figure 3), or as a new division of the graphical representation (See Figure 7). Then the iterative process can take place, until the user positively evaluates the grouping. This process consists of the following steps (See Figure 2).

S1 The system suggests an entity for manual classification.
S2 The system visualizes the data.
S3 The user applies an action to change the grouping.
S4 The system re-groups the entities.

Using these steps a user can iteratively create the grouping he intends by applying actions in step S3. Typical actions are creating, resizing and splitting a group and promoting an entity. Nearest-neighbor classification is used in step S4 to re-group all entities: A group is represented by a set of entities called *leaders*. *Followers* are grouped into the group of the closest leader, if this leader is closer than a given distance r. Entities with no leader closer than r are placed in the *rest group*. In this way, promoting an entity, e.g. the entity suggested in step S1 by an active learning [Set09] module, to a new leader in a group adds all its followers to this group.

Figure 3 illustrates how a user created a grouping by applying several actions in step S3. The chosen visualization is a scatter plot with the entities as dots, colored by their grouping attribute. Figure 3a shows the initial situation where all entities are in the gray rest group. The user decides to split the rest group and chooses to do so into 4 groups. Hidden

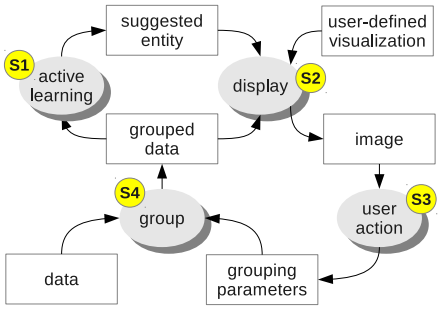
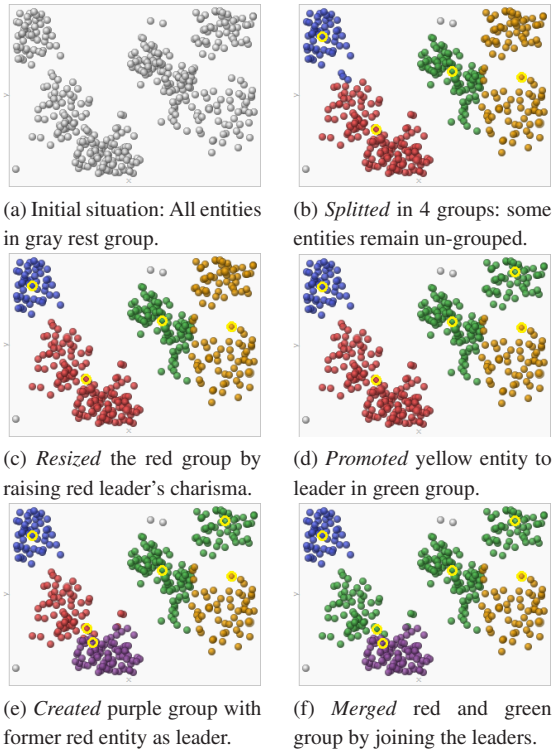


Figure 2: *The steps in our interactive grouping system. In S1 the system suggests a suitable candidate leader. In step S2 the grouped data and the suggestion are displayed according to a predefined view. In step S3 the user applies an action. And finally, in step S4 the data is re-grouped accordingly.*



(a) Initial situation: All entities in gray rest group.

(b) *Splitted* in 4 groups: some entities remain un-grouped.

(c) *Resized* the red group by raising red leader's charisma.

(d) *Promoted* yellow entity to leader in green group.

(e) *Created* purple group with former red entity as leader.

(f) *Merged* red and green group by joining the leaders.

Figure 3: *Grouping by user initiated actions in step S3: Groups are colored, leaders are highlighted.*

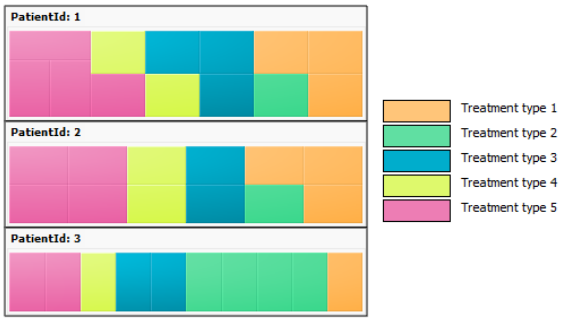


Figure 4: *3 patients entities and their treatment records.*

from the user, the system has applied k-means and assigned leaders in the new groups (see Figure 3b). In this case, some outliers remain in the rest group. In Figure 3c, after the user changed the *charisma* of the red leaders, entities have a shorter distance to those leaders and, consequently, the red group has grown to include some blue outliers. In Figure 3d the user has chosen to promote one of the top right yellow entities to leader in the green group, taking along some other yellow entities. In Figure 3e the user has created a new group with a former red entity as leader. In Figure 3f the red and green group have merged by joining their leaders.

Figure 3 also illustrates that showing the entities and their grouping in visualization step S2 is useful. While other systems visualize the entities using some default visualization method, we allow users to design their own visualizations. On the one hand, they can visualize the data in a way that best fits their domain knowledge. On the other hand, the user-created visualizations then reflect their domain knowledge, which allows incorporation of that knowledge in similarity measures used in classification and clustering.

4. Visualization based Distance measures

Clustering in step S3 and classification in step S4 both use a distance measure for entities. For these steps to work well together the same distance measure is used. This distance measure $D(e_1, e_2)$, for entities e_1 and e_2, may be any distance measure, including the one introduced here that is based on a visualization created by a user for visual comparison of entities. Such visualizations provide the following information that may be used in a distance measure: The visualized entities, the attributes used in the visualization, and the sizes of the graphical elements used to display an entity.

Let $A = \{a_1, \ldots, a_n\}$ be the subset of attributes used in a visualization to color or to group records from an entity. Let R be the set of all records in all entities in the visualization. The value of an attribute a of a record r is denoted by $r.a$ and the value set $\{r.a | r \in R\}$ is denoted by $R.a$. Our visually defined distance measure $d_{Vis}(e_1, e_2)$ compares the visualized size of entities e_1 and e_2 for all attribute-value-combinations with attributes in A. Let $V = R.a_1 \times \ldots \times R.a_n$ be the set of attribute-value combinations. The function d_{Vis} is then defined as follows:

$$d_{Vis}(e_1, e_2) = \sum_{v \in V} |S(e_1, v) - S(e_2, v)|$$

where $S(e, v)$ is the sum of the visualized size $s(r)$ of all records r in e for which $r{\downarrow}A := (r.a_1, \ldots, r.a_n)$ equals v. To re-

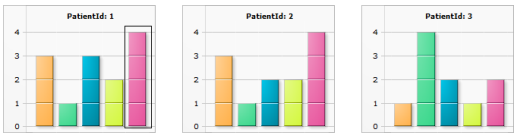


Figure 5: *Per entity (patient) the treatment records colored per treatment type. The 4 records of patient 1 with treatment type 5 are surrounded by a black border.*

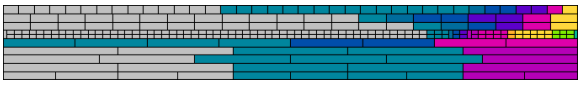


Figure 6: *9 patients and their treatments. Each patient is visualized as a horizontal bar. This bar is divided in treatments. Treatments are colored on treatment type.*

duce the amount of attribute value combinations, we bin numerical attribute values. Resulting discretization issues can be solved by computing $S(e,v)$ by weighing $s(r)$ with a suitable kernel evaluated at the distance between $r{\downarrow}A$ and v.

Figure 4 shows three patients, $p1$, $p2$, and $p3$, and their 13, 12, and 10 treatments, respectively. Treatment types are shown using color. In this case the set A of attributes only contains *treatment type* and V consists of the five treatment types. The sizes of all shown treatment records r are equal: $s(r) = 1$. We calculate per entity e per attribute value v the total size of all visualized records: for instance, for entity $e = p1$ and attribute value v=*treatment type 5* we get $S(v,e) = 4$ (See Figure 5). The distance of two patients is the sum of the absolute differences between the visualized size per treatment type; so, for instance, $d_{Vis}(p1,p2)=|3-3|+|1-1|+|3-2|+|2-2|+|4-4|=1$ and $d_{Vis}(p1,p3) = |3-1|+|1-4|+|3-2|+|2-1|+|4-2|=9$. As expected, when looking at Figure 4, patients 1 and 2 are more similar than patients 1 and 3.

5. Results

We show our system at work on a dataset with 1157 patient entities and 44173 treatment records. Our goal was to group patients that have had the same treatments. Figure 6 shows 9 sample patients, each represented by a horizontal bar in which each treatment is represented by a small rectangle colored and grouped by treatment type. The fourth patient from the top has had many treatments, many of which had the same type. The bottom 3 entities in Figure 6 represent patients with the same treatments, even though one has had more treatments of each type.

Figure 7 shows the groups which we created using our method. We can see 16 groups of patients. Some groups, like the first and second one, consist of patients that had only treatments of one type. Entities within a group are sorted

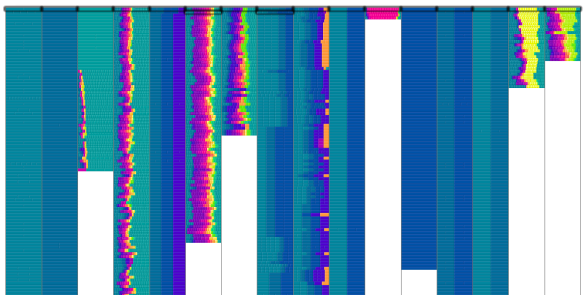


Figure 7: *16 groups of patients, leaders are highlighted. Patients are sorted per group based on distance to leaders. The groups are cut off at the bottom.*

with repect to similarity to a leader. As can be observed in the third group, where the leader is a patient with only one treatment type and patients with a higher percentage of other treatments, are positioned lower.

The dataset also contained the diagnosis treatment code (DBC) from which the costs of the treatment are deduced. In verifying grouping results with this code, we found that some patients were likely given the wrong DBC, possibly resulting in inaccurate billing.

6. Discussion and Conclusions

We propose a system for interactive grouping and realized it in a commercial visualization tool. Its users can manipulate groups with intuitive actions on entities that are visualized in user-defined visualizations. Since the user defined these visualizations to visually compare the entities, we derive distance measures from these visualizations for use by clustering and classification algorithms, whose technical details are hidden from the user. To help users understand our grouping process, we use the metaphor of leaders and followers. Active learning further improves the speed of grouping by suggesting entities for manual grouping.

In the current implementation the system easily handles 200,000 records. The system has no restrictions on the number of attributes. However, users are not likely going to use more than 4 attributes in their visualizations: suitability of the visual distance function for these cases needs further research. This distance may be defined for any visual representation for which the perceptual properties used for visual comparison, in our case mainly size, are quantifiable. Under that condition, new visualizations can be integrated.

During a limited user test the system turned out to be easy-to-use and efficient for creating the grouping the user desired. This leads us to conclude that the central question of this paper can be answered positively for the tested dataset: It is possible to build an easy-to-use and efficient grouping system that shields all technical aspects from the user. In practice, we find that the system also works with other datasets.

References

[AAR*09] ANDRIENKO G. L., ANDRIENKO N. V., RINZIVILLO S., NANNI M., PEDRESCHI D., GIANNOTTI F.: Interactive visual clustering of large collections of trajectories. In *IEEE VAST* (2009), pp. 3–10.

[AEEK99] ANKERST M., ELSEN C., ESTER M., KRIEGEL H.-P.: Visual classification: An interactive approach to decision tree construction. In *KDD* (1999), pp. 392–396.

[BBM04] BASU S., BANERJEE A., MOONEY R. J.: Active semi-supervision for pairwise constrained clustering. In *SIAM International Conference on Data Mining* (Apr. 2004), pp. 333–344.

[BFDL10] BASU S., FISHER D., DRUCKER S. M., LU H.: Assisting users with clustering tasks by combining metric learning and classification. In *AAAI* (2010).

[bfrN12] MAGNAVIEW:. http://www.magnaview.com, 2012.

[CCM03] COHN D., CARUANA R., MCCALLUM A.: Semi-supervised clustering with user feedback. *Constrained Clustering Advances in Algorithms Theory and Applications 4*, 1 (2003).

[CL03] CHEN K., LIU L.: Validating and refining clusters via visual rendering. In *Proceedings of the Third IEEE International Conference on Data Mining* (Washington, DC, USA, 2003), ICDM '03, IEEE Computer Society, pp. 501–.

[dMF07] DESJARDINS M., MACGLASHAN J., FERRAIOLI J.: Interactive visual clustering. In *IUI* (2007), pp. 361–364.

[DRE07] DAVIDSON I., RAVI S. S., ESTER M.: Efficient incremental constrained clustering. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (New York, NY, USA, 2007), KDD '07, ACM, pp. 240–249.

[HOG*12] HOSSAIN M. S., OJILI P. K. R., GRIMM C., MUELLER R., WATSON L. T., RAMAKRISHNAN N.: Scatter/gather clustering: Flexibly incorporating user feedback to steer clustering results. *IEEE Trans. Vis. Comput. Graph. 18*, 12 (2012), 2829–2838.

[KR90] KAUFMAN L., ROUSSEEUW P. J.: *Finding groups in data: an introduction to cluster analysis*. John Wiley and Sons, New York, 1990.

[LS07] LIU Y., SALVENDY G.: Interactive visual decision tree classification. In *Proceedings of the 12th international conference on Human-computer interaction: interaction platforms and techniques* (Berlin, Heidelberg, 2007), HCI'07, Springer-Verlag, pp. 92–105.

[LSG04] LIU D., SPRAGUE A. P., GRAY J. G.: Polycluster: an interactive visualization approach to construct classification rules. In *ICMLA* (2004), pp. 280–287.

[PD08] POULET F., DO T.-N.: Interactive decision tree construction for interval and taxonomical data. In *Visual Data Mining - Theory, Techniques and Tools for Visual Analytics*, Simoff S. J., Böhlen M. H., Mazeika A., (Eds.). Springer-Verlag, Berlin, Heidelberg, 2008, pp. 123–135.

[PPL10] PUOLAMAKI K., PAPAPETROU P., LIJFFIJT J.: Visually controllable data mining methods. In *Proceedings of the 2010 IEEE International Conference on Data Mining Workshops* (Washington, DC, USA, 2010), ICDMW '10, IEEE Computer Society, pp. 409–417.

[Qli12] QLIKVIEW:. http://www.qlikview.com, 2012.

[Set09] SETTLES B.: Active learning literature survey, 2009. Computer Science Technical Report 1648, University of Wisconsin–Madison.

[SSG10] SEIFERT C., SABOL V., GRANITZER M.: Classifier hypothesis generation using visual analysis methods. In *NDT (1)* (2010), pp. 98–111.

[Tab12] TABLEAU SOFTWARE:. http://www.tableausoftware.com, 2012.

[TM03] TEOH S. T., MA K.-L.: Starclass: Interactive visual classification using star coordinates. In *SDM* (2003).

[vdEvW11] VAN DEN ELZEN S., VAN WIJK J. J.: Baobabview: Interactive construction and analysis of decision trees. In *Proceedings IEEE Symposium on Visual Analytics Science and Technology.* (2011), pp. 252–160.

[VvWvdL06] VLIEGEN R., VAN WIJK J. J., VAN DER LINDEN E.-J.: Visualizing business data with generalized treemaps. *Transactions on Visualization and Computer Graphics 12*, 5 (2006), 789–796.

[WCRS01] WAGSTAFF K., CARDIE C., ROGERS S., SCHRÖDL S.: Constrained k-means clustering with background knowledge. In *ICML* (2001), pp. 577–584.