EG UK Computer Graphics & Visual Computing (2016)
Cagatay Turkay and Tao R. Wan (Editors)

MLCut: Exploring Multi-Level Cuts in Dendrograms for Biological Data

A. Vogogias[1], J. Kennedy[1], D. Archambault[2], V. A. Smith[3] and H. Currant[3]

[1]Edinburgh Napier University, School of Computing, United Kingdom
[2]Swansea University, Department of Computer Science, United Kingdom
[3]University of St Andrews, School of Biology, United Kingdom

Abstract

Choosing a single similarity threshold for cutting dendrograms is not sufficient for performing hierarchical clustering analysis of heterogeneous data sets. In addition, alternative automated or semi-automated methods that cut dendrograms in multiple levels make assumptions about the data in hand. In an attempt to help the user to find patterns in the data and resolve ambiguities in cluster assignments, we developed MLCut: a tool that provides visual support for exploring dendrograms of heterogeneous data sets in different levels of detail. The interactive exploration of the dendrogram is coordinated with a representation of the original data, shown as parallel coordinates. The tool supports three analysis steps. Firstly, a single-height similarity threshold can be applied using a dynamic slider to identify the main clusters. Secondly, a distinctiveness threshold can be applied using a second dynamic slider to identify "weak-edges" that indicate heterogeneity within clusters. Thirdly, the user can drill-down to further explore the dendrogram structure - always in relation to the original data - and cut the branches of the tree at multiple levels. Interactive drill-down is supported using mouse events such as hovering, pointing and clicking on elements of the dendrogram. Two prototypes of this tool have been developed in collaboration with a group of biologists for analysing their own data sets. We found that enabling the users to cut the tree at multiple levels, while viewing the effect in the original data, is a promising method for clustering which could lead to scientific discoveries.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Viewing algorithms—H.3.3 [Information Search and Retrieval]: Clustering—Information filtering

1. Introduction

Hierarchical clustering (HC) algorithms are used in many applications for "grouping" data records into a number of non-overlapping sets (*i.e.* clusters). Those algorithms take as input a distance matrix with estimated pairwise dissimilarity scores between all data records. Dissimilarities between records are calculated using a metric or measure which is assumed to be appropriate for the intended type of analysis. The output produced is a simplified hierarchical structure, known as the *dendrogram*, which encapsulates the rationale followed by the HC algorithm. For instance, agglomerative HC algorithms repeatedly merge pairwise data into more abstract entities (*i.e.* dissimilarity levels), until the point that all records are merged into a single group: the root of the tree (Figure 1). While overlapping is not permitted between clusters in HC analysis, often there is ambiguity related to whether a cluster of records is distinct, or it is part of a larger cluster (*i.e.* nested cluster). This ambiguity is more evident in heterogeneous data sets.

The challenge for the user is to explore the different clustering scenarios and identify groups of similar records (*i.e.* patterns) in the data set. For data sets which consist of homogeneous subsets, deciding a single similarity threshold, which cuts the tree at a uniform height, is usually sufficient. However, for larger dendrograms, which often consist of heterogeneous subsets, a more effective approach is needed. A heuristic has been suggested by Langfelder *et al.* [LZH08] that cuts the branches of the dendrogram in different levels based on their shape. However, heuristic approaches are rarely optimal because they can not capture all the pattern variations which could be observed in data sets. Therefore, a better approach would be to first explore the original data in coordination with the dendrogram and then decide the clustering. To support

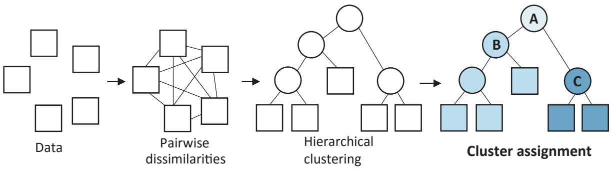


Figure 1: *The flow of constructing the dendrogram.*

this, an effective visualisation approach would enable the user to explore different clustering scenarios by providing different levels of detail. At the highest level, a view of the clusters as part of the whole dendrogram should be supported. At the lowest level of detail, the original data should be visible and linked to their cluster assignment. In the dendrogram the original data records are represented by the leaves. In practice, the user should be able to see the whole dendrogram and at the same time explore different clustering scenarios by viewing details in the original data. During the exploration process, the user should be able to test different clustering configurations by selecting (*i.e.* cutting) branches on demand, as this is shown in (Figure 2).

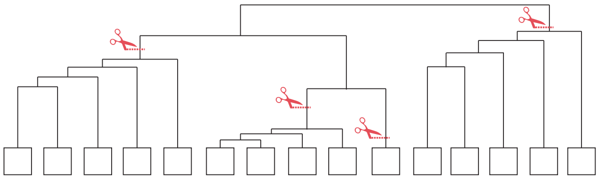


Figure 2: *Multi-level cuts in a heterogeneous dendrogram. The red icons indicate four locally applied similarity thresholds which cut the tree in four branches that form the same number of non-overlapping clusters. This clustering scenario could not be achieved using any single-height similarity threshold.*

In the real world, there is no *"one-size-fits-all"* solution and it is common to ignore special characteristics of clusters [KK99]. Within the same data set some clusters may be dense (low dissimilarity), while some others may be sparse (high dissimilarity). For instance, biologically associated genes may follow a similar expression pattern during the whole experiment, or only for a time period as reported by Mahanta *et al.* [MABK11] and Craig *et al.* [CCKK12]. Therefore, a *"human in the loop"* is needed to visually explore the dendrogram in different levels of detail and select potential sub-clusters manually [SM11]. The idea of *drilling-down* to see more detail in the data is common to many visual analysis tools. Similar steerable approaches have been investigated in the past for exploring graph structures, as in Archambault *et al.* [AMA08] and in Abello *et al.* [AvHK06].

We have developed an interactive tool that enables the user to manually select clusters by applying multi-level cuts on demand. There are two types of thresholds: *global* that applies to the whole dendrogram and *local* that only applies to parts of the tree, such as selected branches of interest, enabling a more finely-grained exploration. In addition, the interactive exploration of the dendrogram is coordinated with a representation of the original data, shown as *parallel coordinates*. The analysis process involves three steps. Firstly, a single-height similarity threshold can be applied using a dynamic slider to identify the main clusters. Secondly, a distinctiveness threshold can be applied using a second dynamic slider to identify *"weak-edges"* that indicate heterogeneity within clusters. Thirdly, the user can drill-down to further explore the dendrogram structure - always in relation to the original data - and cut the branches of the tree at multiple levels. This is an important step for detecting nested clusters and outliers. Interactive drilling-down is supported using mouse events such as hovering, pointing and clicking on elements of the dendrogram. Our tool follows a synergistic approach that combines the strengths of HC algorithms with the ability of humans to visually detect patterns and anomalies in the data.

An earlier version of the tool has been used for allocating single nucleotide polymorphisms (SNPs) to chromosomes of tetraploid species. A second prototype has been developed in collaboration with a group of biologists for finding patterns in a gene expression data set with short time-series.

2. Previous Work

There are several tools which perform clustering analysis, but only a few of them support visual analysis. Even fewer provide interactive exploration capabilities of the clusters in different levels of detail. However, increasingly more of these tools recognise the importance of visual interaction for performing clustering analysis. This relatively new paradigm is founded on the idea that expert users are capable of steering the analysis to produce more successful results [NYO*12]. The actions of the users are often driven by tacit knowledge which can not be easily encoded to become part of an algorithm. Therefore, involving a human in the loop for taking decisions and for guiding the analysis is essential.

Visual support tools have been always used in the analysis of biological data. An evaluation of microarray visualisation tools has been presented by Saraiya *et al.* [SND04] and a more recent survey has been presented by Pavlopoulos *et al.* [PMP*15]. Specialised tools for microarray data analysis often incorporate visualisation features for different analysis tasks, including HC analysis. *Chipster* [KTH*11] and *Mayday* [BSN10] are two of the most complete open source microarray data analysis platforms. Due to the importance of time-course gene expression data, there is also a number of tools for clustering such data sets in particular. *STEM* [EBJ06] is a software tool for automatic profiling and clustering of short time-series data. A more flexible and user-driven approach, in matters of statistical analysis capabilities, is provided by *PESTS* [SM11]. All of those tools support some visualisation features for HC analysis, but they provide little or no support for interactive exploration of the data.

There are several tools and techniques for visualising trees, but most of them do not focus on the HC analysis task addressed in this paper. For instance, *TreeJuxtaposer* [MGT*03] is a tool designed for biologists who want to perform structural comparison of large trees, such as phylogenetic trees, which look similar to dendrograms. *MizBee* [MMP09] uses three views that correspond to three levels of detail in order to provide support for comparing whole genomes. Both tools can be used to compare different data sets. However, our task is different as we focus on the challenge of finding patterns in the original data, through the interactive exploration of a single dendrogram. A technique presented in Chen *et al.* [CMP09] uses a uniform threshold to provide improved visibility by simplifying the dendrogram representation. This is a useful technique for summarising the dendrogram in a selected level of detail and making it fit in smaller displays. However, it does not provide support for multi-level cuts or data exploration at multiple levels.

The most similar to our visualisation tool is the *Hierarchical Clustering Explorer (HCE)* [SS02]. It has been designed for supporting interactive genomic microarray data analysis. It provides a dendrogram linked to a heatmap. It supports dynamic querying using a minimum similarity bar, which specifies a single similarity threshold in which the dendrogram is cut. In contrast to other tools, HCE provides interactivity and it is still a powerful tool for hierarchical clustering analysis. HCE deals with the same task as our tool, but it does not support multi-level cuts. Our approach enables the user to simultaneously look at different levels of detail in multiple locations of the dendrogram. Moreover, this can be achieved without missing the *"big picture"* (*i.e.* a view to the whole dendrogram).

To our knowledge no other visual support tool focuses on applying multi-level cuts to target the problem of ambiguity in heterogeneous data during the HC analysis. However, there are tools that provide this kind of visual support for exploring the data in different levels of abstraction for other clustering algorithms. For instance, *Spark* [NYO*12] provides two views, one in the level of clusters (found by *k-means*) and one in the level of regions within clusters. We follow a very similar approach. However, we mostly focus on visualising the output of hierarchical clustering (HC) algorithms rather than *k-means*. *Spark* does not directly address the particular challenges related to HC analysis (*i.e.* exploring the dendrogram structure in coordination with the original data, before deciding the clustering).

On the other hand, fully automated approaches that implement the idea of multi-level cuts, such as the *Dynamic Tree Cut (DTC)* [LZH08], use heuristic criteria, which are not unique and produce different clustering results. Those criteria are tailored in identifying pre-determined shapes or patterns in the dendrogram. Therefore, they do not provide any exploration support for finding new patterns in the data. Semi-supervised approaches such as the ones presented by Dotan-Cohen *et al.* [DCMK07], Navlakha *et al.* [NWN*09] and in *HCsnip* [OAMvdW15] integrate prior knowledge into the algorithm in order to improve results. The configuration can not be generalised for unstructured data sets without assuming any background knowledge related to them. Hence, the solution is based on assumptions about the specific data in hand. Our approach is unsupervised and it aims at providing interactive visual support to users who want to explore their data, based on their own tacit knowledge and not on pre-defined assumptions.

3. Design

The design of this tool was inspired by a real world problem, which often occurs in biology. The problem can be summarised as the analysis task of finding structure (*i.e.* patterns or other elements of interest) in heterogeneous data sets. The problem is common to all unsupervised methods that deal with the analysis of multidimensional data. However, the scope of this paper is more specific. It focuses on the challenge of exploring biological data through HC analysis. In particular, the aim was to apply HC in common types of genetic data, such as single nucleotide polymorphisms (SNPs) and gene expression time series. For HC analysis, elements of interest would be: distinct clusters, nested clusters and outliers. In order to explore those elements of interest more effectively, we followed an approach which would enable the user to cut the dendrogram in multiple levels and see the effect in the original data. Thus, our focus was on designing a visualisation system which would help solve the underlying biological - HC analysis - problem, by enabling a more effective exploration of the dendrogram and the original data.

Most of the design decisions were made after consulting the end users. However, some of the design choices were based on studies about human perception and cognition. The visual encoding was based on principles found in MacKinlay [Mac86], while the colour palette was created using *ColorBrewer* [HB03].

We followed an iterative process of continuously refining design guidelines and evaluating results, in order to improve the system. Two main development cycles took place, which resulted in the development of two prototypes. Each of those prototypes was used by different research groups for analysing different types of data. The design of the first prototype was focused on the challenge of creating an effective dendrogram representation, while in the second prototype the focus was on linking the dendrogram with the original data. Multiple iterations took place for developing each of the prototypes. During the development of the second prototype we followed the methodology of the *nested model* [Mun09]. After several meetings with end users, biological analysis tasks have been clarified and mapped to visualisation tasks. Details about design decisions related to visual encoding and user interface (UI) controls are explained in the next sections which describe the two main iterations for developing this tool.

3.1. Requirements

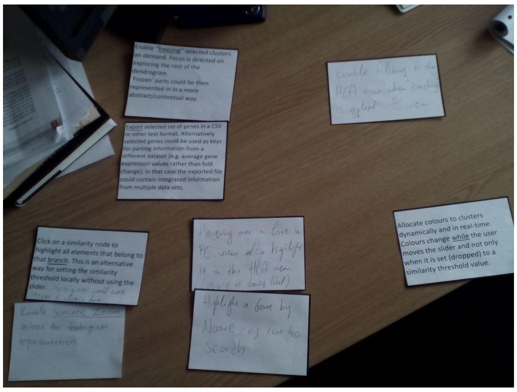
User requirements have been captured during informal conversations with end users. The initial design of this tool has been developed after a series of discussions with one expert user, who is a senior statistician specialised in computational methods for analysing biological data. Several design decisions were made during those unstructured discussions, which took place during the first iteration of developing this tool. The final design has been developed during the second iteration, after involving two more users. The first is an expert user specialised in computational biology and the second is a biology graduate with experience in computational methods. These two users are co-authors of this paper.

Our collaborators were interested in analysing short time-series gene expression data in order to extract modules or features which could be used for constructing a network model. In their research, they commonly use HC algorithms to find clusters of genes (*modules*) that follow a similar temporal pattern and then aggregate them or select a representative (*feature*) to become one of the variables in the final model.

During our first meetings the users were asked to view different examples of visualisation techniques and choose the ones that seemed more useful for representing their data. Thus, different techniques were informally evaluated in matters of their relevance to users' tasks. Those that appeared more interesting would be explained in more detail. Further discussions about the biological analysis problem would lead to the clarification of user requirements and the formation of more technical specifications.

Except from unstructured discussions, we also performed a *card sorting* session (Figure 3) following the three steps of *preparation*, *execution* and *analysis*, as described by Sakai *et al.* [SA15]. This session not only helped in characterising tasks in the biological domain, but also helped in stimulating the creative process for articulating and prioritising existing requirements. In summary, we found that the design of the tool should support the following:

- an effective dendrogram representation that scales well for large data sets
- an effective representation of the original data, in coordination with the dendrogram representation
- the ability to interactively explore the dendrogram and the representation of the original data, in different levels of detail
- the ability to maintain multiple cluster selections on display during the exploration process, without loosing a view to the whole data set
- the ability to export selected clusters so that they could be used for further analysis



Enable "freezing" selected clusters on demand. Focus is directed on exploring the rest of the dendrogram.
'Frozen' parts could be then represented in in a more abstract/contextual way.

Enable filtering to the HCA view when "brushing" is applied to PC view.

Export selected set of genes in a CSV or other text format. Alternatively selected genes could be used as keys for parsing information from a different dataset (e.g. average gene expression values rather than fold change). In that case the exported file could contain integrated information from multiple data sets.

Click on a similarity node to highlight all elements that belong to that branch. This is an alternative way for setting the similarity threshold locally without using the slider. Hovering over would work better + [illegible]

Hovering over a line in PC view also highlights it in the HCA view. (Clarity of drawing label)

Allocate colours to clusters dynamically and in real-time. Colours change while the user moves the slider and not only when it is set (dropped) to a similarity threshold value.

Enable semantic zoom in/out the dendrogram representation.

Highlight a Gene by Name -> eg text box Search

Figure 3: *A picture taken after a card sorting session. Both existing and new requirements were grouped based on their relevance and prioritised based on their importance.*

4. First Prototype

Modern sequencing technologies enable thousands of single nucleotide polymorphisms (SNPs) to be measured in genetic mapping populations. The first step in a genetic linkage analysis is to cluster the SNPs into separate chromosomal groups so that SNPs in different groups are inherited independently.

The tool was used to partition over 5000 SNPs measured on 190 offspring in a cross between two tetraploid potato lines. The distance metric between each pair of SNPs was estimated from the significance of a chi-square test for independence [LHB*01], and average-linkage was used as HC algorithm.

The users found that the main clusters agree well with position information from the sequenced potato genome. Detail within the clusters shows SNPs located on the different homologous chromosomes within each linkage groups. Full details of the genetic mapping are given in Hackett *et al.* [HMB13].

For this analysis only the first prototype of this visualisation component has been used. The prototype has been incorporated into a larger software package, used for the analysis of genetic data in tetraploid species called *TetraploidMap* [HMBL07].

4.1. Design decisions and UI controls

During the development of the first prototype, a series of design decisions were taken. Data records, which are leaves in the dendrogram, are represented as rectangles. Dissimilarity scores, which are always intermediate branch nodes, are represented as circles of diameters proportional to their value. The first prototype only supported a dendrogram view, while the original SNP marker data were shown in a tabular format within the tool.

There are several basic kinds of tree representations such as node-link, icicle, radial, concentric circles, nested circles, treemap and indented outline which are described in McGuffin *et al.* [MR10]. In one of our first attempts to visualise the dendrogram, we implemented a top-down node-link tree representation. In this representation the length of the branches becomes more evident (Figure 4). This dendrogram layout was found suitable for identifying outliers and clusters based on the length or shape of their branches. However, it was found that this representation did not address ambiguities in the data sufficiently and also it did not scale well for larger data sets.

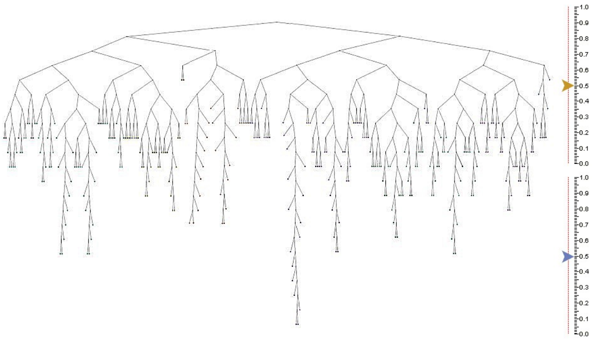


Figure 4: *Dendrogram displayed in the top down node-link tree layout.*

In the typical top-down dendrogram representation (Figure 2), each edge length is proportional to the overall distinctiveness score of the branch it connects with the rest of the tree. However, for large dendrograms, this convention does not always produce intelligible results. This is mainly because, above 400 leaves approximately, the size of the hierarchical structure is too large to browse. Considering the limitations of a standard monitor display and that edge length is a relative metric (therefore, not easily comparable with the naked eye), it is unrealistic to expect users to visually compare edge lengths in large dendrograms. In our approach, we adopted a space efficient radial layout and we quantify and control the property of distinctiveness using a *dynamic slider* [AS94].

Whilst Burch *et al.* [BKH*11] shows that radial node-link (NL) representations take longer to read than top-down NL views, at least up to 500 nodes, radial NL layouts are more space efficient than top-down NL representations [MR10], and also more efficient than left-to-right NL representations if labels are not shown, as is the case here. The radial layout utilizes better the space available for displaying the data and limits the use of scrollbars. Because the hierarchy is wide, the top down tree layout in Figure 4 would not be visible in a single view without scrollbars. Another option is to zoom out to a scale where the dendrogram was entirely visible, but the colour of the nodes, used to encode the clusters, would not be distinguishable. For the same task a radial dendrogram is much more compact, allowing the different colours to be distinguished, while the whole data set could fit in a standard monitor display.

4.1.1. Dynamic sliders

A global similarity threshold can be applied using a dynamic slider. Branches that belong to the same cluster get the same colour. In Figure 5, snapshots I and II demonstrate the process of merging a large number of smaller clusters into three main groups by moving the top slider, which controls the similarity threshold. This interaction is useful for identifying the main clusters and also for testing the different scenarios that the single-height approach can investigate.

A second dynamic slider can be used for identifying nested clusters, which appear considerably more distinctive than the larger ones in which they often belong to. Hence, the second slider sets the maximum allowed similarity distance between a parent main cluster and a child sub-cluster. Distinctive *"weak-edges"* between neighbouring nodes are *thicker*, *dashed* and coloured *red*. Experimenting with different *distinctiveness* thresholds, can help in identifying possible outliers and nested clusters. Figure 5 III demonstrates the identification of a distinct nested cluster (shown in *black*) by moving the bottom slider, which controls the *distinctiveness* threshold.

5. Second Prototype

In the context of gene expression analysis, HC algorithms are used for data partitioning and variable selection. When analysing gene expression data using Bayesian network (BN) models, a subset of variables (from thousands) must be selected for inclusion in the network [MSF*09]. The mapped visualisation objective to this analysis goal was to develop a visualisation approach which would help researchers use HC algorithms for this variable selection by enabling them to explore, view, select and finally export aggregated clusters of genes, or single representative genes, from groups of similar genes. The clusters, or representative single genes selected, would become the variables represented as nodes in the BN model.

Towards this analysis goal, a prototype of this tool has been developed in collaboration with a small group of computational biologists for analysing a gene expression data set with short time-series. For such data sets each of the clusters corresponds to a distinct temporal profile, or pattern [WWLC08]. A draft of this prototype was presented in Vogogias *et al.* [VKA16].

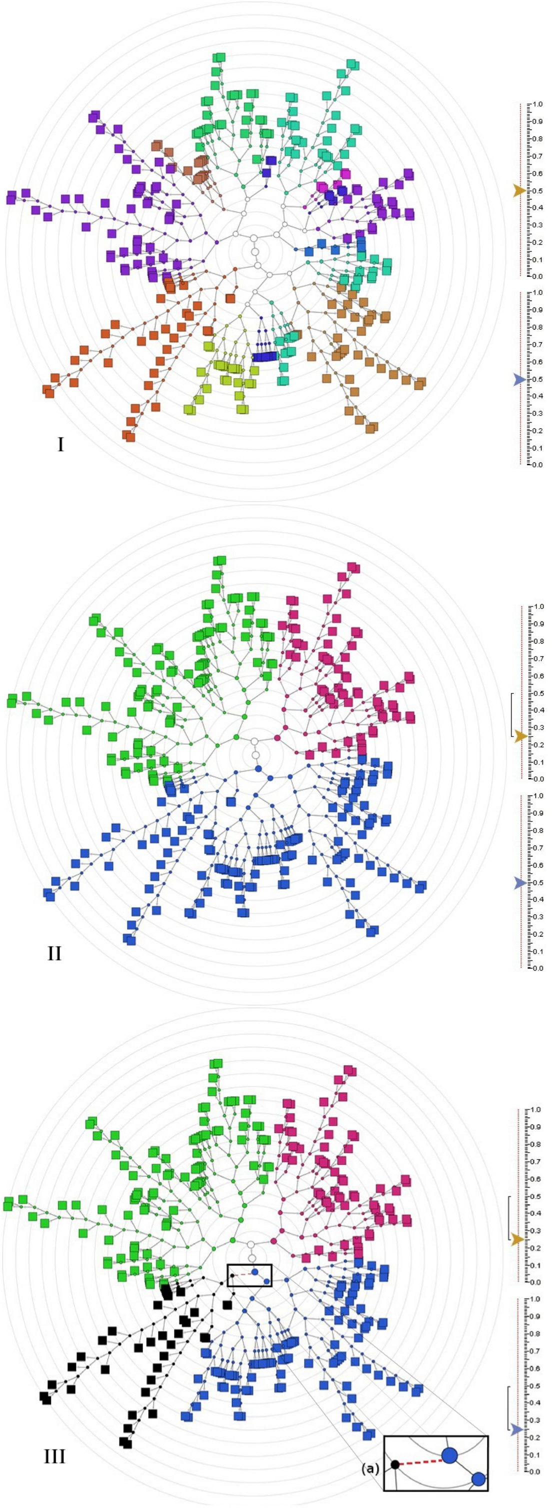


Figure 5: *Dynamic query sliders in use. The top slider in II sets the similarity threshold and the bottom slider in III sets the distinctiveness threshold.*

5.1. Coordinated views

During HC, information is extracted from the original data. However, depending on users' decisions, some information is either filtered, or it is aggregated to form more abstract entities (*i.e.* dissimilarity levels), which are represented as intermediate nodes in the dendrogram. If the wrong combination of distance metric/measure, algorithm and its parameters is used, important information could be lost and results could be even misleading. Therefore, our collaborators also asked to provide a representation of the original data, in coordination with an effective representation of the dendrogram. In this way, the users would be able to interact with the dendrogram while seeing the effect their choices have in the original data.

In our effort to satisfy this requirement, we developed the following design approach. The user interface (UI) is composed of two linked view components (Figure 6). The top view constitutes a radial representation of the dendrogram like the one used in the first prototype, while the bottom view is a representation of the original data using *parallel coordinates* [ID91].

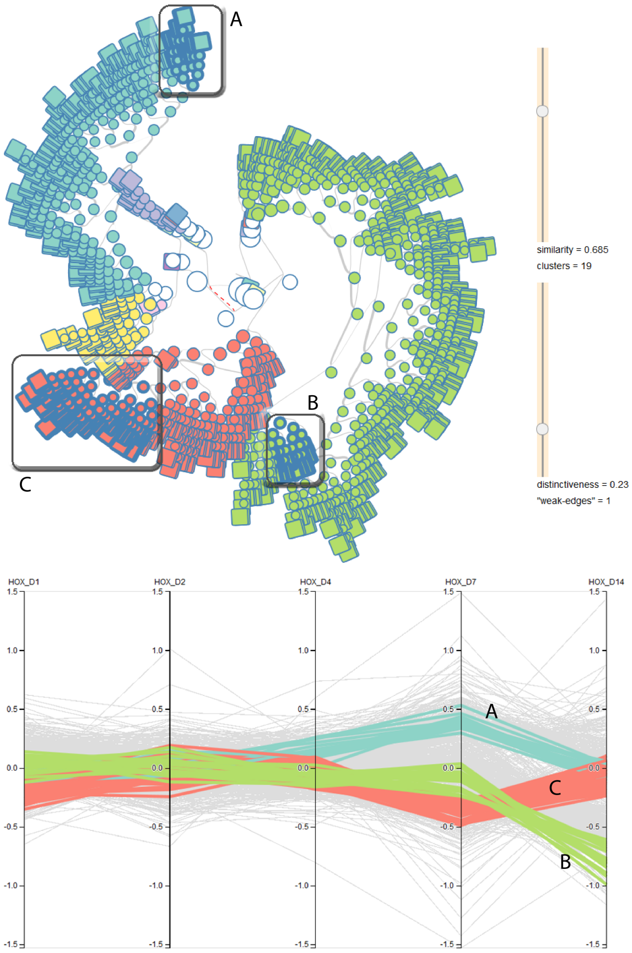


Figure 6: *Three sub-clusters of genes (A, B and C) that exhibit distinctive time patterns. Each sub-cluster belongs to a larger main cluster, visually encoded using colour.*

Each data record, which is is represented as a rectangle in the dendrogram (*i.e. top*) view, is linked to a line in the parallel coordinates (*i.e. bottom*) view. Data records are normalised and every axis in the parallel coordinates is scaled to the same minimum and maximum values, in order to enable comparisons. The user can explore main clustering assignments using the two sliders and also interact with the branches of the dendrogram to explore the effect of potential multi-level cuts, shown in the parallel coordinates view. The interaction is done by hovering over the circles of intermediate branch nodes. This gives a real-time preview of the effect the branch-cut would have in the original data. Clicking on a circle selects the whole branch including its leaves. Previewing and selecting branches interactively can be done in any level of detail: from the whole tree until a single leaf. This flexibility enables the exploration of potential sub-clusters within the main clusters identified using the sliders. Selected branches are highlighted with thicker borders at the top view and with thicker lines at the bottom view as it is shown in Figure 6.

In order to reduce the visual clutter at the bottom view, records that do not belong to any of the selected branches are shown in *light gray*. The top view supports most of the interactivity and the colour encoding is preserved in order to enable the further exploration of the dendrogram. Moreover, a record of particular interest could be further highlighted by double clicking on its mapped rectangle at the top view. Finally, each selected cluster or sub-cluster could be exported as a comma separated values (CSV) file.

In the first prototype, possible outliers and nested clusters found using the distinctiveness slider, were shown in *black*. However, this encoding changed in the final prototype because using the same colour was confusing when multiple *"weak-edges"* were found close to each other in the dendrogram. Therefore it was decided to retain the colour encoding that characterises the parent cluster and only show the *"weak-edges"* using thick, dashed, red lines.

5.2. Evaluation

Usability testing has been done informally for this tool during and after the development of the second software prototype. A real usage scenario took place in which a gene expression data set with short time-series was explored. The data set consist of the fold change of 800 differentially expressed genes in five time points and it can be found in the *Gene Expression Omnibus (GEO)* [EDL02] repository with accession number *GSE49577* [KLHS14].

Initially, different distance measures have been used for calculating pairwise dissimilarities between time series such as: euclidean distance, autocorrelation coefficient and dynamic time warping. Also different agglomerative HC algorithms have been tested using the *TSclust* [MV14] package in R. The combination of euclidean distance with an average-linkage HC algorithm has been selected as the best option for the task.

Using our tool, we managed to find three distinctive temporal profiles of late gene expression (Figure 6). This was achieved by interactively exploring the branches for potential sub-clusters, and eventually by cutting the dendrogram in multiple levels. The difference in gene expression patterns occurs between the third and the fifth parallel coordinates. Gene expression in the cluster shown in Figure 6*A* first increases and then decreases, while in Figure 6*C* the opposite happens (first decreases and then increases). Gene expression in the cluster shown in Figure 6*B* remains stable between the third and fourth time points and decreases after that. These patterns not only agree well with clusters related to late gene expression as reported in Koussounadis *et al.* [KLHS14], but also provide a more refined cluster assignment scenario. In addition, using our tool's ability to export genes in selected clusters, BNs were created from cluster means and three gene clusters identified which may be potentially prognostic.

The case study demonstrated the benefits of this tool in practice and helped refine the prototype. Some additional feedback was given through the card sorting session, originally completed for capturing and clarifying user requirements. However, some of the design choices were based on known perceptual principles and not on users' feedback. Finally, anecdotal feedback was given through emails and also verbally during our discussions with the users.

One of the users wrote: *"I cannot seem to download the gene list by clicking on the genes.. It is still working with the HOX data set but for some reason it will not let me in the OV data sets.. Thank you for all of your help so far and other than this the tools are excellent!"*.

Another user wrote: *"I really like how the tool lets you see both the expression lines and clusters, and how this changes as you change the clustering. I can really see the applications for being about to chose sub-clusters based on visual match rather than having to blindly slice the tree at one level only, and am looking forward to seeing what we can discover using this method of clustering the data"*.

6. Implementation

The first prototype was developed in Java. It does not include an implementation of coordinated views between the dendrogram and the original data but it supports semantic zooming and a top-down tree layout for the dendrogram. The second and final prototype was written in JavaScript (using *D3* [BOH11]). The implementation lacks semantic zooming capabilities and the top-down tree layout representation for the dendrogram, but it includes the implementation of coordinated views between the parallel coordinates and the dendrogram. Back-end data pre-processing and HC analysis was done using R.

7. Limitations and Future Work

The proposed visualisation method has potential applications in many clustering challenges in high-dimensional molecular biology. It can be applied to any type of HC algorithms and could be extended to support hybrid methods for HC [CT06]. Therefore, one of the future challenges would be to identify requirements and prioritise technical specifications which would guide the further development of this approach to a more generic tool for clustering.

Also a future challenge would be to deal with scalability issues for larger time series or multidimensional data sets. Currently this tool carries the inherent limitations of the parallel coordinates technique. However, the scalability of the current implementation could be improved by using edge bundling to reduce clutter and occlusion in large data sets. Also the tool could provide enhanced functionality for merging and splitting clusters on demand by supporting aggregated and expanded views. For example, support for summarizing/filtering branches that are of less interest. Additionally, a more effective algorithm for allocating colours to clusters when moving the similarity threshold slider could be developed. Finally, a more formal usability study could take place to further evaluate this tool.

8. Conclusion

HC is a common unsupervised method for analysing biological data. For heterogeneous data sets there are ambiguities in the way that data records could be allocated to clusters, which can not be resolved algorithmically. Therefore, a synergistic approach for exploring clustering scenarios in the context of the original data records, the output of the HC algorithm and users' tacit knowledge is required. We developed MLCut, an interactive software tool that enables a visual exploration approach for performing hierarchical clustering analysis. Human intervention is not sufficient when it is limited in choosing a similarity threshold for cutting the dendrogram at a single level. With MLCut, the user can cut the dendrogram at multiple levels and see the effect in the original data. Our research has shown that this method provides more transparency and confidence to the process of assigning data records to clusters and could lead to scientific discoveries.

9. Acknowledgements

The authors wish to thank Dr Christine Hackett of Biomathematics and Statistics Scotland for introducing us to the challenge of hierarchical clustering, Dr Martin Graham for helping with an earlier version of this paper, and Judit Urgelles for helping in curating the figures.

References

[AMA08] Archambault D., Munzner T., Auber D.: Grouseflocks: Steerable exploration of graph hierarchy space. *IEEE Transactions on Visualization and Computer Graphics 14*, 4 (2008), 900–913. doi:10.1109/tvcg.2008.34. 2

[AS94] Ahlberg C., Shneiderman B.: Visual information seeking: tight coupling of dynamic query filters with starfield displays. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (1994), ACM, pp. 313–317. doi:10.1145/191666.191775. 4

[AvHK06] Abello J., van Ham F., Krishnan N.: Ask-graphview: A large scale graph visualization system. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006), 669–676. doi:10.1109/tvcg.2006.120. 2

[BKH*11] Burch M., Konevtsova N., Heinrich J., Hoeferlin M., Weiskopf D.: Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011), 2440–2448. doi:10.1109/TVCG.2011.193. 4

[BOH11] Bostock M., Ogievetsky V., Heer J.: D^3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011), 2301–2309. doi:10.1109/TVCG.2011.185. 7

[BSN10] Battke F., Symons S., Nieselt K.: Mayday-integrative analytics for expression data. *BMC Bioinformatics 11*, 1 (2010), 121. doi:10.1186/1471-2105-11-121. 2

[CCKK12] Craig P., Cannon A., Kukla R., Kennedy J.: Matse: The microarray time-series explorer. In *Symposium on Biological Data Visualization (BioVis)* (2012), IEEE, pp. 41–48. doi:10.1109/biovis.2012.6378591. 2

[CMP09] Chen J., MacEachren A. M., Peuquet D. J.: Constructing overview+ detail dendrogram-matrix views. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 889–896. doi:10.1109/tvcg.2009.130. 2

[CT06] Chipman H., Tibshirani R.: Hybrid hierarchical clustering with applications to microarray data. *Biostatistics 7*, 2 (2006), 286–301. doi:10.1093/biostatistics/kxj007. 7

[DCMK07] Dotan-Cohen D., Melkman A. a., Kasif S.: Hierarchical tree snipping: Clustering guided by prior knowledge. *Bioinformatics 23*, 24 (2007), 3335–3342. doi:10.1093/bioinformatics/btm526. 3

[EBJ06] Ernst J., Bar-Joseph Z.: Stem: a tool for the analysis of short time series gene expression data. *BMC Bioinformatics 7*, 1 (2006), 1. doi:10.1186/1471-2105-7-191. 2

[EDL02] Edgar R., Domrachev M., Lash A. E.: Gene expression omnibus: Ncbi gene expression and hybridization array data repository. *Nucleic acids research 30*, 1 (2002), 207–210. doi:10.1093/nar/30.1.207. 6

[HB03] Harrower M., Brewer C. A.: Colorbrewer. org: an online tool for selecting colour schemes for maps. *The Cartographic Journal 40*, 1 (2003), 27–37. doi:10.1179/000870403235002042. 3

[HMB13] Hackett C. A., McLean K., Bryan G. J.: Linkage Analysis and QTL Mapping Using SNP Dosage Data in a Tetraploid Potato Mapping Population. *PLoS ONE 8*, 5 (2013), 1–21. doi:10.1371/journal.pone.0063939. 4

[HMBL07] Hackett C. A., Milne I., Bradshaw J. E., Luo Z.: Tetraploidmap for windows: Linkage map construction and qtl mapping in autotetraploid species. *Journal of Heredity 98*, 7 (2007), 727–729. doi:10.1093/jhered/esm086. 4

[ID91] Inselberg A., Dimsdale B.: Parallel coordinates. In *Human-Machine Interactive Systems*. Springer, 1991, pp. 199–233. doi:10.1007/978-1-4684-5883-1_9. 6

[KK99] Karypis G., Kumar V.: Chameleon: hierarchical clustering using dynamic modeling. *Computer 32*, 8 (1999), 68–75. doi:10.1109/2.781637. 2

[KLHS14] Koussounadis A., Langdon S., Harrison D., Smith V. A.: Chemotherapy-induced dynamic gene expression changes in vivo are prognostic in ovarian cancer. *British journal of cancer 110*, 12 (2014), 2975–2984. doi:10.1038/bjc.2014.258. 6, 7

[KTH*11] Kallio M. A., Tuimala J. T., Hupponen T., Klemelä P., Gentile M., Scheinin I., Koski M., Käki J., Korpelainen E. I.: Chipster: user-friendly analysis software for microarray and other high-throughput data. *BMC Genomics 12*, 1 (2011), 1–14. doi:10.1186/1471-2164-12-507. 2

[LHB*01] Luo Z. W., Hackett C. A., Bradshaw J. E., McNicol J. W., Milbourne D.: Construction of a genetic linkage map in tetraploid species using molecular markers. *Genetics 157*, 3 (2001), 1369–1385. URL: http://www.genetics.org/content/157/3/1369. 4

[LZH08] Langfelder P., Zhang B., Horvath S.: Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r. *Bioinformatics 24*, 5 (2008), 719–720. doi:10.1093/bioinformatics/btm563. 1, 3

[MABK11] Mahanta P., Ahmed H., Bhattacharyya D., Kalita J. K.: Triclustering in gene expression data analysis: a selected survey. In *2nd National Conference on Emerging Trends and Applications in Computer Science (NCETACS)* (2011), IEEE, pp. 1–6. doi:10.1109/ncetacs.2011.5751409. 2

[Mac86] Mackinlay J.: Automating the design of graphical presentations of relational information. *ACM Trans. Graph. 5*, 2 (1986), 110–141. doi:10.1145/22949.22950. 3

[MGT*03] Munzner T., Guimbretière F., Tasiran S., Zhang L., Zhou Y.: TreeJuxtaposer: Scalable Tree Comparison using Focus+Context with Guaranteed Visibility. *ACM SIGGRAPH 22* (2003), 453–462. doi:10.1145/1201775.882291. 2

[MMP09] Meyer M., Munzner T., Pfister H.: Mizbee: a multiscale synteny browser. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 897–904. doi:10.1109/TVCG.2009.167. 2

[MR10] McGuffin M. J., Robert J.-M.: Quantifying the space-efficiency of 2d graphical representations of trees. *Information Visualization 9*, 2 (2010), 115–140. doi:10.1057/ivs.2009.4. 4, 5

[MSF*09] Matthäus F., Smith V. A., Fogtman A., Sommer W. H., Leonardi-Essmann F., Lourdusamy A., Reimers M. A., Spanagel R., Gebicke-Haerter P.: Interactive molecular networks obtained by computer-aided conversion of microarray data from brains of alcohol-drinking rats. *Pharmacopsychiatry 42* (2009), S118–S128. 5

[Mun09] Munzner T.: A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 921–928. doi:10.1109/TVCG.2009.111. 3

[MV14] Montero P., Vilar J.: Tsclust: An r package for time series clustering. *Journal of Statistical Software 62*, 1 (2014), 1–43. doi:10.18637/jss.v062.i01. 6

[NWN*09] Navlakha S., White J., Nagarajan N., Pop M., Kingsford C.: *Finding Biologically Accurate Clusterings in Hierarchical Tree Decompositions Using the Variation of Information*. Springer Berlin Heidelberg, 2009, pp. 400–417. doi:10.1007/978-3-642-02008-7_29. 3

[NYO*12] Nielsen C. B., Younesy H., O'Geen H., Xu X., Jackson A. R., Milosavljevic A., Wang T., Costello J. F., Hirst M., Farnham P. J., et al.: Spark: a navigational paradigm for genomic data exploration. *Genome Research 22*, 11 (2012), 2262–2269. doi:10.1101/gr.140665.112. 2, 3

[OAMvdW15] Obulkasim A., A Meijer G., van de Wiel M.: Semi-supervised adaptive-height snipping of the hierarchical clustering tree. *BMC Bioinformatics* (2015), 1–11. doi:10.1186/s12859-014-0448-1. 3

[PMP*15] Pavlopoulos G. A., Malliarakis D., Papanikolaou N., Theodosiou T., Enright A. J., Iliopoulos I.: Visualizing genome and systems biology: technologies, tools, implementation techniques and trends, past, present and future. *GigaScience 4*, 1 (2015), 1–27. doi:10.1186/s13742-015-0077-2. 2

[SA15] Sakai R., Aerts J.: Card Sorting Techniques for Domain Characterization in Problem-driven Visualization Research. In *Eurographics Conference on Visualization (EuroVis) - Short Papers* (2015), Bertini E., Kennedy J., Puppo E., (Eds.), The Eurographics Association. doi:10.2312/eurovisshort.20151136. 4

[SM11] Sinha A., Markatou M.: A platform for processing expression of short time series (pests). *BMC Bioinformatics 12*, 1 (2011), 1. doi:10.1186/1471-2105-12-13. 2

[SND04] Saraiya P., North C., Duca K.: An evaluation of microarray visualization tools for biological insight. In *Symposium on Information Visualization* (2004), IEEE, pp. 1–8. doi:10.1109/INFVIS.2004.5. 2

[SS02] Seo J., Shneiderman B.: Interactively exploring hierarchical clustering results [gene identification]. *Computer 35*, 7 (2002), 80–86. doi:10.1109/mc.2002.1016905. 3

[VKA16] Vogogias A., Kennedy J., Archambault D.: Hierarchical Clustering with Multiple-Height Branch-Cut Applied to Short Time-Series Gene Expression Data. In *EuroVis 2016 - Posters* (2016), Isenberg T., Sadlo F., (Eds.), The Eurographics Association. doi:10.2312/eurp.20161127. 5

[WWLC08] Wang X., Wu M., Li Z., Chan C.: Short time-series microarray analysis: Methods and challenges. *BMC Systems Biology 2*, 1 (2008), 58. doi:10.1186/1752-0509-2-58. 5