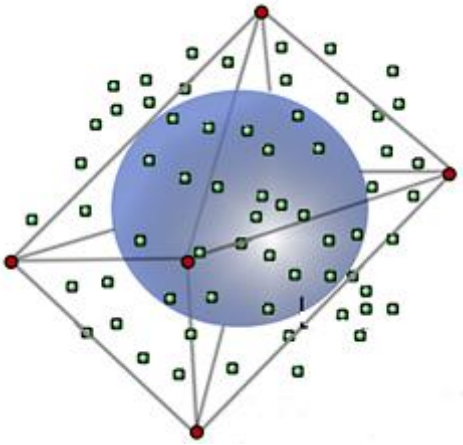
3D-VisualCluster Manual



Prototype tool and manual written by

José A. Castellanos Garzón[1] and Carlos Armando González[2]

[1]Department of Computer Science,

University of Valladolid

[2]Department of Computer Science and Automatics,

University of Salamanca

October, 2011

http://www.analiticavisual.com/jcastellanos/3DVisualCluster/3D-VisualCluster

CONTENTS

INTRODUCTION

3D-VisualCluster is a tool that combines existing visualizations with the novel ideas of our approach; it is addressed to capitalize on added value gained from the interaction between the approaches, and thus maximize the benefits to the user. As a first prototype of our approach a 3D-VisualCluster has been developed.

3D-VisualCluster is oriented to explore the quality of dendrograms, clusterings and clusters generated by the clustering methods (on the R language, http://www.r-project.org/) applied to DNA-microarray data. Therefore, this tool is based on principal component analysis (PCA) to reduce data dimensionality to a 3D space. So, a first approximation of the data distribution can be analyzed on a 3D scatter plot. Furthermore, visualization on parallel coordinates and views of DNA-microarray data are also presented in an interactive way. Note that connecting multiple visualizations through interactive linking and brushing provides more information than considering the component visualizations independently.

To conclude, this tool was implemented under research *"A Visual Analytics Framework for DNA Microarray Data Cluster Analysis"*; *José A. Castellanos Garzón, Carlos Armando García, Paulo Novais (Universidade do Minho) and Fernando Díaz (University of Valladolid)*. It has been submitted to journal "BMC Bioinformatics", http://www.biomedcentral.com/bmcbioinformatics/. This work has been partially funded by the Spanish Ministry of Science and Innovation, the Plan E from the Spanish Government, the European Union from the ERDF (TIN2009-14057-C03-02).

INSTALATION AND SETUP

Before starting to use the tool, Java should be installed, see installation tutorial "Java and Java3D Installation.pdf" on the web page of the tool for more details. After that, download and unzip tool file "3D-VisualCluster32bits.zip" for a 32-bits operating system (or "3D-VisualCluster64bits.zip" for a 64-bits operating system). This file contains "3D_VC_win.bat" for the windows system (and "3D_VC_linux.sh" for the linux system) that runs the tool, showing the following window:

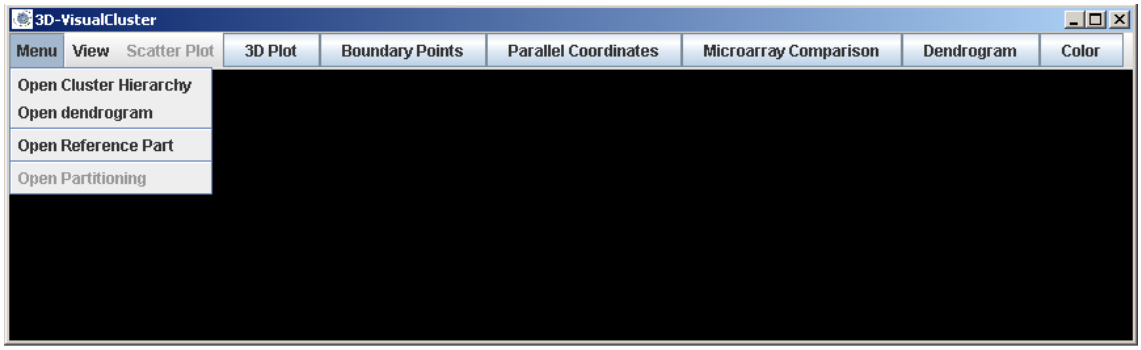


Fig. 1: Main window of the 3D-VisualCluster tool.

This tool can load four types of text files through the Menu option of Fig. 1. These files can be generated from R language, in fact; we provide a source file in R called "SaveToFiles.R" (available in the web site) with function *CreateFiles that* generates three types of files (dataset, cluster hierarchy and dendrogram graph) from the output of the hierarchical clustering methods in R (that is, from an object of classes twins, agnes, hclust, etc.). Function Create.*RefPartitionFile* creates the fourth file, which is a reference partition of the used data set. So, the first thing to do when we open the tool is to load the files to analyze, through the Menu option, that is:

a) Option "Open Cluster Hierarchy" loads two files. In this option the file with name ended in "_dendo.txt" is loaded. This is the file with the cluster hierarchy on a data set. Moreover, it also loads the analyzed data set with the name given in function *CreateFiles.* Fig. 2 shows a part of the internal structure of the file loaded by this option. Note that the first row of this file has the name of the used data set, whose structure is shown in Fig. 3. This data set has row and column header; moreover, each value in it is separated by a ";".

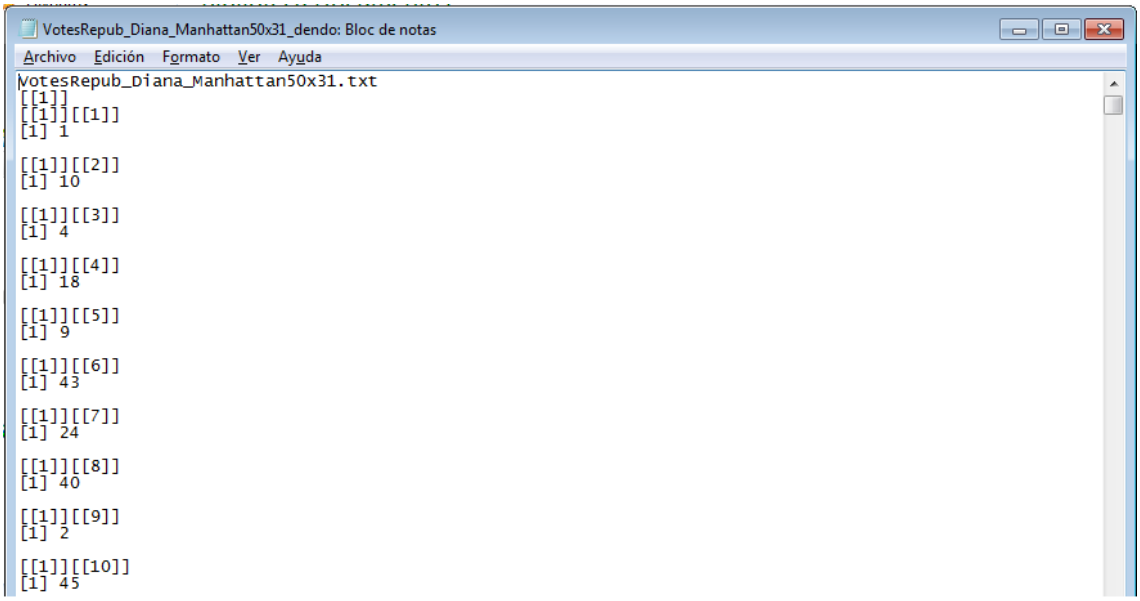


Fig. 2: Internal structure of the file loaded by option Menu of 3D-VisualCluster (cluster hierarchy file).

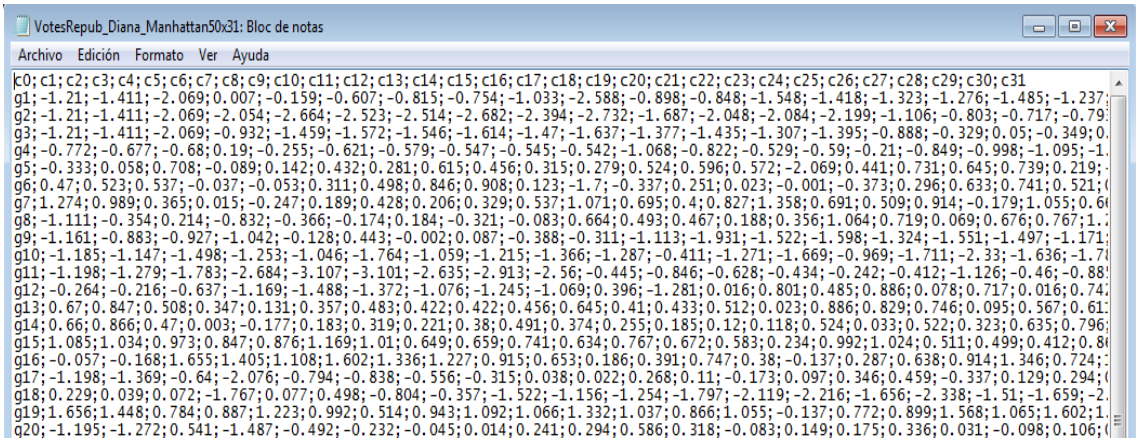


Fig. 3: Structure of a data set loaded by the tool (50 rows by 31 columns).

b) Once loaded the cluster hierarchy and the data set in step a), the graphic of the dendrogram should be loaded by the "Load Dendrogram" option. The name of the file to be loaded ends in "_dendo_gr.txt" and its structure is shown in Fig. 4. Note that this structure is the same as the internal one of a dendrogram in the R language.

c) If there is a file with a reference partition of the data set, then it is loaded by option "Open Reference Part" and its structure is in Fig. 5. This reference partition has five clusters as shown in the figure. The numbers in each cluster represent the objects to be clustered, which in this case are 384 objects.

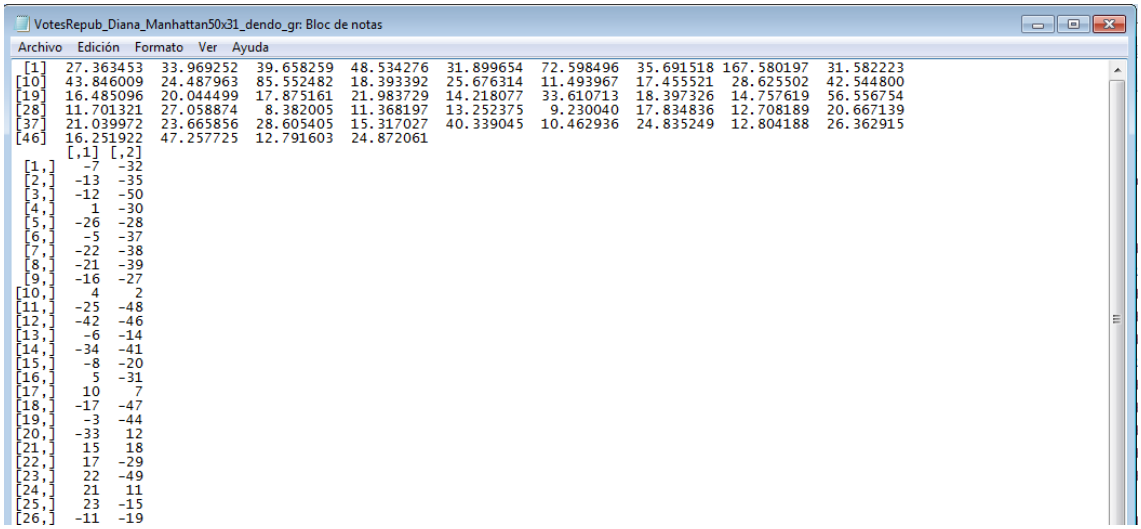


Fig. 4: Structure of the file that graphics the dendrogram in the tool.

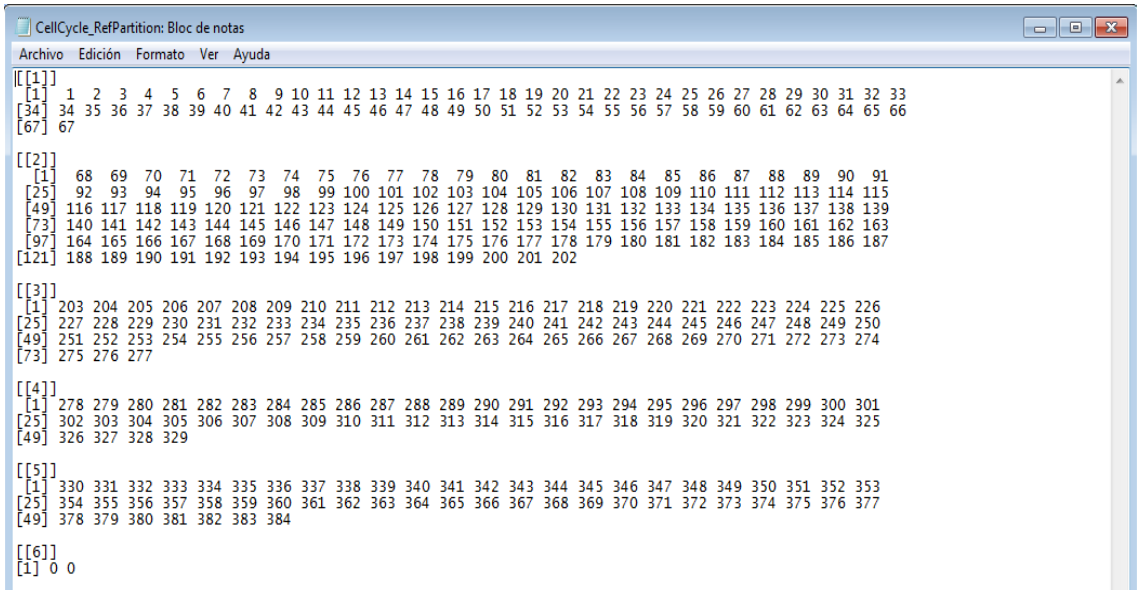


Fig. 5: Structure of a reference partition of a data set of 384 objects, which is clustered into 5 clusters.

Finally, we provide three examples on the web site of the tool, which can be downloaded and analyzed:

1. File “Test_Agnes.zip” (Example A) has a set of test data of a matrix of 18x4, where the Agnes method was applied.
2. File “VotesRepub_Diana.zip” (Example B) has a data set called *votes*.repub (50x31) given by R, which was modified and adapted as test data for the tool. The Diana method was used.
3. File “Cellcycle_Diana.zip” (Example C) has a public data set called *cellcycle* (384 genes x 17 samples) at http://faculty.washington.edu/kayee/cluster. This file also provides a reference partition of *cellcycle.* The Diana method was applied.

USING DIFFERENT COLORS IN THE MICROARRAY VIEWS

Before starting to use the microarray, dendrogram and parallel coordinates views, it is necessary to choose the color scale that best fits the used data set. To show that, we have used Example B, for which the view of Fig. 6 is generated.

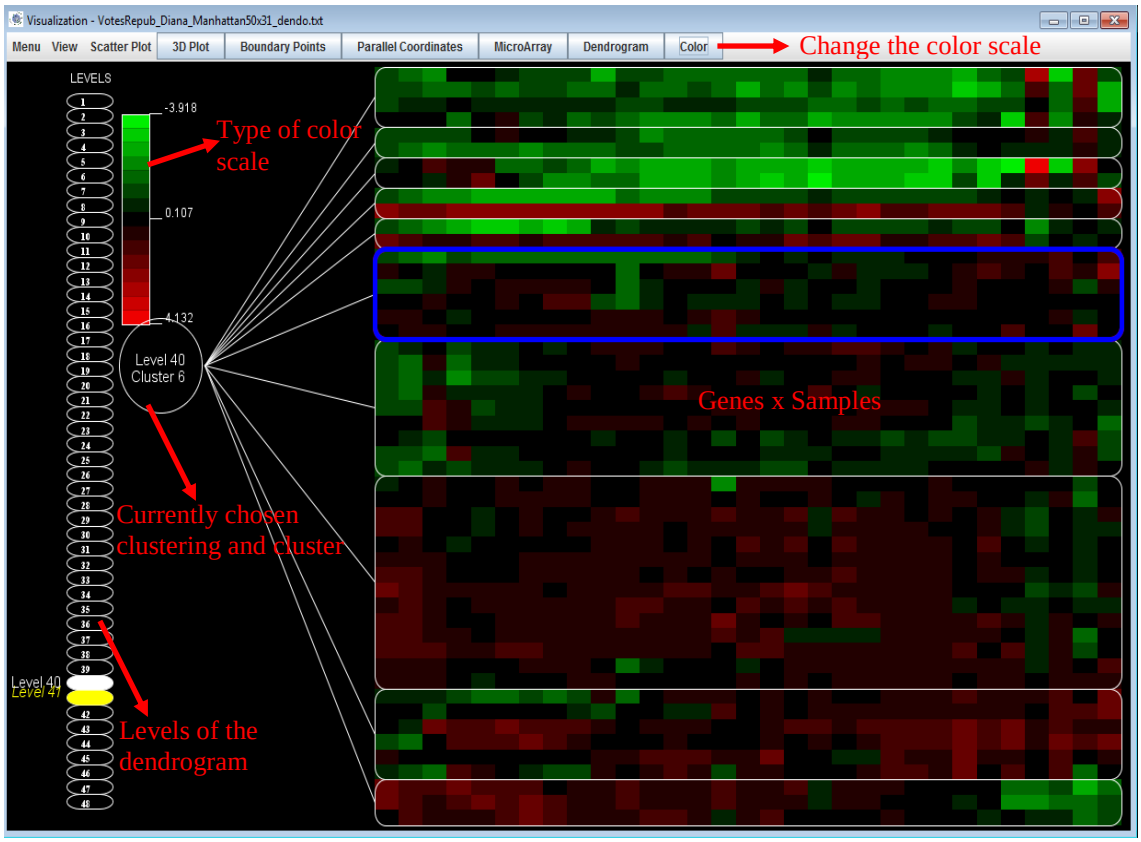


Fig. 6: Main window of the tool with Example B. This view performs as an explorer of clustering.

On the left side this figure shows, the levels of the dendrogram, color scale and the currently chosen level (clustering) with the current cluster. On the right side, we show the microarray where rows mean genes and columns mean samples. All clusters of current clustering are delimited by white rectangle, and the selected cluster is represented by a blue rectangle. Note that this view allows exploring each level and each cluster of the dendrogram. There are three types of color scale (combining colors green and red) and each scale can select the degree of color degradation, by default it is 15. This can be seen pressing button "Color" of the task bar in Fig. 6, which yields the view of Fig. 7.

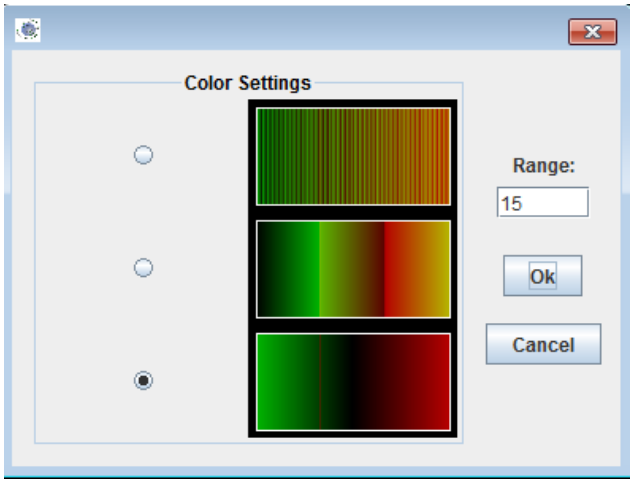


Fig. 7: Different color scales for the microarray.

As shown in this figure, the active color scale is the third (as also shown in Fig. 6) with 15 colors (Range option). The first and second scales yield the following color combinations shown in Fig. 8 and Fig. 9 respectively.

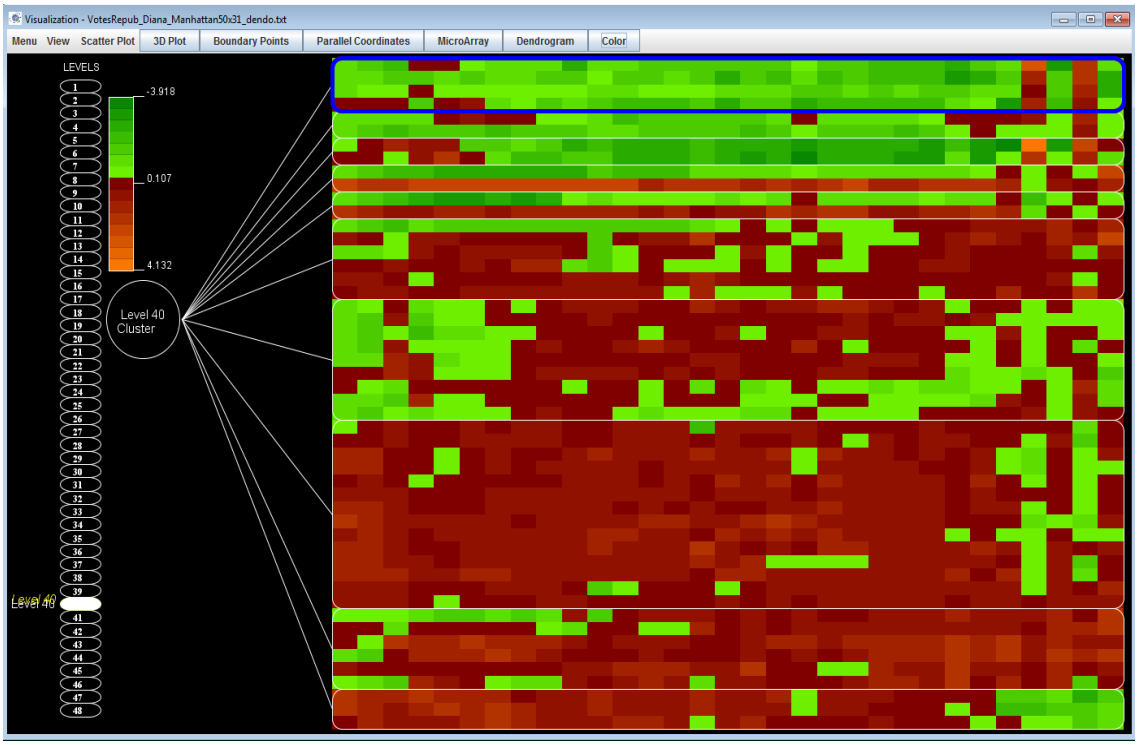


Fig. 8: Microarray selecting the first color scale of Fig. 7

Fig. 10 shows the microarray selecting the third color scale of Fig. 7 but in this case, decreasing the number of colors from 15 to 5 (Range = 5).

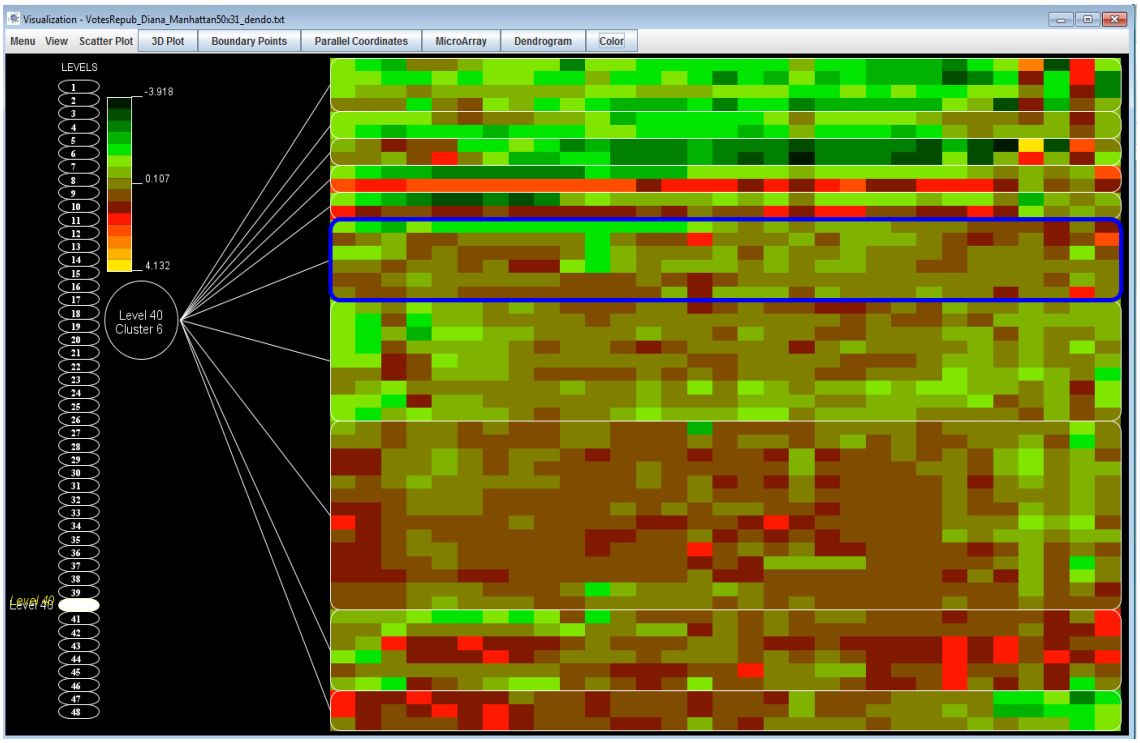


Fig. 9: Microarray selecting the second color scale of Fig. 7

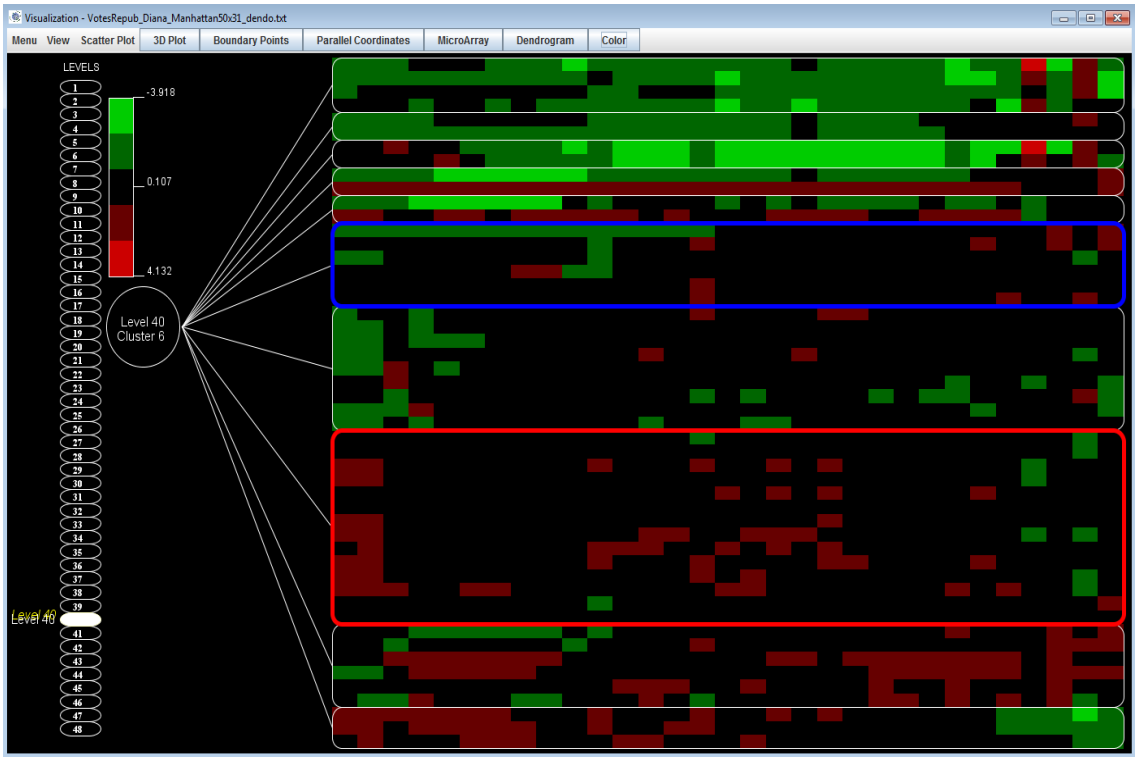


Fig. 10: Microarray selecting the third color scale of Fig. 7 but Range = 5 (only five colors).

COMBINING MICROARRAY VIEWS WITH OTHER VIEWS

We can now combine and validate microarray views with other views that use other techniques of visualization. First, the main window (Fig. 6) is linked with the other views (interactivity) and any modification in the selection of the level or cluster within level is updated on the remaining views. Continuing with Example B of Fig. 6, we can open the view that combines dendrogram and microarray from button "Dendrogram" of the task bar of this figure. The dendrogram is shown in Fig. 11. This figure provides an overall view of the clusters formed in the process of grouping. Note that the current cluster is underlined. Moreover, the value in the intersection of each row and column is shown when the mouse pointer is over it. In general, these values are shown for all views with microarray. Additionally, in option "Data Table" of menu "View" in the task bar of the tool main window we can display the data set values in form of matrix.

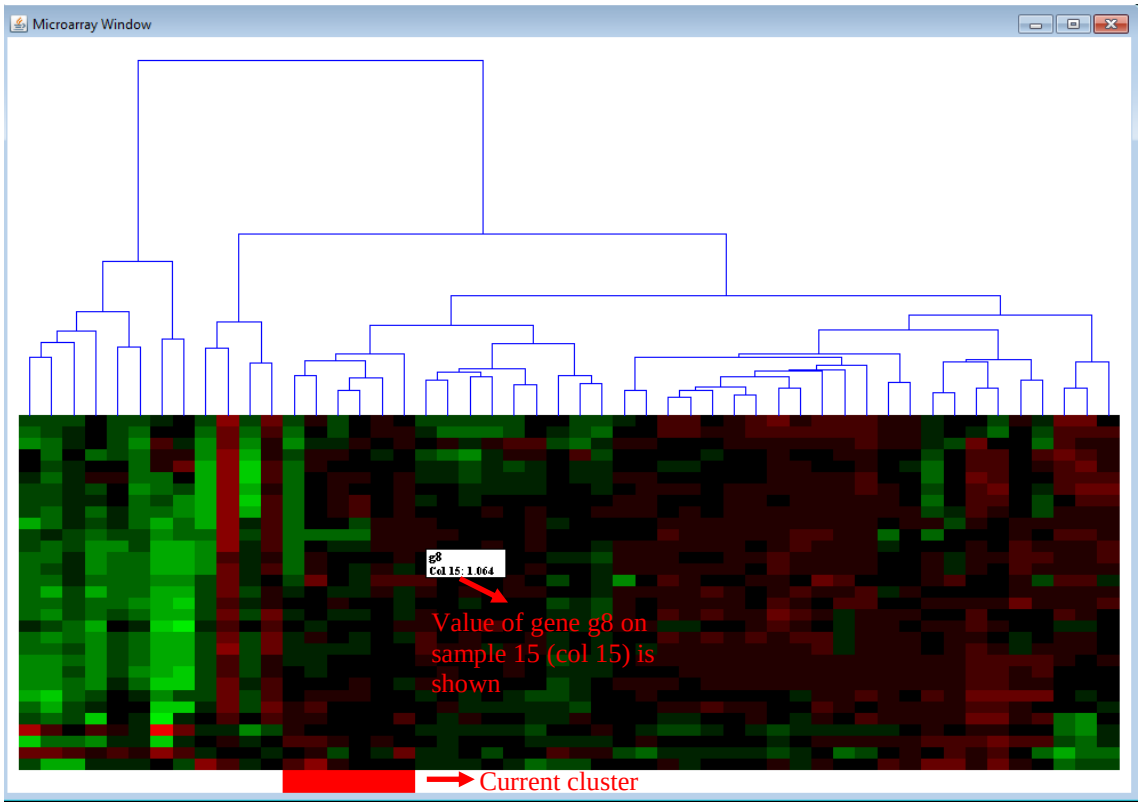


Fig. 11: Combination of dendrogram and microarray. Current cluster of Fig. 6 is shown.

The following view shown in Fig. 12 is generated by pressing button "Microarray Comparison" of the task bar in Fig. 6. This figure shows a comparison of the original data set with regard to the ordered one by the applied clustering method. Note that the red lines between both microarrays relate the position of each gene of the current cluster of the ordered microarray with the position of the same genes but in the original microarray. All the above allow seeing how genes in the original microarray were organized to create grouping structures.

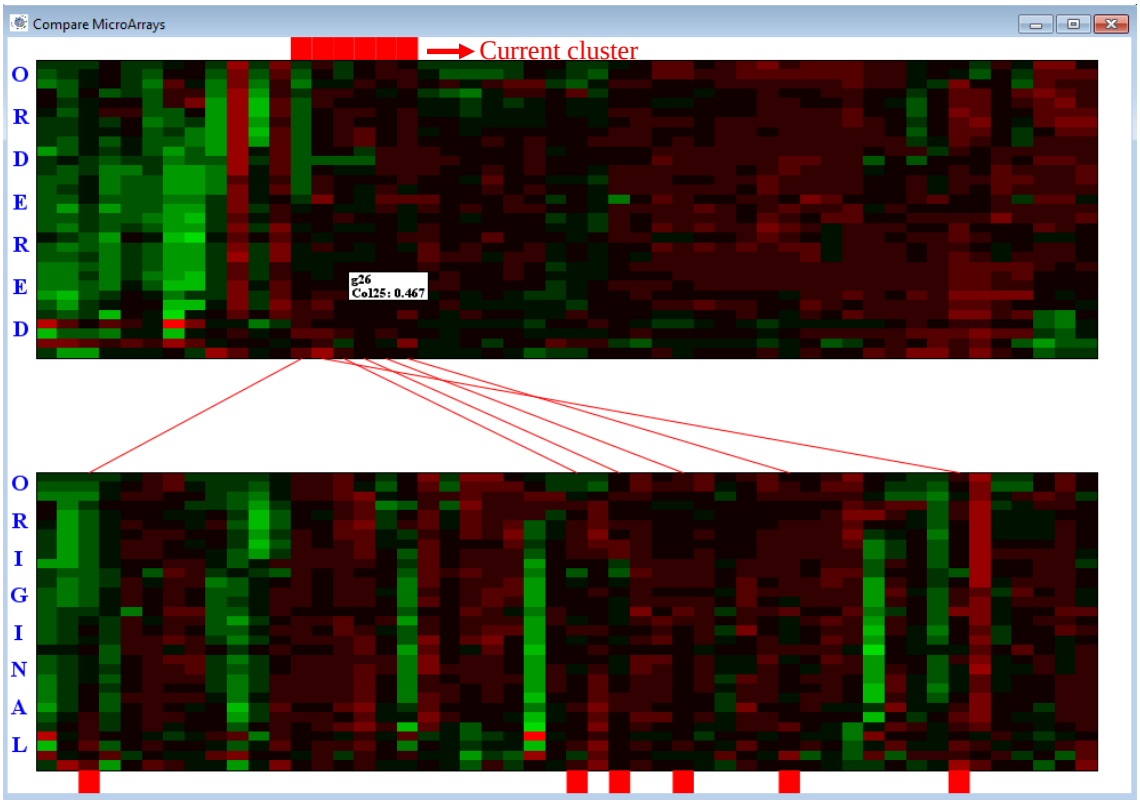


Fig. 12: Comparison of the original microarray with the one ordered by the Diana method.

To conclude on different microarray views, Fig. 13 shows a combination of parallel coordinates on samples of the genes of the active cluster in Fig. 6. This figure represents a zooming of the current microarray and makes a validation of cluster quality with regard to parallel coordinates on the samples. Note that each of the four views means a different way of visual cluster analysis.

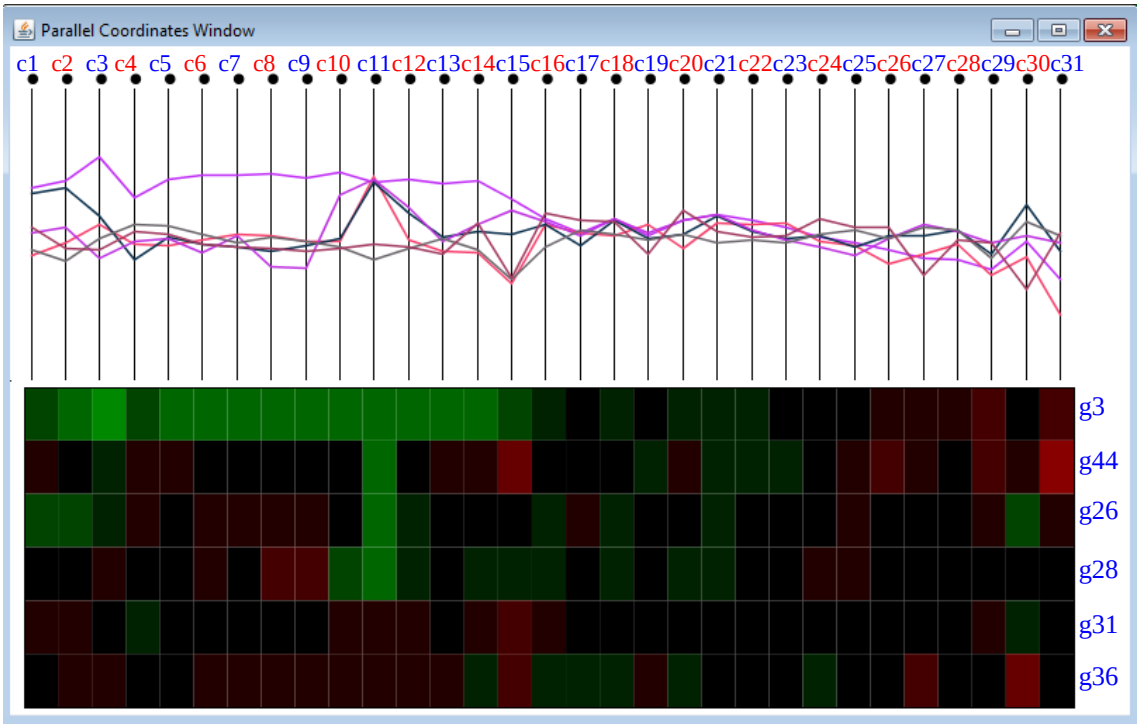


Fig. 13: Comparison of a cluster in form of microarray (genes x samples) with regard to the sample parallel coordinates.

3D SCATTER PLOT VIEWS

The 3D scatter plot view is also linked to the previously analyzed views, and provides a set of functionality such as: different representations (in form of points) of the whole data set according to the level chosen in the main window, comparison of a reference partition (as 3D surfaces) of the data set with the current clustering (as 3D points), computation of boundary gene-points of the current cluster, different 3D surface reconstructions of the current cluster among others.

In order to show the reliability of the scatter plot view, we have chosen Example C, which has a reference partition of the data set. This example yields the following microarray view of Fig. 14. The dendrogram of this example is shown in Fig. 15.

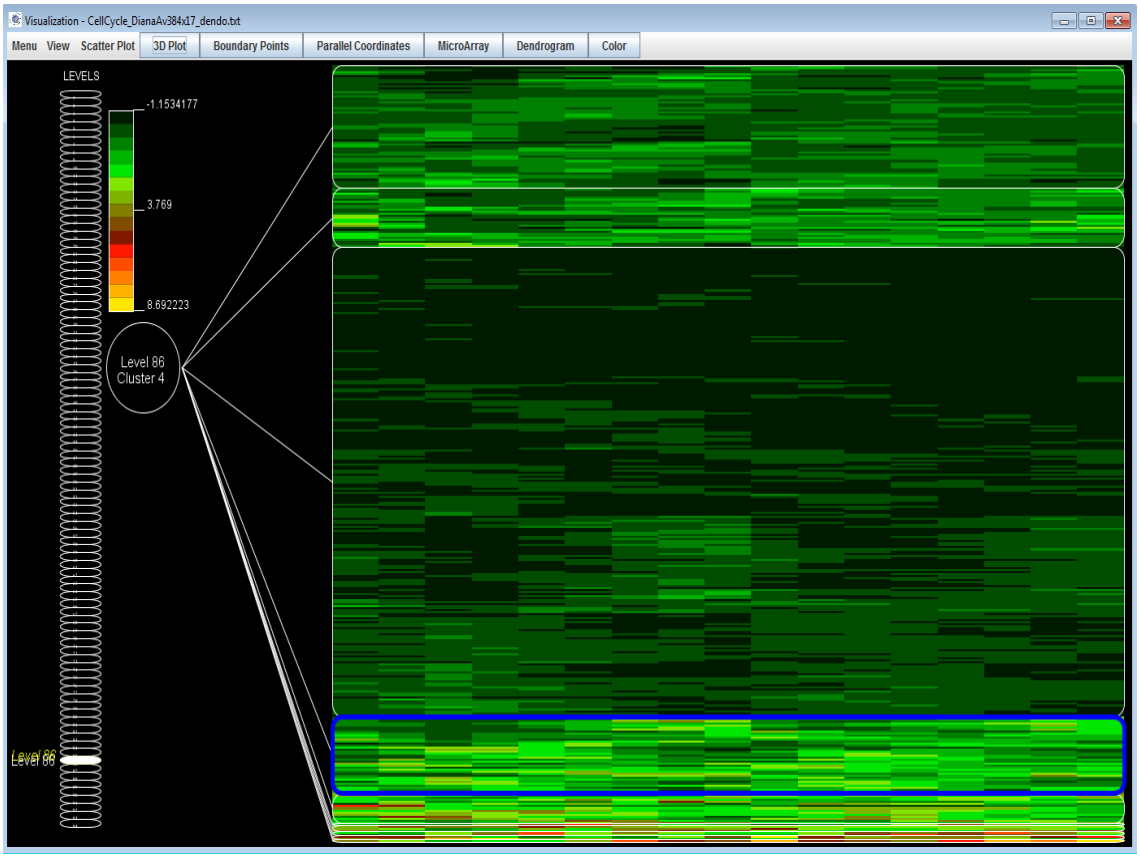


Fig. 14: Main window microarray view of Example C.

Fig. 16 shows a representation in form of points of the used data set in a 3D scatter plot view (applying Principal Component analysis). Points in the same cluster are painted in the same color, whereas points in different clusters are painted in different colors. Each point represents a gene of the data set. The gene name associated to a point can be seen putting the mouse pointer over that point in the scatter plot.

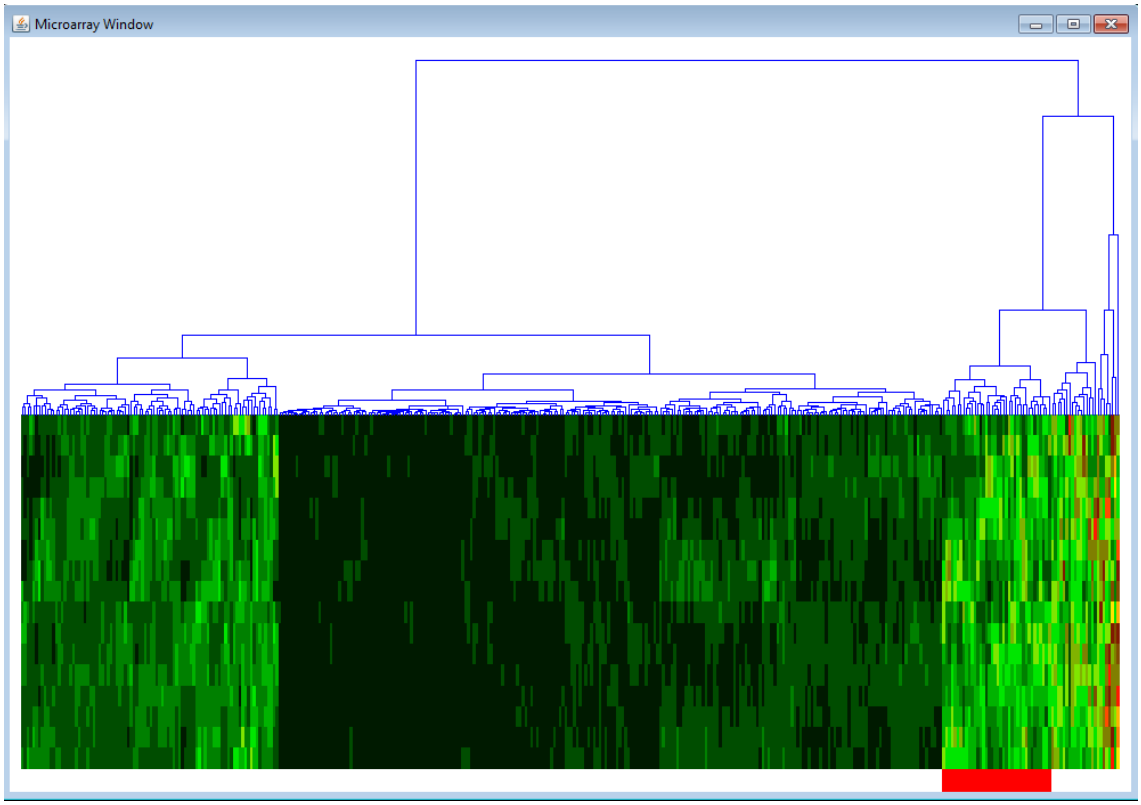


Fig. 15: Dendrogram and microarray view of Example C.

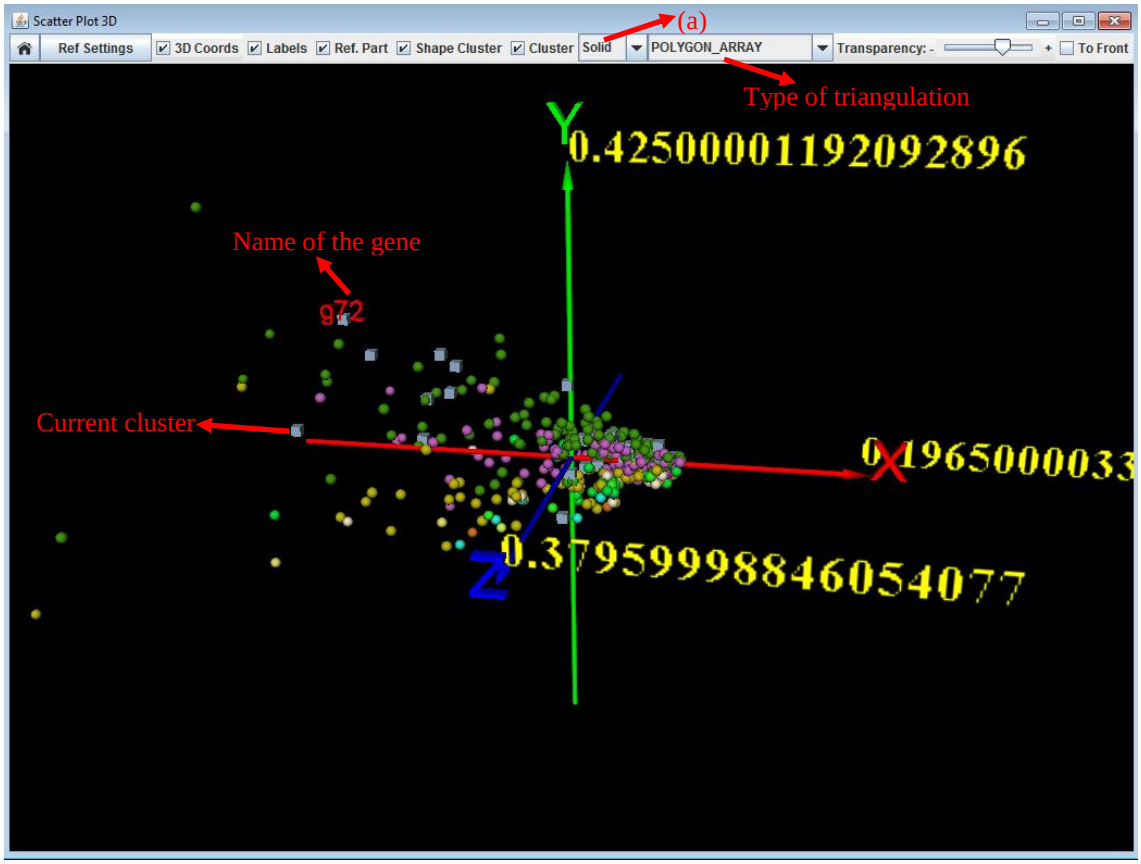


Fig. 16: 3D scatter plot of the data set of Example C. Each cluster of the current clustering of Fig. 14 is shown in a different color.

Additionally, menu “View” of the task bar of the main window provides three forms of showing the current cluster:

1. option “Change Shape Cluster”: points of the current cluster shown in form of cubes as in Fig. 16;
2. option “Compare Clusters”: points of the current cluster shown in a color and the remaining points shown in another color (Shown in Fig. 17);
3. and option “Show Cluster” shows only the points of the current cluster (insulating the current cluster) as in Fig. 18.

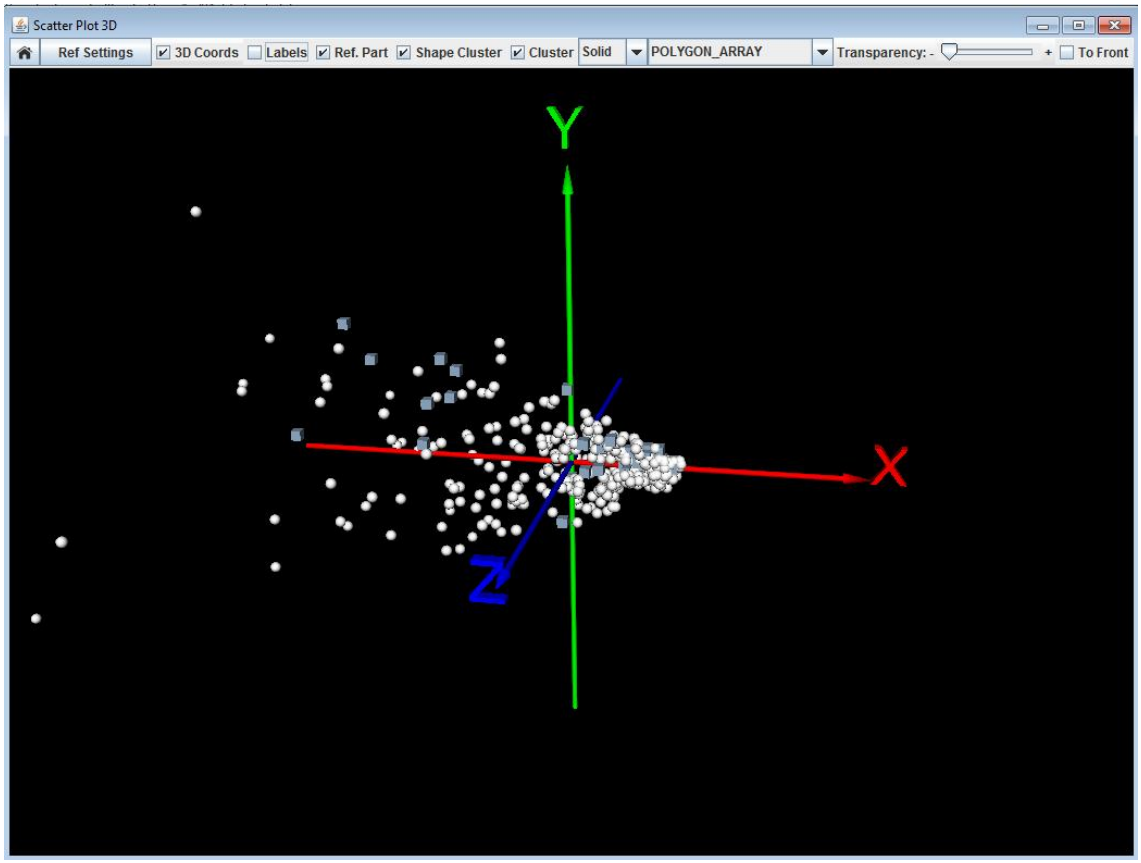


Fig. 17: Points in the current cluster are shown of a different color and shape of the remaining points.

Boundary Points and Surface Reconstruction of a Cluster:

Boundary points (boundary genes) of a cluster can be computed in three ways, which provide three different cluster boundaries according to the selected radio as shown in Fig. 19. This window is shown from option “Boundary Points” of the task bar in the main window. The result of applying the algorithm of computing boundary points of a cluster according to selected radio is shown in Fig. 20. Note that passing from maximum radius to minimum radius, the number of boundary points increases; in fact that increases the accuracy of the boundary of a cluster.

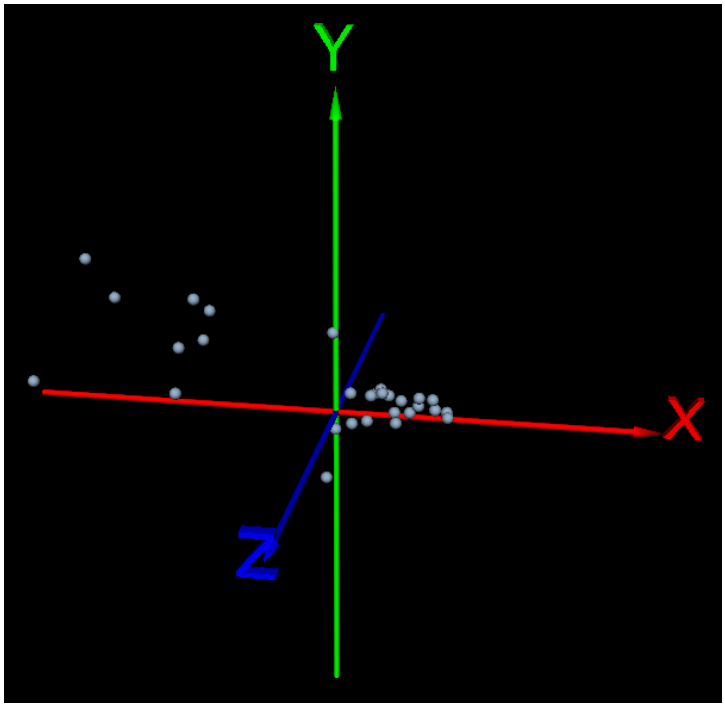


Fig. 18: Only points of the current cluster are displayed.

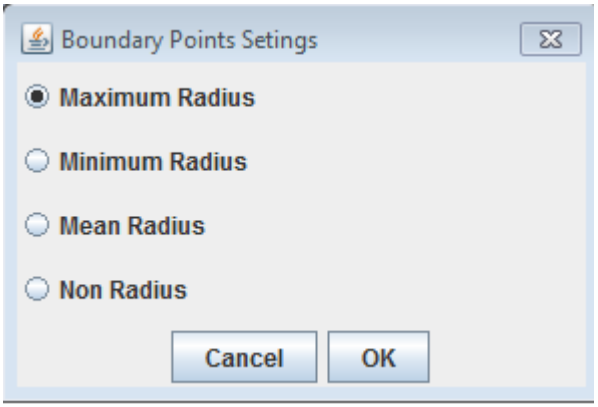


Fig. 19: Different radios to reconstruct the cluster boundary.

The boundaries of the cluster of Fig. 20 generate the 3D surfaces of Fig. 21, which display a cluster in form of surface. Additionally, we can choose three types of triangulation (for each radio) to reconstruct the surface of a cluster as shown in the task bar of the Fig. 16. The surfaces of Fig. 21 were reconstructed using triangulation "POLYGON_ARRAY". The remaining triangulations "TRIANGLE_FAN_ARRAY" and "TRIANGLE_STRIP_ARRAY" are shown in Fig. 22 by selecting option "Mean Radius" of Fig 19. The three types of triangulations are also shown in Fig. 23 but in form of lines on the boundary points of the cluster.

Options "Point" (Fig. 20), "Solid" (Fig. 21) and "Lines" (Fig. 23) can be selected in option a) of the task bar of Fig. 16, after computing the boundary of the cluster. Note that through the scatter plot we offer three additional ways of displaying a cluster with respect to previous visualizations: a cluster displayed trough its points (Fig. 18), boundary points (Fig. 20) or shape (Fig. 21).

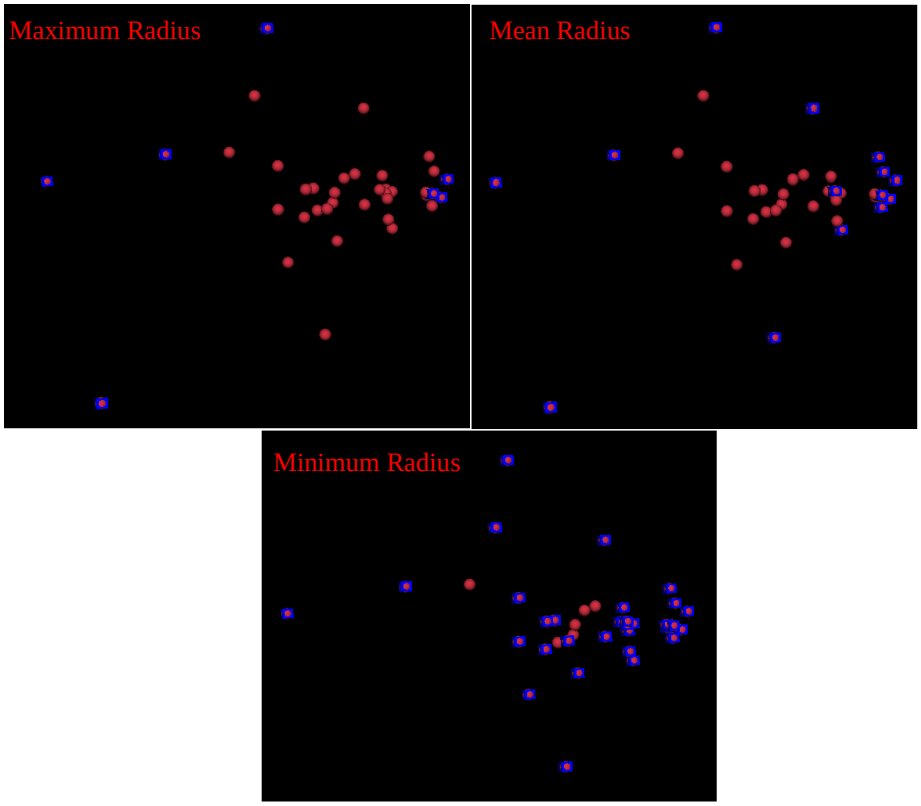


Fig. 20: Three types of boundaries for a gene cluster, boundary genes marked in blue color.

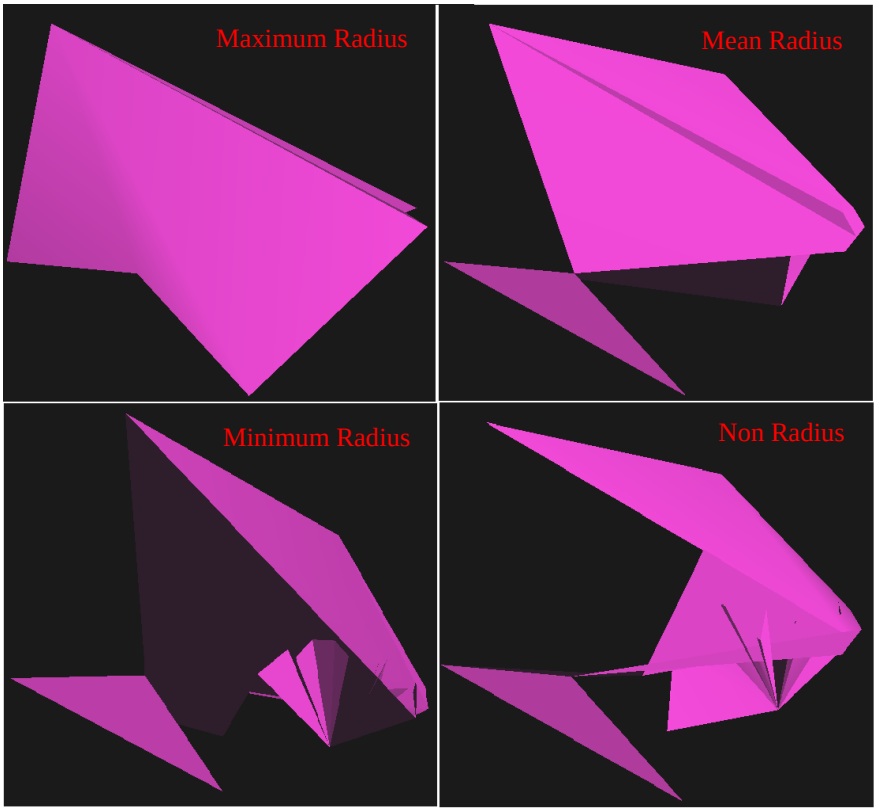


Fig. 21: 3D surfaces (POLYGON_ARRAY triangulation) representing the same cluster of Fig. 20 according to the selected radio.

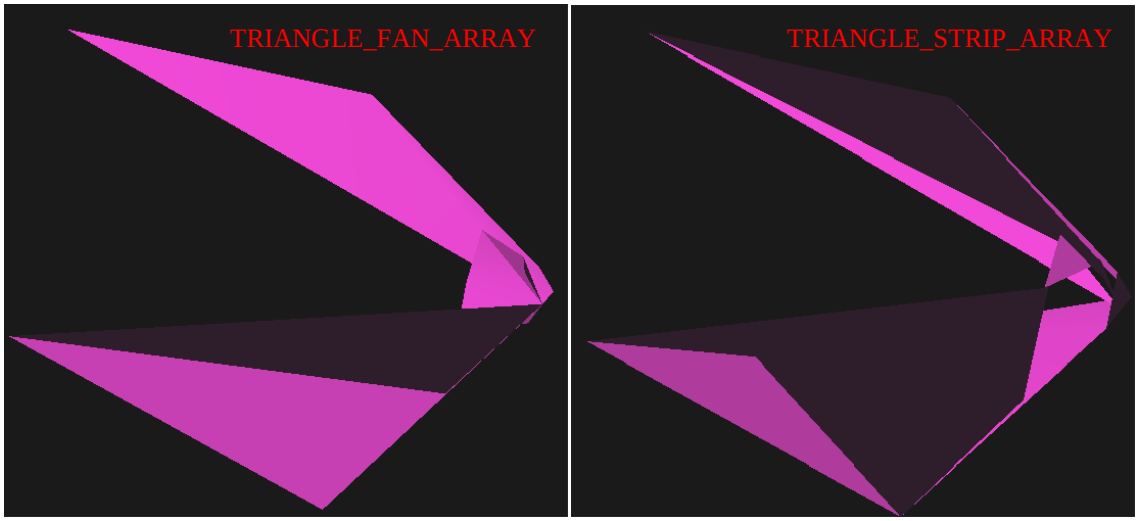


Fig. 22: 3D surfaces of the cluster of Fig. 19 with a different triangulation using option “Mean radius”.

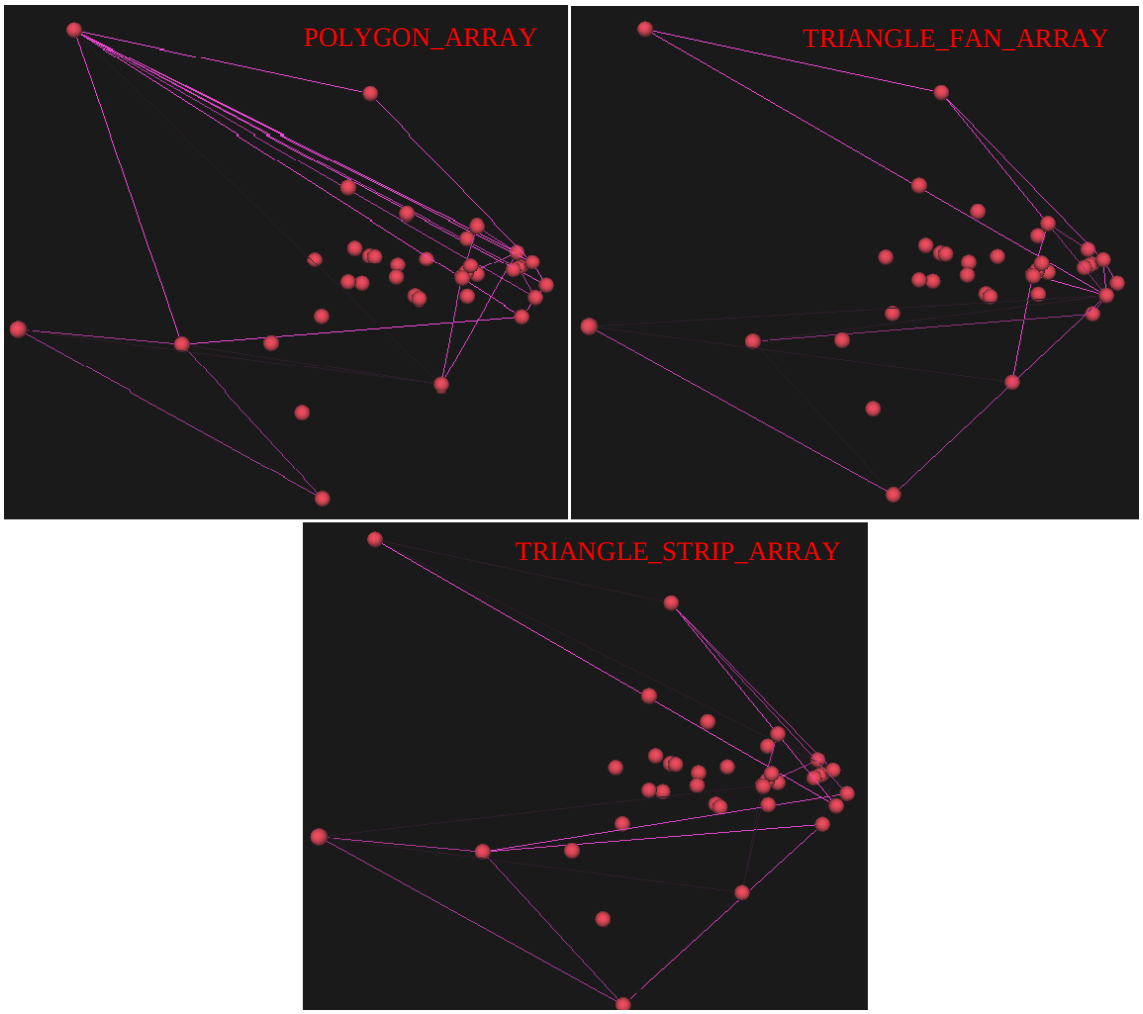


Fig. 23: Representation of the shape of a cluster through the lines connecting boundary points.

Reference Partition and Clusters:

One of the main applications of the cluster surface reconstruction based on boundary points in this paper is representing reference partitions. There are several statistical indicators to compare a clustering with a reference partition; however, there is no visual approach to make this comparison. Each cluster of the reference partition of Example C is represented through a 3D surface (shown in Fig. 24). Clusters of a reference partition are called partitions.

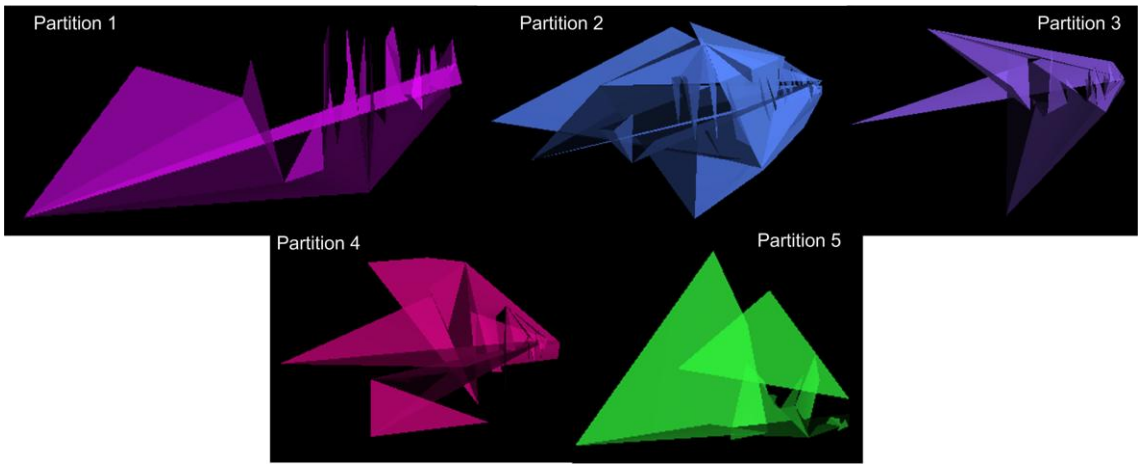


Fig. 24: Clusters (partitions) of the reference partition of Example C in form of translucent 3D surfaces.

Note that comparing a reference partition with a clustering we can verify the degree of agreement between both or simply, verify the results of statistic measures. This way, we can visually choose the level of a dendrogram which better approximates the reference partition. Each surface-partition can be selected (or unselected) from option "Ref Settings" of the task bar of Fig. 16, which shows the window of Fig. 25.

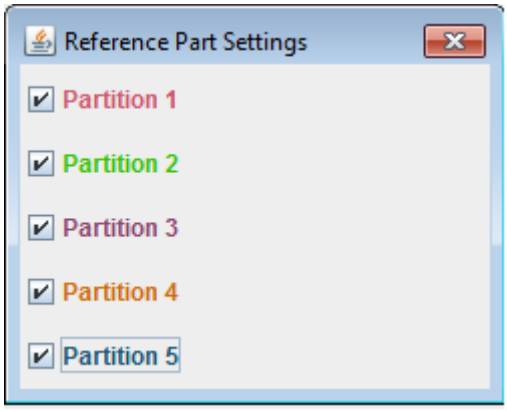


Fig. 25: Displays each surface-partition of the reference partition on the scatter plot.

Fig. 26-a) displays a cluster that will be compared with the partitions of reference partition to choose the one which is the most similar to the cluster.

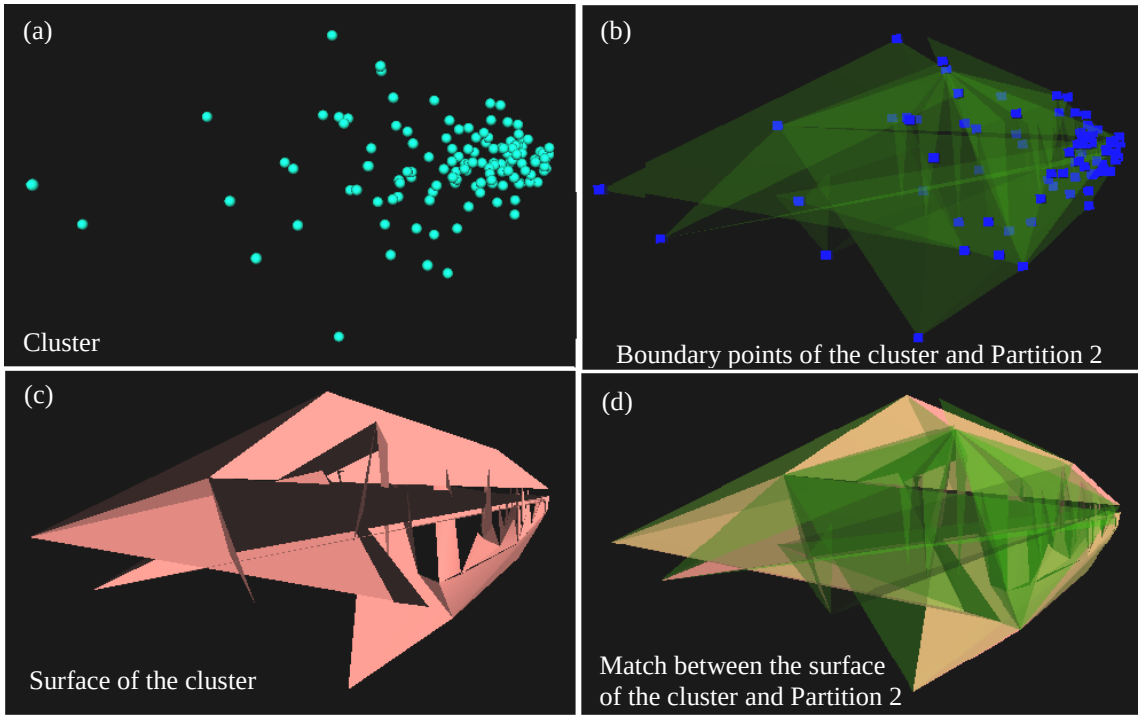


Fig. 26: Displays the similarity between a gene cluster and a partition of the reference partition.

As shown in Fig. 26-b), Partition 2 shows a high degree of similarity with respect to the boundary points of the selected cluster; thus, the remaining points of the cluster will be within of the surface of Partition 2. Fig. 26-c) shows the surface of the cluster which is very similar to surface-partition 2 and finally, Fig. 26-d) shows an almost perfect matching between Partition 2 and the surface of the selected cluster. This shows the selected cluster is a good one according to reference partition. Note that the above analysis can be done for any other cluster of the clustering selected on the dendrogram of the tool main window.

To summarize, we can say that the most important contributions of our tool focus on the scatter plot, where the knowledge discovery process of the remaining views can be improved by using it.