Output File Formats for SC^{2}ATmd v3

Last Updated on 6/13/2012 by Amy Olex

Index

This section describes the format and content of all the output files for each analysis function. Output files come in 3 types: text files, image files, and Matlab .fig files.

Important Notes:

- ***All output will be saved in the same location as the input file unless the user specifies another destination.***

Figure of Merit Analysis

top

By default the figure of merit analysis produces a text file (.txt) and an image file (.fig). The image may be saved in additional image file formats according to user preference. A description of each follows.

Text file: *myfom.txt*

A summary of the FOM analysis is output in a text file that is similar to Figure 1.

```
Figure of Merit analysis using the original Euclidean-biased FOM.
Cluster list: 2   6  10  14  18  22  26  30  34
Optimal Cluster Algorithm is K-means

Hierarchical
Clusters:     2       6    10    14    18    22    26    30    34
FOMscores:  9.02    5.61  5.31  5.14  4.81  4.59  4.49  4.42  4.30

K-means
Clusters:     2       6    10    14    18    22    26    30    34
FOMscores:  7.12    5.44  4.95  4.68  4.57  4.47  4.36  4.30  4.24

Random
Clusters:     2       6    10    14    18    22    26    30    34
FOMscores: 10.65   10.64 10.58 10.69 10.67 10.57 10.68 10.56 10.66

The Optimal Cluster Range is [6  10]
```

Figure 1: FOM text output file.

The first line identifies the figure of merit version that was used. If Euclidean distance was chosen as the similarity measure, then the original Euclidean-biased version of the FOM will be used, else if Pearson's correlation coefficient is used then the correlation-biased FOM will be used. The second line reiterates the list of cluster numbers the user entered. The third line indicates the clustering algorithm that performed the best on this data; this is the recommended clustering algorithm. The next three sections list the sequence of FOM scores for each clustering algorithm chosen. This is followed by the identification of the optimal number of clusters to use with the input data set.

Image files: *myfom.fig, .jpg, .tiff, .bmp, .eps, .ai and .pdf*

For each iteration of the FOM algorithm the scores are plotted on a graph that is automatically saved as a .fig file; the other image file formats may also be generated if the user chooses. In Figure 2 the x-axis lists the number of clusters used in each iteration of the FOM algorithm, and the y-axis is the FOM score. Each line on the plot indicates the series of FOM scores calculated for a clustering algorithm.
The .fig file can be opened and manipulated in Matlab; otherwise the other image files can just be imported into documents as-is.

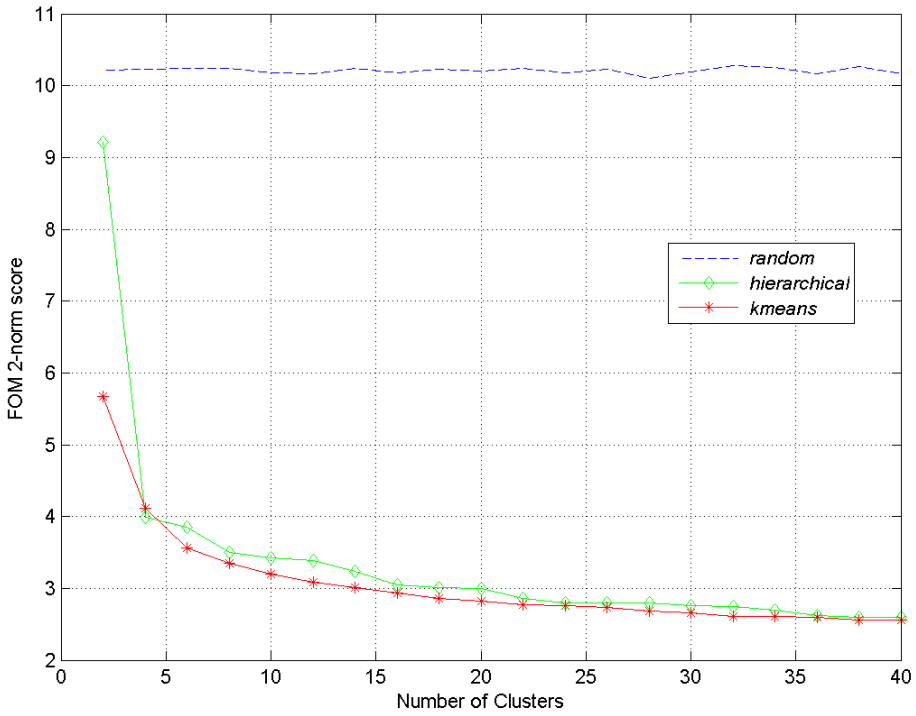

Figure 2: FOM graph output.

What is a FOM analysis?
The FOM analysis is a quantitative analysis that compares the performance of different clustering algorithms on a set of data. The clustering method that gets the lowest FOM score creates the most homogeneous clusters, and is therefore the best method. The FOM analysis reveals which clustering algorithm is best suited for each data set, and it gives a range for how many clusters are optimal. For more details on what the FOM is see (Yeung, Haynor et al. 2001).

Determining the Optimal Clustering Method
The lower the FOM score, the better; therefore the line closest to the bottom of the graph is the optimal clustering method. This program calculates the average FOM score for each method, and the method with the lower average is chosen as optimal; this information can be found in the text output file discussed previously.

Choosing the Optimal Number of Clusters
The number of clusters is chosen based on where the 'elbow' in the graph is. This 'elbow' indicates that increasing the number of clusters is not improving the overall cluster homogeneity, so the FOM score is not improving much and is flattening out. The optimal number of clusters is indicated in the text summary file; any number of clusters that fall in the enclosed range are acceptable.

Customizable Graph Features (Matlab users)
Matlab's .fig file gives the user the capability to customize the look of the FOM plot. Matlab tutorials can be found on the web; below is a short, non-comprehensive list of customizable graph features.

- Font and font size
- Grid lines on/off
- Axis labels
- Title and Legend
- Axis markers
- Line color, shape and size
- Data marker color, shape and size
- Background color
- Graph dimensions

Standard Clustering Analysis top

This section describes the output generated by the standard clustering analysis. One text output file is generated automatically, and the user has the option of also creating a text file containing the cluster statistics, and image files representing each cluster as a heatmap.

The standard clustering algorithm performs the selected clustering method on the loaded data set once, and generates a text file with all the clustering results. If the 'Generate Heatmap' option is chosen, one heatmap with dendrogram will be generated for each cluster and saved as a Matlab .fig file; other image file formats may also be selected. If hierarchical clustering is selected a global dendrogram is output in addition to the heatmaps as a .fig and .jpeg file; this dendrogram relates each cluster to the others so that the entire hierarchical tree can be reconstructed if desired.

Text file: *myclusters.txt*

Figure 3 is an example of the clustering output file if heatmaps are also generated. If heatmaps are not generated, then the second column, 'clusterOrder' will not be included.

cluster#	clusterOrder	AffyID		1hr	3hr	6hr	12hr	24hr
1	1	1455581_x_at	0.8	2.3	3.1	3.2	2.4	
1	2	1436172_at	0.9	2.4	3.1	3.2	2.5	
1	3	1446090_at	0.9	2.3	3.4	3.9	2.6	
1	4	1448436_a_at	1.2	3.2	3.1	3.2	2.3	
1	5	1432548_at	1.3	3.1	3.7	3.4	2.6	
1	6	1446457_at	1.1	3	3.7	3.3	2.4	
1	7	1450446_a_at	0.7	3	3.4	3.3	1.9	
1	8	1458512_at	1.4	2.8	3	4.2	1.9	
2	1	1459659_at	3.1	1.8	0.3	0.4	-0.3	
2	2	1460312_at	3.1	1.5	0.5	0.4	-0.1	
2	3	1443392_at	2.3	2	0.1	1	-0.2	
2	4	1447301_at	2.1	3.1	0.8	0.2	-0.9	
2	5	1458202_at	2.2	2.4	1	0.1	-0.1	
2	6	1459147_at	2.1	3.6	0	0.2	0	
2	7	1457235_at	2.2	2.2	-1.6	0.6	0.8	
2	8	1445471_at	3.5	0.4	0.8	0.9	0.7	
2	9	1456720_at	4.4	1	1.2	0.6	-0.1	
2	10	1423175_s_at	3.7	3.8	0.3	1.9	2.9	
2	11	1432808_at	3	3.7	1.3	1	1.9	
2	12	1432904_at	3.4	2.9	0.3	0.6	1.5	
2	13	1443789_x_at	2.3	2.5	0.7	0.6	2.2	
2	14	1459044_at	2.2	2.8	0.6	0.5	1.8	
2	15	1450823_at	2.4	2.6	0.4	1.4	1.7	
3	1	1437754_at	2	1.5	2.7	1.8	2.2	
3	2	1458737_at	1.6	2	2.9	1.7	2.5	
3	3	1443694_at	0.8	1.1	2.6	1	2.7	
3	4	1446990_at	0.8	0.5	3	1.2	3.2	

Figure 3: Clustering text output file.

Text file description:

- Column 1: This column contains the cluster assignments for each gene in the file.
- Column 2: This column contains the order of genes in each cluster on the heatmaps.
- Column 3: The unique gene labels provided by the user.
- Columns 4-n: The imported data for each gene that was used to generate the clusters.

Heatmap Image files: *myclusters-clusterX.fig*
The clustering algorithms provide the option to generate one hierarchically clustered heatmap for each cluster. The image files may be used as-is, or the Matlab .fig file may be customized by advanced Matlab users. An example heatmap and dendrogram are shown in Figure 4.

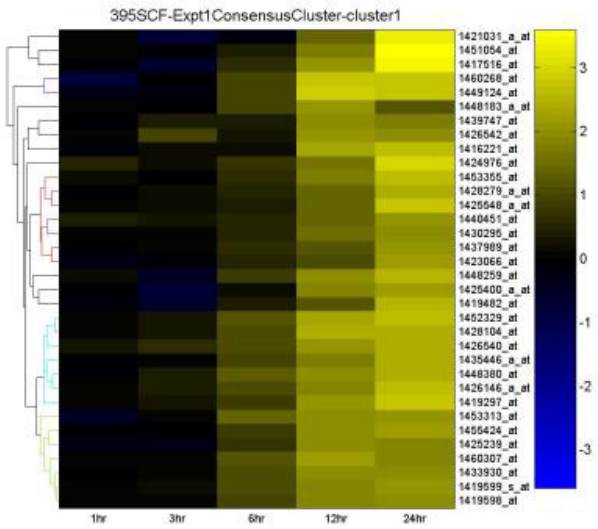

Figure 4: Clustering heatmap output.

Each column of the heatmap represents an experimental condition, and each row is one gene. Column labels are located at the bottom, row labels on the right, and the file name is at the top. The dendrogram to the left was generated by re-clustering each cluster using HAC and Euclidean distance in order to organize each het map. This way, each gene can be related to the others that are within the same cluster where the height of each branch indicates how similar two genes or groups of genes are (shorter branches indicate a stronger similarity). The color index is located on the right; either a red-green, yellow-blue or red-white-blue color map can be selected where red/yellow indicate an increase from the control, green/blue represents a decrease from the control, and black/white is no change.

Global Dendrogram files: *myclusters-gden.fig and myclusters-gden.jpg*

This unique implementation of the traditional hierarchical clustering algorithm results in the complete hierarchical tree being divided into the pre-selected number of clusters. To relate the individual clusters, the 'top' of the complete dendrogram is saved as a .fig and jpeg file, so that the user is able to relate each reported cluster and reconstruct the entire tree. An example dendrogram is shown in Figure 5 where each numbered leaf represents a cluster of genes for which a heatmap was created as above.

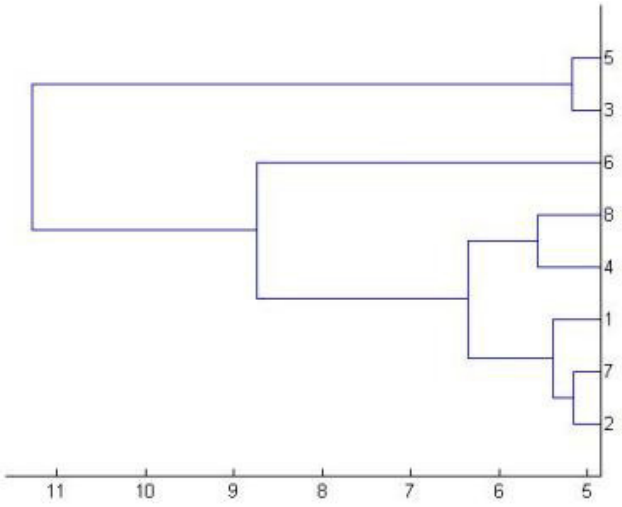

Figure 5: Hierarchical clustering global dendrogram output.

Consensus Clustering Analysis

top

This section describes the output generated by the consensus clustering analysis, which is similar to the standard clustering output except for 2 additional files. Two text output files are generated automatically, and the user has the option of also creating a text file containing the cluster statistics, and image files representing each cluster as a heatmap.

The consensus clustering algorithm performs the selected type of consensus clustering (see Tutorial), and generates a text file with all clustering results in the same format as that shown in Figure 3. If the 'Generate Heatmap' option is chosen, one heatmap for each cluster of appropriate size (see Tutorial) will be generated and saved as a Matlab .fig file; other image file formats may also be chosen (see Figure 4).

Text file: *myclusters-ClusterRuns.txt*

Consensus clustering also lets the user export the multiple clustering runs used to extract the consensus clusters to a text file (Figure 6). This is helpful if the raw clustering data ever needs to be re-analyzed. The first column always contains the gene IDs, followed by the clustering solutions. In Figure 6, 2 data sets were clustered (dset1 and dset2) 3 times each (rep1, rep2 and rep3) using Kmeans and Euclidean distance (ED) into 8 clusters. For each gene, the cluster number it was assigned to is indicated for each clustering solution column. This is the actual data used to calculate the consensus clusters.

	A	B	C	D	E	F	G
1	GeneID	Kmeans ED dset1rep1	Kmeans ED dset1rep2	Kmeans ED dset1rep3	Kmeans ED dset2rep1	Kmeans ED dset2rep2	Kmeans ED dset2rep3
2	1415710_at	5	2	1	2	6	2
3	1415725_at	3	4	8	3	4	7
4	1415728_at	3	8	5	7	8	1
5	1415733_a_at	3	8	8	3	4	7
6	1415735_at	3	8	8	3	4	7
7	1415768_a_at	5	2	1	2	6	2
8	1415771_at	3	8	8	7	8	1
9	1415772_at	2	4	6	6	5	7
10	1415773_at	2	4	6	7	8	1
11	1415774_at	2	4	8	7	8	1
12	1415785_a_at	3	8	8	7	8	1
13	1415793_at	3	8	8	7	8	1
14	1415802_at	2	4	2	3	4	7
15	1415807_s_at	3	8	8	3	4	7
16	1415829_at	6	7	6	6	5	7

Figure 6: Example of consensus clustering run output.

Text file: *myclusters_network.txt*

Finally, each consensus clustering run automatically exports a tab-delimited file containing data for visualizing the clusters as a network (Figure 7, see Tutorial for details), which is a new feature in version 3. Each row of data represents an edge in the network, and ALL POSSIBLE edges are included in the file and must be filtered to obtain clusters (see Tutorial). Column A (ID1) and Column B (ID2) indicate the two genes connected by an edge with the edges attribute 'NumClusteredTogether'. The edge attribute is important as it tells you how many times those two genes were found in the same cluster. The data shown in Figure 7 was generated from the same data in Figure 6; thus, 6 total clustering solutions were run, so the maximum number of times 2 genes can be found in the same cluster is 6—which represents 100% for this data. If one were to have more clustering solutions, like 30, then two genes would have to be found in the same cluster 30 times to be considered at the 100% threshold (see Tutorial for details). To view this file as a network, simply import it into a network visualization software like Cytoscape (Smoot, Ono et al. 2011).

	A	B	C
1	ID1	ID2	NumClusteredTogether
2	1415710_at	1415710_at	6
3	1415768_a_at	1415710_at	6
4	1415841_at	1415710_at	6
5	1415975_at	1415710_at	4
6	1415976_a_at	1415710_at	3
7	1416035_at	1415710_at	5
8	1416041_at	1415710_at	6
9	1416101_a_at	1415710_at	6
10	1428767_at	1415710_at	4
11	1428796_at	1415710_at	3
12	1428838_a_at	1415710_at	3
13	1429002_at	1415710_at	6
14	1429003_at	1415710_at	4
15	1429082_at	1415710_at	6

Figure 7: Example of cluster network output file.

Cluster Mapping Output top

The cluster mapping function outputs one tab-delimited text file that describes one clustering solution in terms of another. An example output file is shown in Figure 8

```
Solution1   Solution2:NumGenes
1      2:21
2      1:6     4:153
3      1:135   3:4
4      3:34
```

Figure 8: Cluster Mapping text output file.

The first column lists each cluster in the first solution. The rest of the columns list how many genes in each clustering solution1 are located in the corresponding clustering solution2. The file can be read as follows: the solution1 cluster #1 is composed of 21 genes from the solution2 cluster #2; the solution1 cluster #2 is composed of 6 genes from the solution2 cluster #1 and 153 genes from the solution2 cluster #4; the solution1 cluster 3 is composed of 135 genes from the solution2 cluster #1 and 4 genes from the solution2 cluster #3; and the solution1 cluster #4 is composed of 34 genes in solution2 cluster #3.

Generate Heatmap Output top

The heatmap generation function outputs *n* .fig files, one for each of the *n* clusters that are specified in the input file; other image file formats may also be chosen as well as the minimum cluster size (see Tutorial). These files are the same as those output by the standard clustering function, only the input clusters are pre-defined. Also, a text output file, similar to that shown in Figure 3 is output.

Cluster Statistics Output top

The ClusterStats function outputs one text file containing information about each cluster. An example file is shown in Figure 9.

```
General Cluster Information
Total number of clusters: 8
Total number of singleton clusters: 0
Average cluster size (excluding singletons): 49
Largest cluster size: 92 genes
Smallest cluster size: 14 genes


------------------------------
Cluster 1:
Average Euclidean Distance Score:    1.64
Number of genes in cluster:   92
Average Expression Profile:    0.08  0.53  1.35  2.18  1.94
Standard Deviations:     0.46  0.58  0.58  0.44  0.81
Standard Errors:   0.05  0.06  0.06  0.05  0.08
Count of positive expressions:      31    75    91    92    89
Count of negative expressions:      38    15    1     0     2
Count of zero expression:     23    2     0     0     1

------------------------------
Cluster 2:
Average Euclidean Distance Score:    1.64
Number of genes in cluster:   76
Average Expression Profile:    0.14  1.45  2.51  3.13  2.93
Standard Deviations:     0.38  0.61  0.48  0.49  0.81
Standard Errors:   0.04  0.07  0.06  0.06  0.09
Count of positive expressions:      36    76    76    76    76
Count of negative expressions:      21    0     0     0     0
Count of zero expression:     19    0     0     0     0

------------------------------
```

Figure 9: Cluster Statistics text output file.

The very first section is a global summary of the clusters contained in this analysis.

A description of each per-cluster element in the output file is below:

- *Average Euclidean Distance Score:* This score reflects the overall homogeneity of each cluster with respect to Euclidean distance (ED). ED looks for clusters that have highly similar levels of expression; thus, a lower ED score indicates higher homogeneity with respect to similar expression levels. Note that if the clusters were generated using Pearson's correlation coefficient then the ED score will most likely be high since correlation clusters are not homogeneous with respect to Euclidean distance.
- *Number of genes in cluster:* The size of the cluster.
- *Average Expression Profile:* The expression values in each condition/column were averaged to obtain an average profile for each cluster.
- *Standard Deviations:* The standard deviation for each condition/column in a cluster is calculated. This is a measure of the variance at each time point. Lower StdDev indicate more homogeneity for a given condition.
- *Standard Errors:* The StdDev for each condition was divided by the total number of genes in the cluster to obtain the standard error.
- *Count of positive transcripts:* A count of how many transcripts were up-regulated for each condition (expression > 0).
- *Count of negative transcripts:* A count of how many transcripts were down-regulated for each condition (expression < 0).
- *Count of zero transcripts:* A count of how many transcripts showed no change for each condition (expression = 0).

References top

Smoot, M. E., K. Ono, et al. (2011). "Cytoscape 2.8: new features for data integration and network visualization." Bioinformatics **27**(3): 431-432.

Yeung, K. Y., D. R. Haynor, et al. (2001). "Validating clustering for gene expression data." Bioinformatics **17**(4): 309-318.