Transcriptograms and Differentially Expressed Modules of Leukemia Patients

Instalation

- Tools for transcriptogram: https://github.com/joseflaviojr/transcriptograma/wiki

Use Case: Leukemia Patients

- **Reference:** Macrae T, Sargeant T, Lemieux S, Hébert J et al. RNA-Seq reveals spliceosome and proteasome genes as most consistent transcripts in human cancer cells. PLoS One 2013;8(9):e72884.
- **Data:** http://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE4817
- **Patients:** 72 = 17 (Healthy) + 12 (ALL) + 43 (AML)

Main Input Files

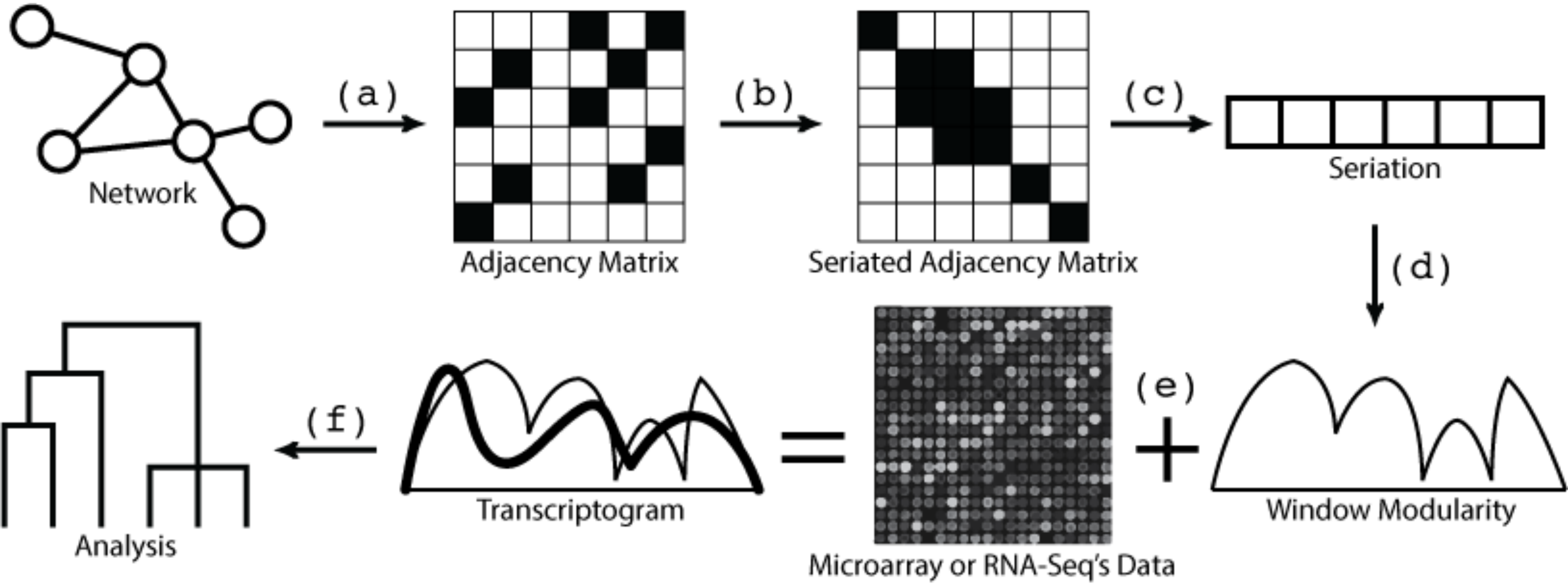
- **Expressions.txt** : RNA-Seq data / leukemia and healthy patients
- **Network.txt** : Human protein/gene network
- **EnsemblDB.txt** : Gene name map

- Source: https://github.com/joseflaviojr/transcriptograma/tree/master/UseCase-Leukemia

Protein/Gene Network

- **Reference:** Rolland T, Taşan M, Charloteaux B, et al. A proteome-scale map of the human interactome network. Cell. 2014;159(5):1212-1226. doi:10.1016/j.cell.2014.10.050.
- **Data:** http://interactome.dfci.harvard.edu/H_sapiens/
- **Name:** HI-II-14
- **Genes:** 4303
- **Edges:** 13685

Pipeline



a) Make the adjacency matrix
b) Seriate
c) Extract the seriation
d) Calculate the window modularity
e) Put the expression data and calculate the transcriptogram
f) Analysis the result

Step 1: Adjacency Matrix

- Generate the adjacency matrix of the network.
- Determine the initial sequence of the genes.

```
> MatrizAdjacencias.sh "Network.txt" nao tab
> mv "Network.txt.matriz.csv" "Network.csv"
> mv "Network.txt.nomes.txt" "Seriation_Initial_Genes.txt"
```

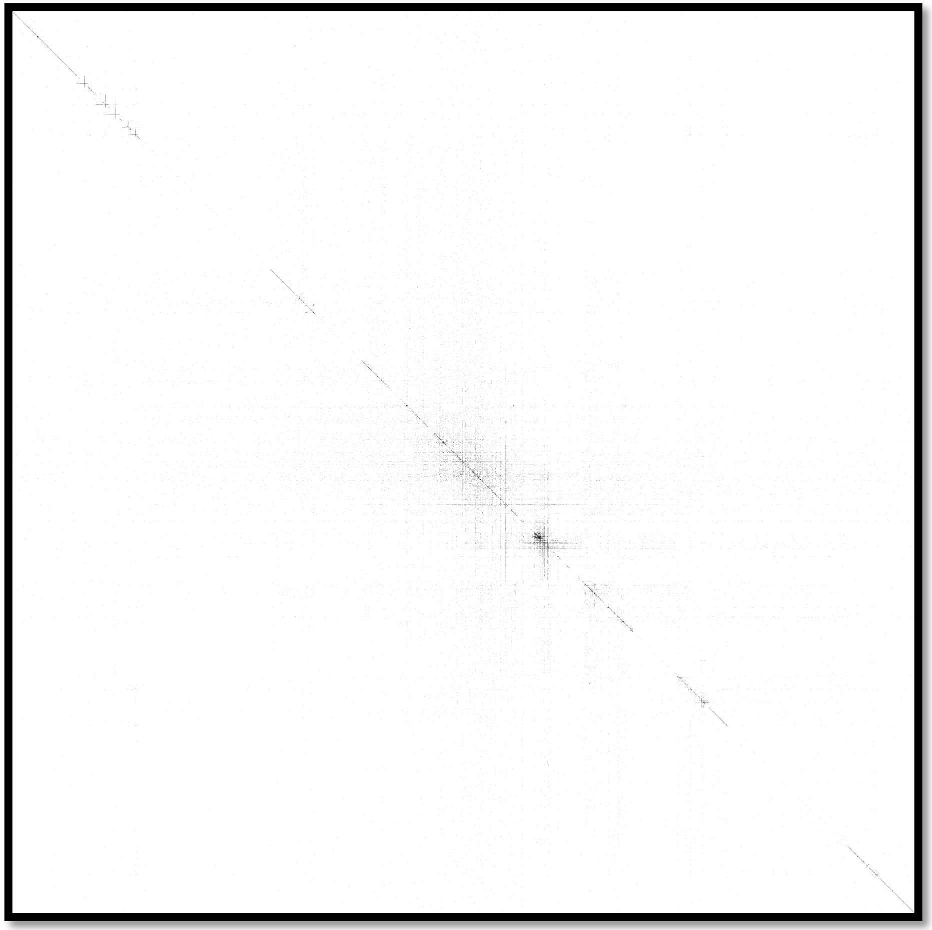
Step 2[INPUT]: Seriation

- Seriate the adjacency matrix with the Claritate algorithm – by 10 hours = 3600 seconds.
- Generate evoluation snapshots.
- Determine the final sequence of the genes.
- Get general informations about the seriated network.

```
> Experimento.sh "*.csv" 3600 96 1 Cla
> GerarImagensDeExperimento.sh .

> cp "Network.csv.CLA[1].ordem.txt" "Seriation_Final_Sequence.txt"
> ConverterNumerosParaNomes.sh "Seriation_Final_Sequence.txt"
  "Seriation_Initial_Genes.txt" > "Seriation_Final_Genes.txt"

> Informacao.sh "Network.csv" "Seriation_Final_Sequence.txt" >
  "Information.txt"
> GerarImagem.sh "Network.csv" "Seriation_Final_Sequence.txt"
  "Network.png"
```

Step 2[OUTPUT]: Seriation

- Seriated adjacency matrix of the network.
- Final sequence of the genes.



MAPK7, UBE2C, ASTE1, LIG4, XRCC4, ATP5A1, …

Step 3: Translation

- Translate, if necessary, the gene names to other identity patterns: Ensembl, Entrez, etc.

```
> ConverterGenes.sh Hs ALIAS2EG
  "Seriation_Final_Genes.txt"
  "Seriation_Final_Genes_Entrez.txt"


> TraduzirColuna.sh "Seriation_Final_Genes.txt" 1
  "EnsemblDB.txt" 1 3 "#" >
  "Seriation_Final_Genes_Entrez2.txt"
```

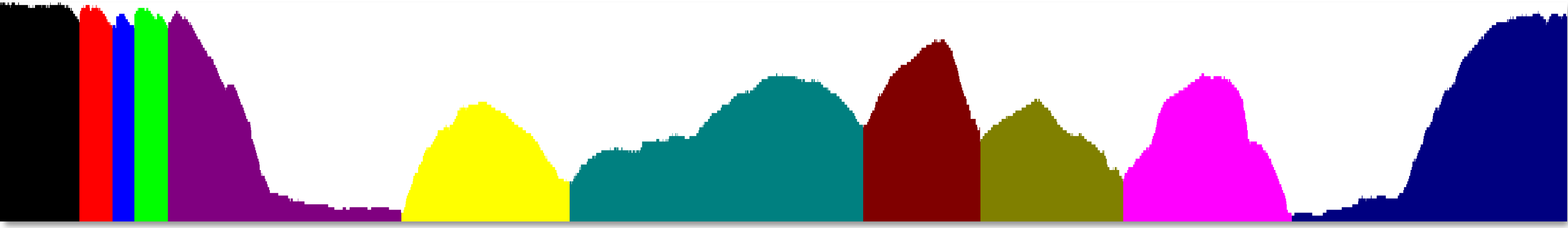
Step 4[INPUT]: Window Modularity

- Calculate modularities of the seriated network, identify borders and paint the modules.

```
> ModularidadeJanela.sh "Network.csv"
  "Seriation_Final_Sequence.txt" 251 > "WindowModularity.txt"
> ModularidadeDensidade.sh "Network.csv"
  "Seriation_Final_Sequence.txt" 60 > "DensityModularity.txt"
> GerarGrafico.sh "WindowModularity.txt" area 600 000000
  "WindowModularity.png"
> GerarGrafico.sh "DensityModularity.txt" area 600 000000
  "DensityModularity.png"
> Fronteiras.sh "WindowModularity.txt" 50 4 >
  "WindowModularity_Borders.txt"
> ColorirModulos.sh "WindowModularity.png"
  "WindowModularity_Colored.png" <
  "WindowModularity_Borders.txt"
```

Step 4[OUTPUT]: Window Modularity

- Modules of the related genes, based only on the seriated network.



Step 5: Quality Metrics

- Check the quality of the clusters/modules.

```
> Qualidade.sh "Network.txt" "Seriation_Final_Genes.txt"
  "WindowModularity_Borders.txt"
  "Silhouette,Dunn,Connectivity"
  "WindowModularity_Quality.txt"
> Silhouette.sh "Network.txt" "Seriation_Final_Genes.txt"
  "WindowModularity_Borders.txt"
  "WindowModularity_Silhouette.txt"
```

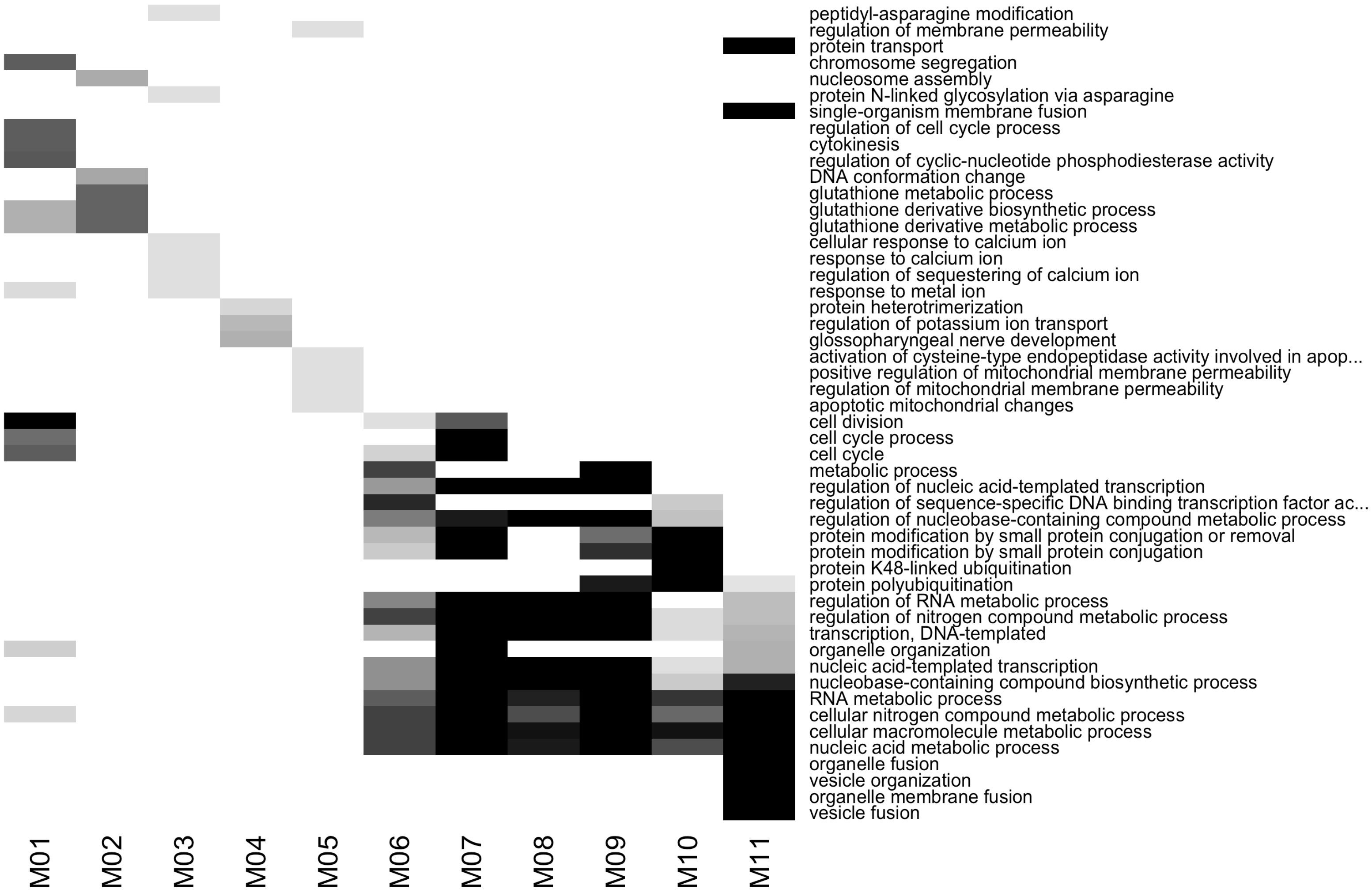
Step 6[INPUT]: Functional Enrichment

- Detect the biological functions of the modules, based on enrichment analysis of the Gene Ontology Consortium.

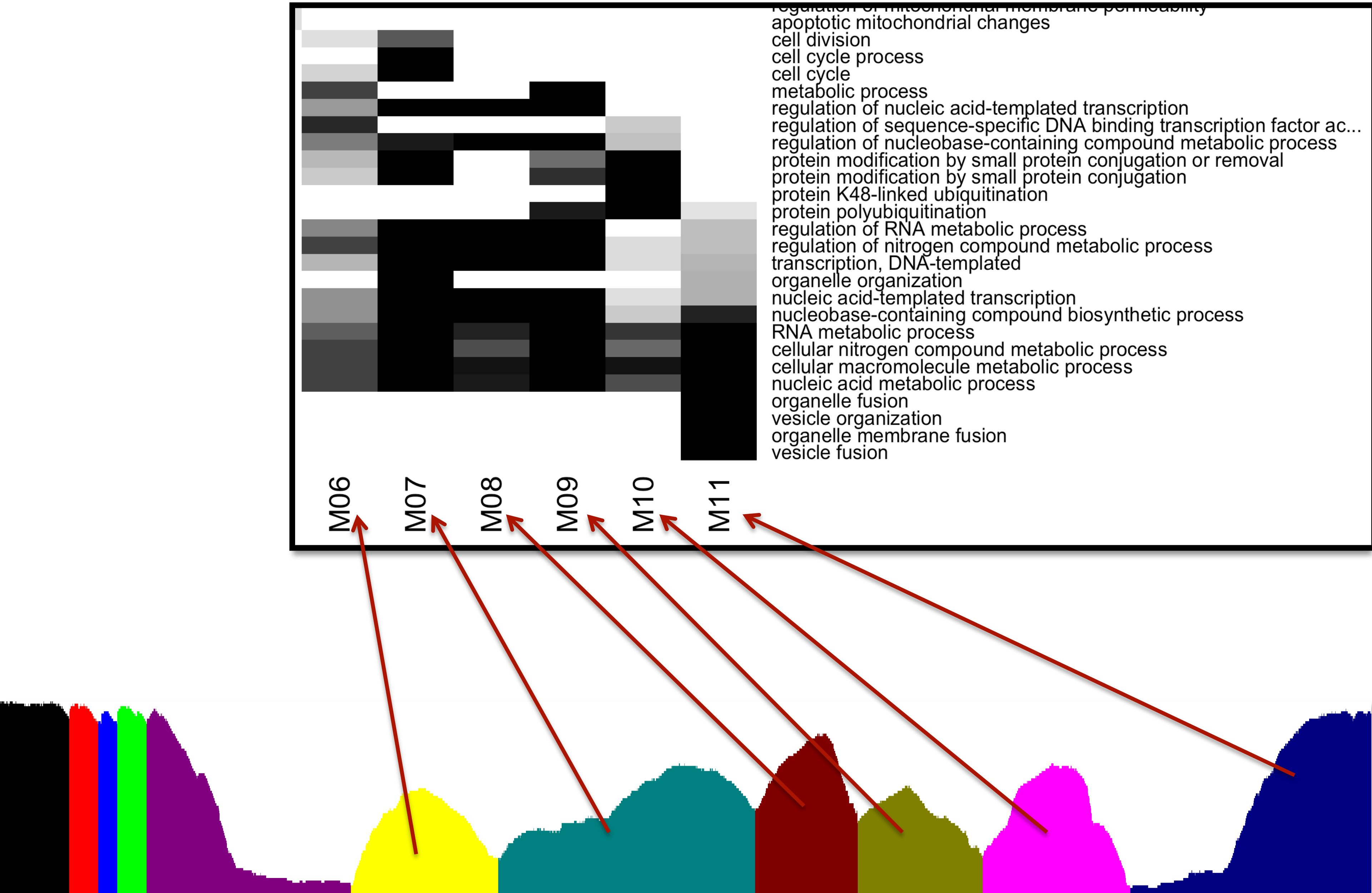
```
>  mkdir WindowModularity_Enrichment
>  cd WindowModularity_Enrichment
>  SepararModulos.sh "../Seriation_Final_Genes_Entrez.txt" "../
   WindowModularity_Borders.txt" "#" "M"
>  files=""
>  for f in M*txt; do
>     files="$files$f "
>  done
>  Enriquecer.sh Hs BP GeneOntology_BP.txt $files
>  Enriquecer.sh Hs MF GeneOntology_MF.txt $files
>  Enriquecer.sh Hs CC GeneOntology_CC.txt $files
>  Heatmap.sh GeneOntology_BP.txt M 50
>  Heatmap.sh GeneOntology_MF.txt M 50
>  Heatmap.sh GeneOntology_CC.txt M 50
>  cd ..
```

Step 6[OUTPUT]: Functional Enrichment

- BP enrichment's top result for window modules.



Step 6[OUTPUT]: Functional Enrichment

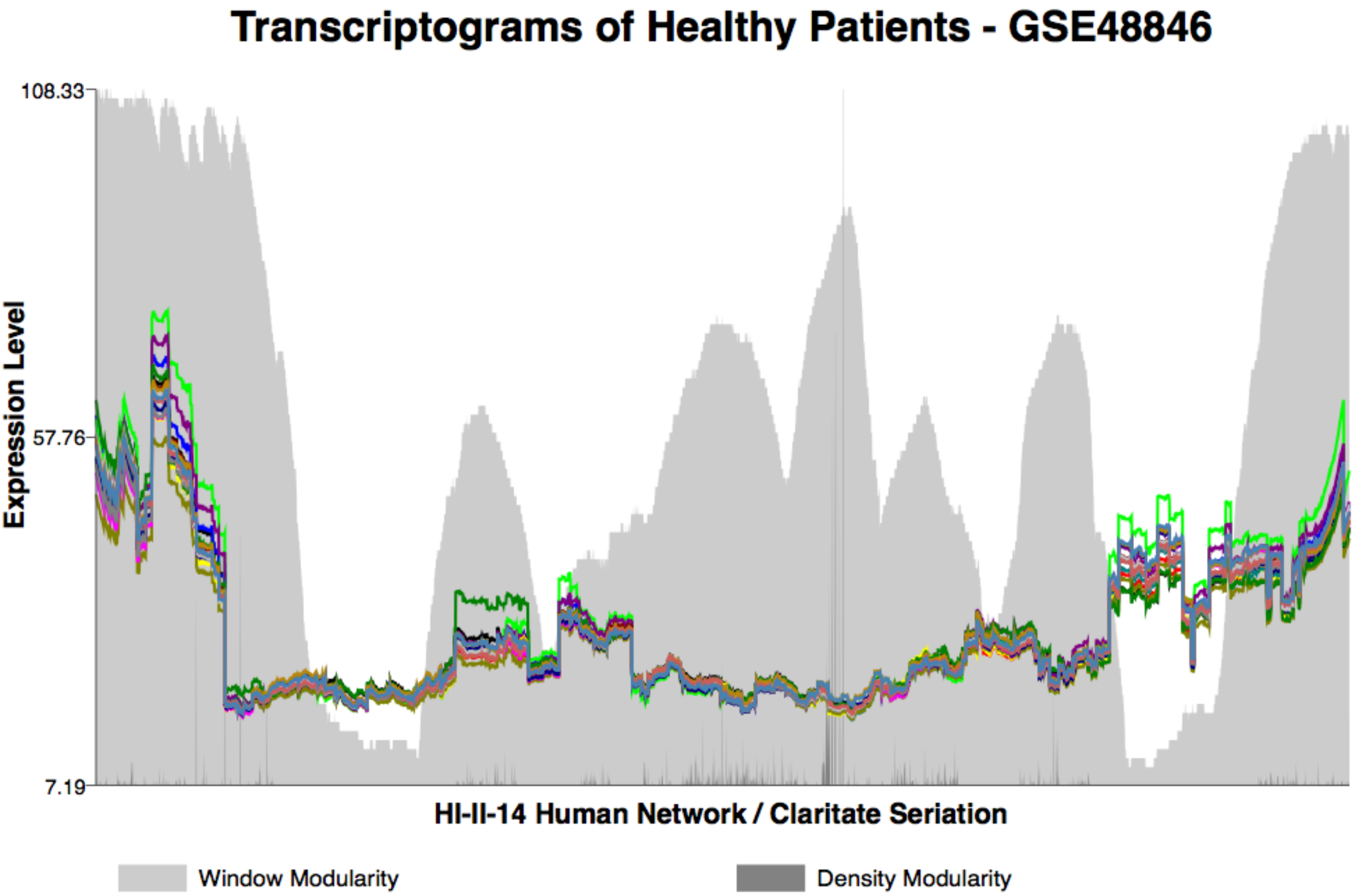


Step 7[INPUT]: Transcriptogram

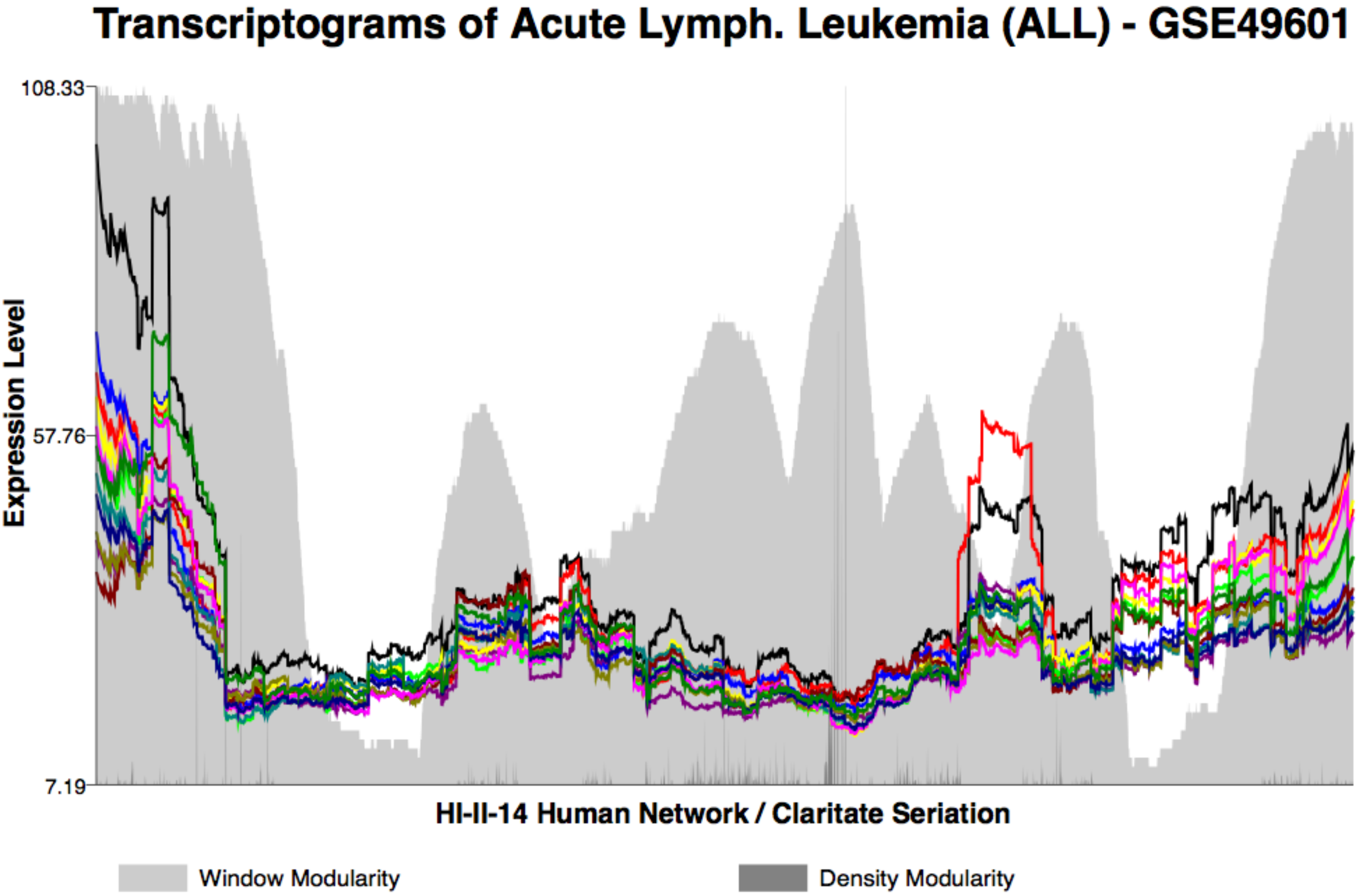
- Calculate and plot the transcriptograms of the patients.

```
> OrganizarTabela.sh "Seriation_Final_Genes.txt" "Expressions.txt" tab >
  "Expressions_Seriated.txt"

> Transcriptograma.sh "Expressions_Seriated.txt" 251 >
  "Transcriptograms.txt"

> GraficoTranscriptograma.sh "WindowModularity.txt"
  "DensityModularity.txt" "Transcriptograms.txt" "1-17"
  "Expressions_Labels.txt" "Chart_Healthy_Labels.txt" "Chart_Healthy.svg"
> GraficoTranscriptograma.sh "WindowModularity.txt"
  "DensityModularity.txt" "Transcriptograms.txt" "18-29"
  "Expressions_Labels.txt" "Chart_ALL_Labels.txt" "Chart_ALL.svg"
> GraficoTranscriptograma.sh "WindowModularity.txt"
  "DensityModularity.txt" "Transcriptograms.txt" "30-72"
  "Expressions_Labels.txt" "Chart_AML_Labels.txt" "Chart_AML.svg"
```

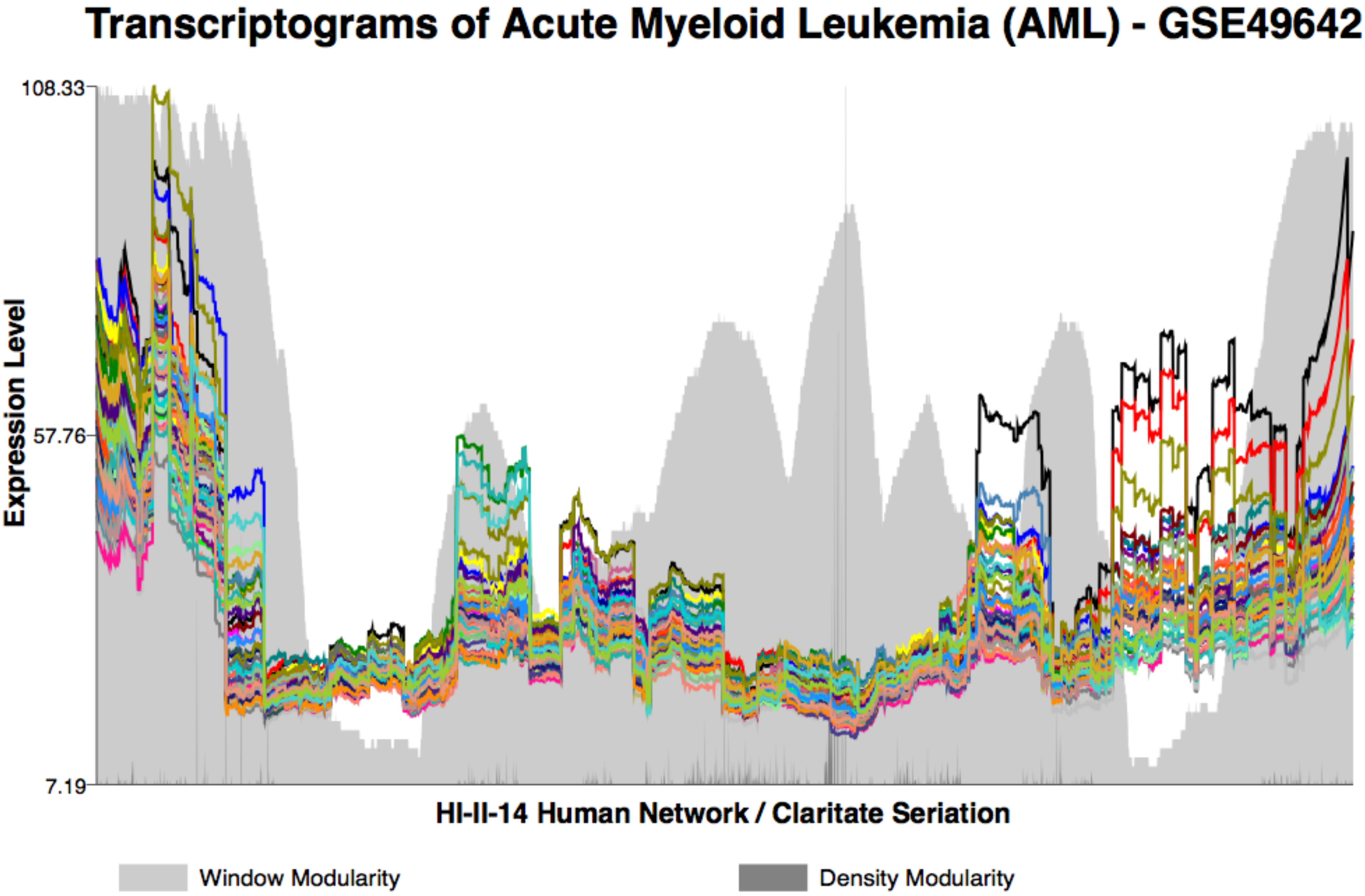
Step 7[OUTPUT]: Transcriptogram



Step 7[OUTPUT]: Transcriptogram



Step 7[OUTPUT]: Transcriptogram



Transcriptograms of Acute Myeloid Leukemia (AML) - GSE49642

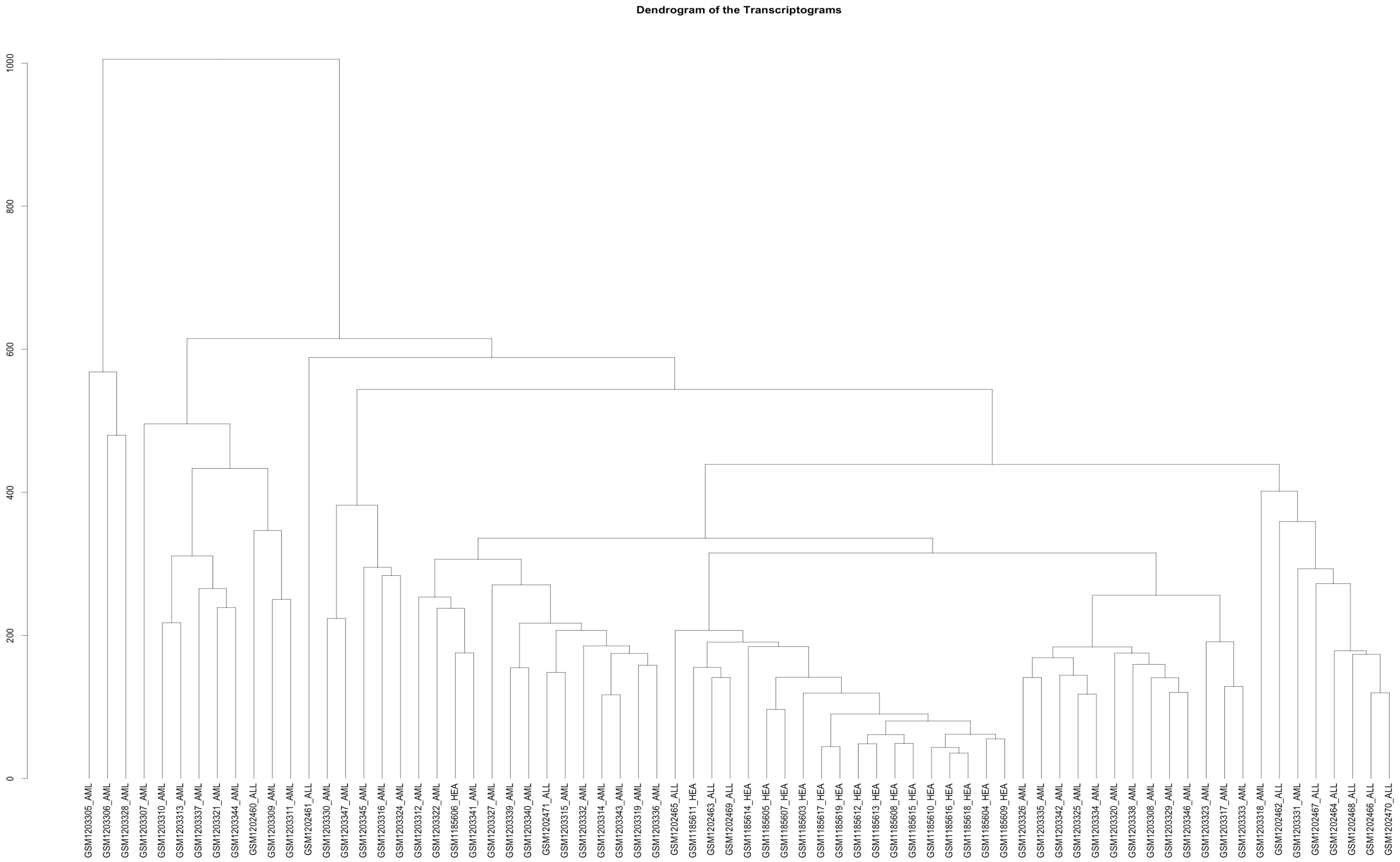
Step 8[INPUT]: Dendrogram

- Execute script to plot dendrograms of the patients.

```
> R -q --no-save -e "source('Dendrograms.R');"
```

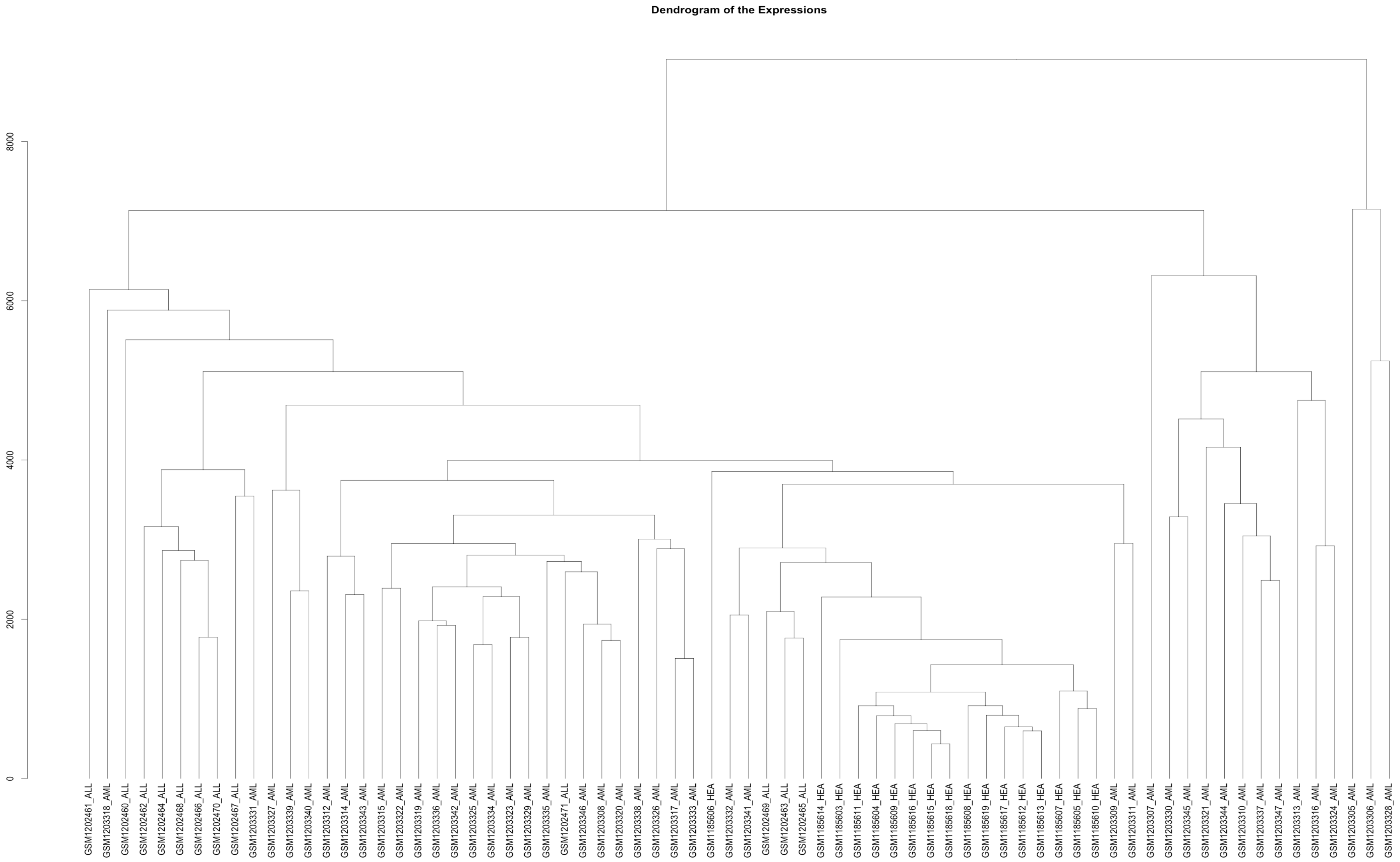
Step 8[OUTPUT]: Dendrogram

- Dendrogram of the patients based on its transcriptograms.



Step 8[OUTPUT]: Dendrogram

- Dendrogram of the patients based on its expressions.



Step 9[INPUT]: DEG

- Differentially Expressed Genes (DEG)
- ALL versus Healthy
- AML versus Healthy

```
> DEG.sh "Transcriptograms.txt" "DEG_ALL.txt" "1-17"
  "18-29”


> DEG.sh "Transcriptograms.txt" "DEG_AML.txt" "1-17"
  "30-72"
```

Step 9[OUTPUT]: DEG

- DEG ALL x Healthy – Top 10

```
index statistic   pvalue   gene
3554  5.65301188833966  4.59046631073878e-05 RFESD
3555  5.64786121828344  4.63451047695296e-05 ZNF224
3541  5.64143519355814  4.69007683987499e-05 PRR22
3540  5.64116599274164  4.69241969430767e-05 PDGFRA
3537  5.6373459045242   4.72579729804412e-05 NUDT22
3538  5.6364247921491   4.73388226822635e-05 TDRD7
3536  5.63559620286918  4.74116740529773e-05 PCTP
3535  5.63470685562561  4.7489997023531e-05  SREK1
3539  5.63315502537733  4.76269856117817e-05 NID2
3556  5.63066181312944  6.2001738832751e-05  RAD18
... And more
```

Step 9[OUTPUT]: DEG

- DEG AML x Healthy – Top 10

```
index statistic   pvalue   gene
931   -13.6610252098166 0  CASP7
932   -13.6147187598722 0  APOL1
884   -13.3959850475511 0  CDKN1B
882   -13.3938753304491 0  IL11
896   -13.3825079405206 0  CCDC130
929   -13.3638040101552 0  CARD10
930   -13.3518726386199 0  JMJD7
928   -13.3502276393306 0  FOXJ2
972   -13.3435712717987 0  BUD13
883   -13.3413873221972 0  FAM129A
... And more
```

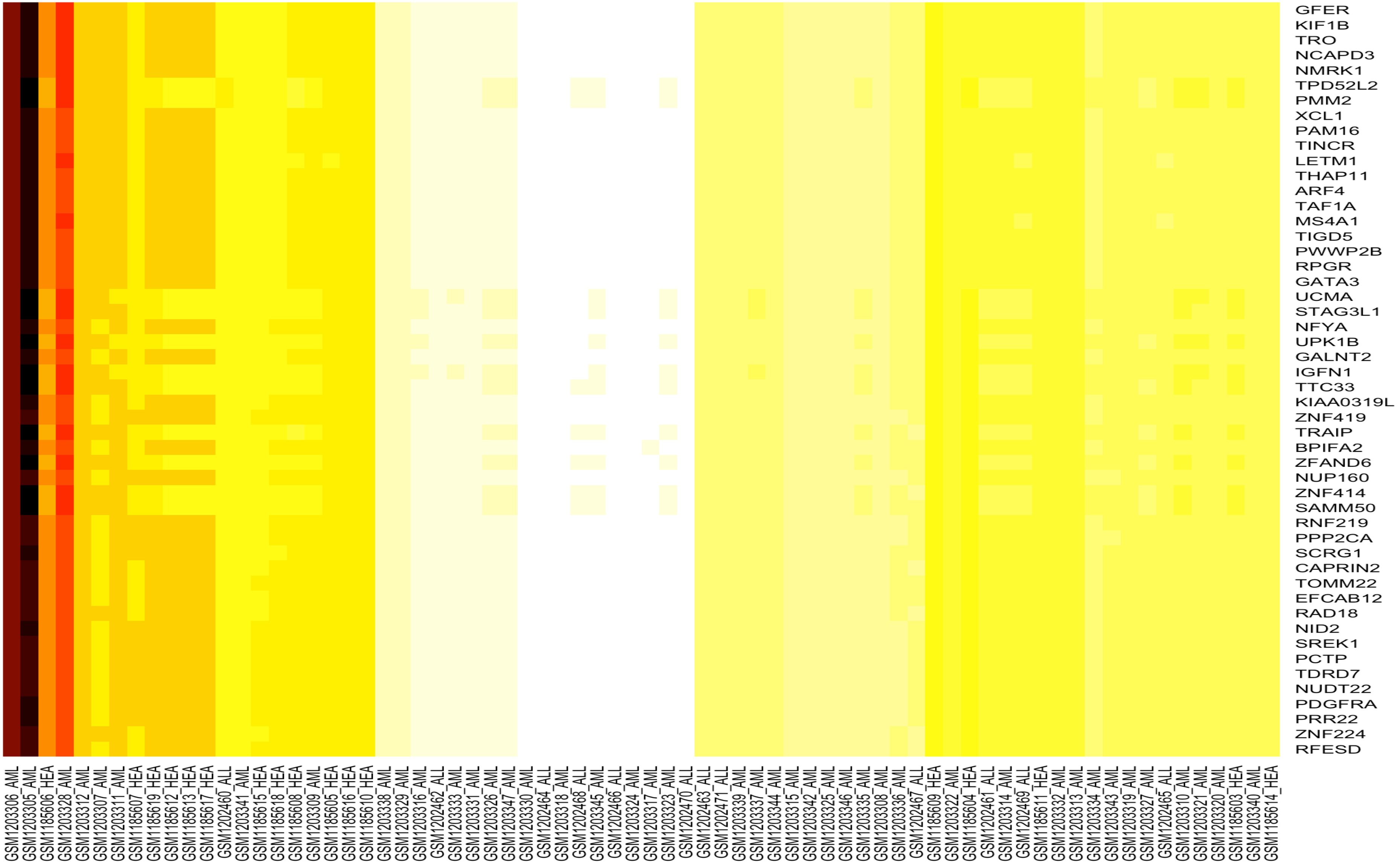
Step 10[INPUT]: DEG’s Heatmap

- Top DEG’s heatmap relative to the transcriptograms of the patients.

```
> HeatmapDEG.sh "Transcriptograms.txt"
  "Expressions_Labels.txt" "DEG_ALL.txt" "DEG_ALL.png”

> HeatmapDEG.sh "Transcriptograms.txt"
  "Expressions_Labels.txt" "DEG_AML.txt" "DEG_AML.png"
```

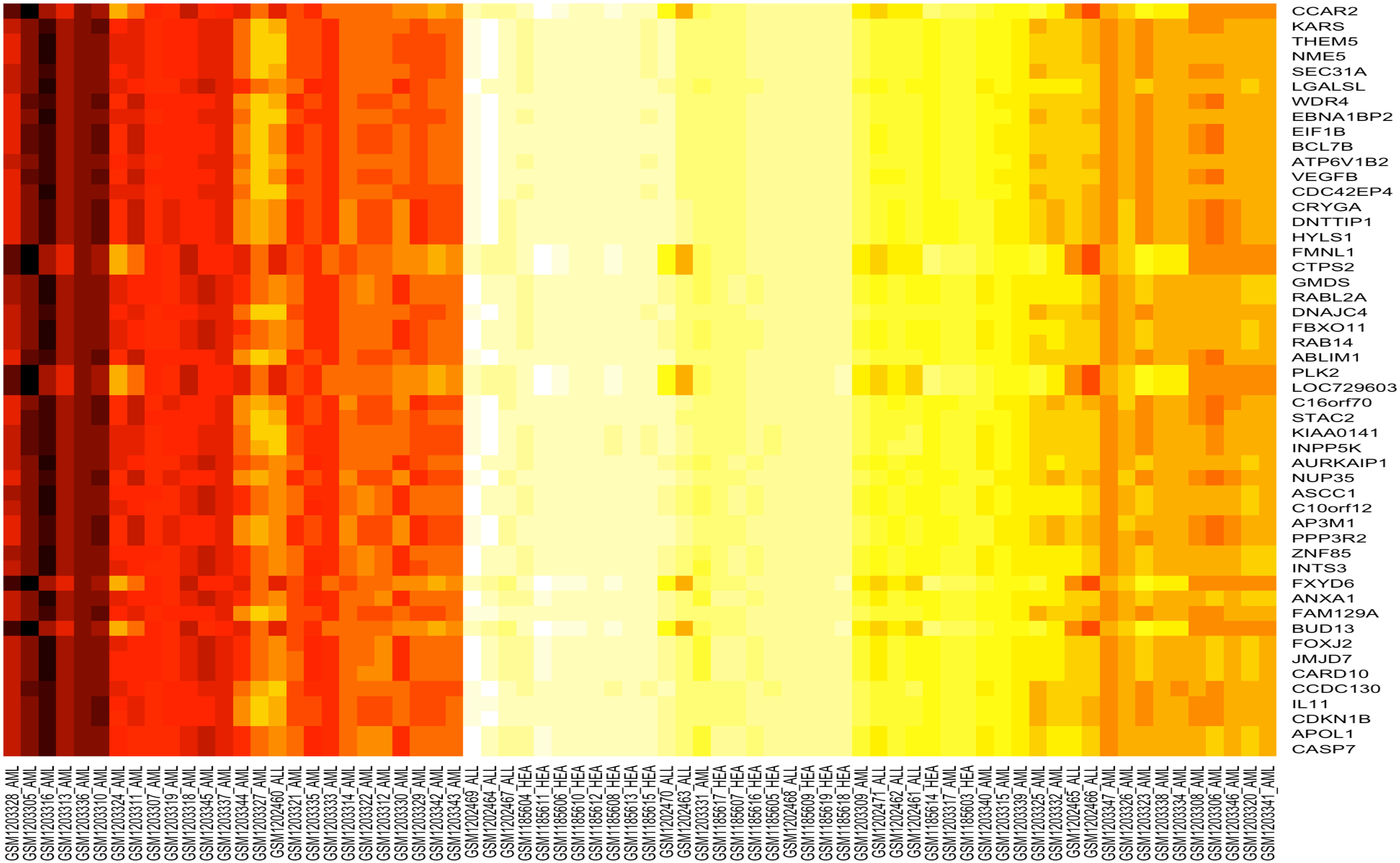
Step 10[OUTPUT]: DEG's Heatmap

- Top DEG of the ALL/Healthy in all transcriptograms.



Step 10[OUTPUT]: DEG's Heatmap

- Top DEG of the AML/Healthy in all transcriptograms.



Step 11: Seriation of the DEG

- Place the DEG’s results in the order of the network seriation.

```
> R -q --no-save -e "source('DEG_Seriated.R');"

> Normalizar.sh "DEG_Seriated_ALL_Values.txt" virg 1000 >
  "DEG_Seriated_ALL_Values_Norm.txt”

> Normalizar.sh "DEG_Seriated_AML_Values.txt" virg 1000 >
  "DEG_Seriated_AML_Values_Norm.txt"
```

Step 12: Differentially Expressed Modules

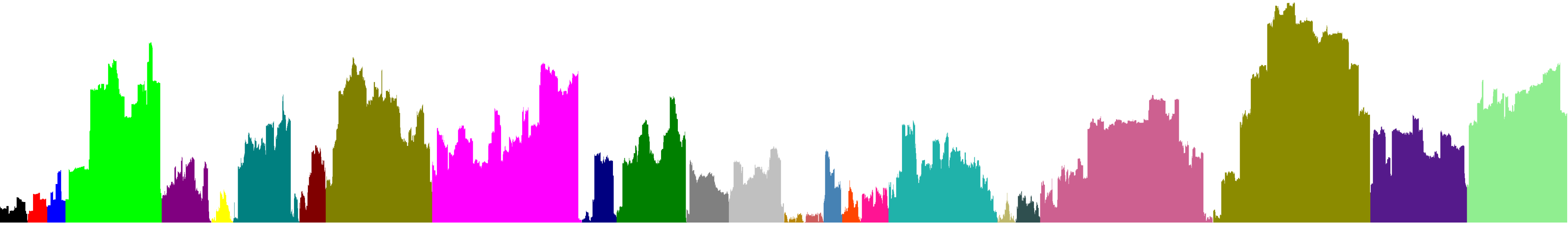
- Identify differentially expressed modules (DEM) based on DEG’s seriated results: ALL/Healthy and AML/Healthy.

```
> GerarGrafico.sh "DEG_Seriated_ALL_Values_Norm.txt" area 600
  000000 "DEG_Seriated_ALL.png"
> FronteirasDEG.sh "Seriation_Final_Genes.txt" "DEG_ALL.txt" S 4 2 S
  DESC 50 1000 "DEG_ALL_Borders.txt" "DEG_ALL_Modules.txt"
> ColorirModulos.sh "DEG_Seriated_ALL.png"
  "DEG_Seriated_ALL_Colored.png" < "DEG_ALL_Borders.txt"


> GerarGrafico.sh "DEG_Seriated_AML_Values_Norm.txt" area 600
  000000 "DEG_Seriated_AML.png"
> FronteirasDEG.sh "Seriation_Final_Genes.txt" "DEG_AML.txt" S 4 2 S
  DESC 50 1000 "DEG_AML_Borders.txt" "DEG_AML_Modules.txt"
> ColorirModulos.sh "DEG_Seriated_AML.png"
  "DEG_Seriated_AML_Colored.png" < "DEG_AML_Borders.txt"
```

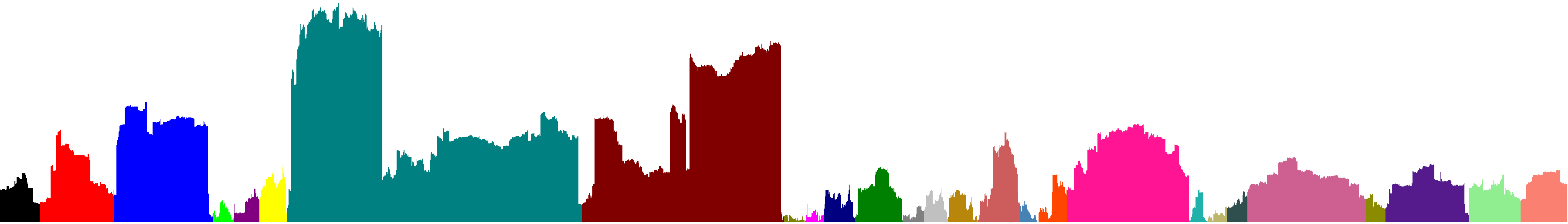
Step 12: Differentially Expressed Modules

The borders were calculated based on tail of the DEG's list.

- Seriated DEG and expressed modules of the ALL/Healthy.



- Seriated DEG and expressed modules of the AML/Healthy.



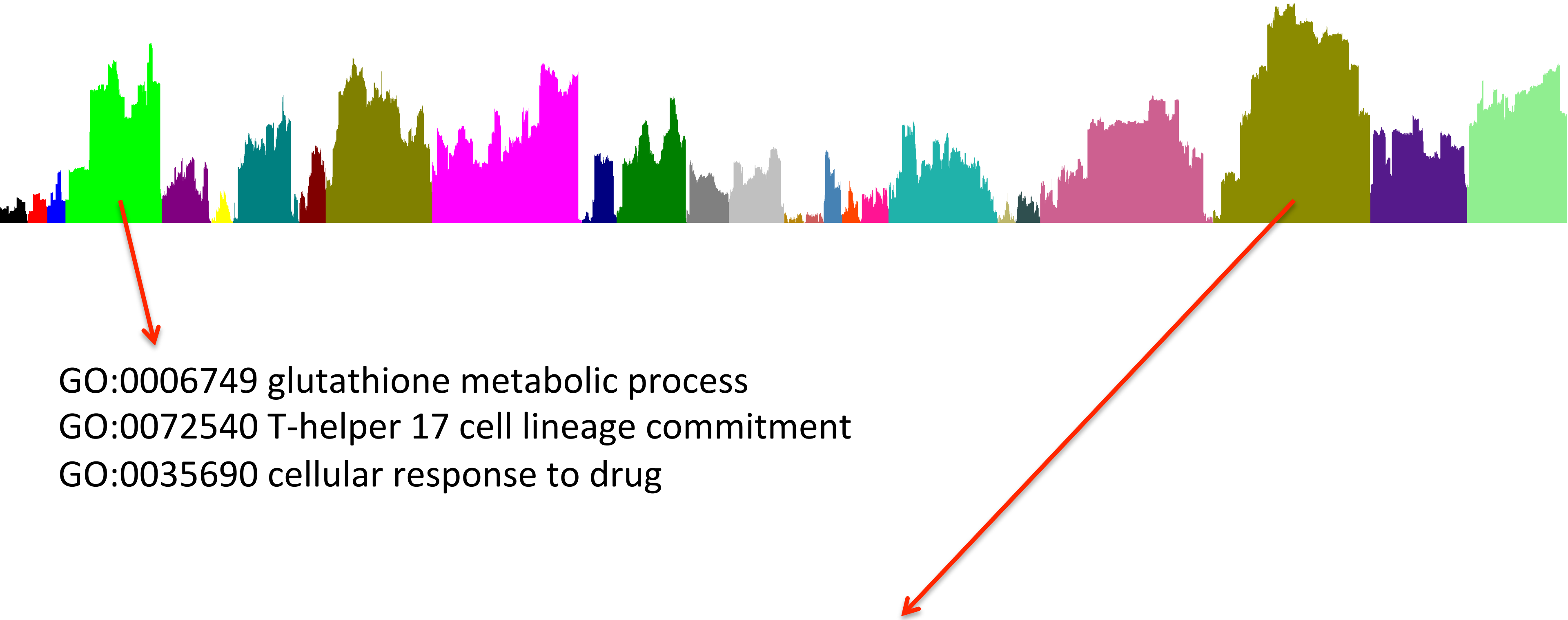
Step 12: Differentially Expressed Modules

- Enrichment of the expressed modules of the DEG ALL/ Healthy.

```
>   mkdir DEG_ALL_Enrichment
>   cd DEG_ALL_Enrichment

>   SepararModulos.sh "../Seriation_Final_Genes_Entrez.txt" "../DEG_ALL_Borders.txt" "#" "M"

>   files=""
>   for f in M*txt; do
>       files="$files$f "
>   done

>   Enriquecer.sh Hs BP GeneOntology_BP.txt $files
>   Enriquecer.sh Hs MF GeneOntology_MF.txt $files
>   Enriquecer.sh Hs CC GeneOntology_CC.txt $files
>   Heatmap.sh GeneOntology_BP.txt M 50
>   Heatmap.sh GeneOntology_MF.txt M 50
>   Heatmap.sh GeneOntology_CC.txt M 50
>   cd ..
```

Step 12: Differentially Expressed Modules

- Samples of the Biological Process (Gene Ontology) terms detected in the functional enrichment of the differentially expressed modules of the DEG ALL/Healthy.



GO:0006749 glutathione metabolic process
GO:0072540 T-helper 17 cell lineage commitment
GO:0035690 cellular response to drug

GO:0010467 gene expression
GO:0044260 cellular macromolecule metabolic process
GO:0008380 RNA splicing

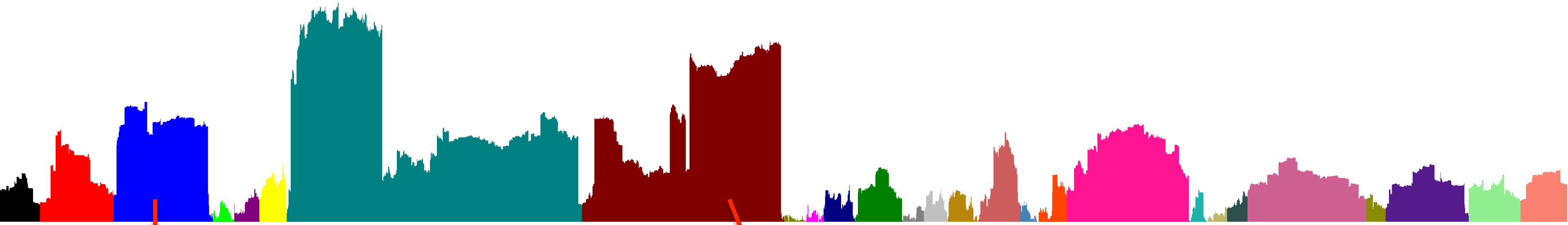
Step 12: Differentially Expressed Modules

- Enrichment of the expressed modules of the DEG AML/Healthy.

```
>   mkdir DEG_AML_Enrichment
>   cd DEG_AML_Enrichment

>   SepararModulos.sh "../Seriation_Final_Genes_Entrez.txt" "../DEG_AML_Borders.txt" "#"
    "M"

>   files=""
>   for f in M*txt; do
>      files="$files$f "
>   done

>   Enriquecer.sh Hs BP GeneOntology_BP.txt $files
>   Enriquecer.sh Hs MF GeneOntology_MF.txt $files
>   Enriquecer.sh Hs CC GeneOntology_CC.txt $files
>   Heatmap.sh GeneOntology_BP.txt M 50
>   Heatmap.sh GeneOntology_MF.txt M 50
>   Heatmap.sh GeneOntology_CC.txt M 50
>   cd ..
```

Step 12: Differentially Expressed Modules

- Samples of the Biological Process (Gene Ontology) terms detected in the functional enrichment of the differentially expressed modules of the DEG AML/Healthy.



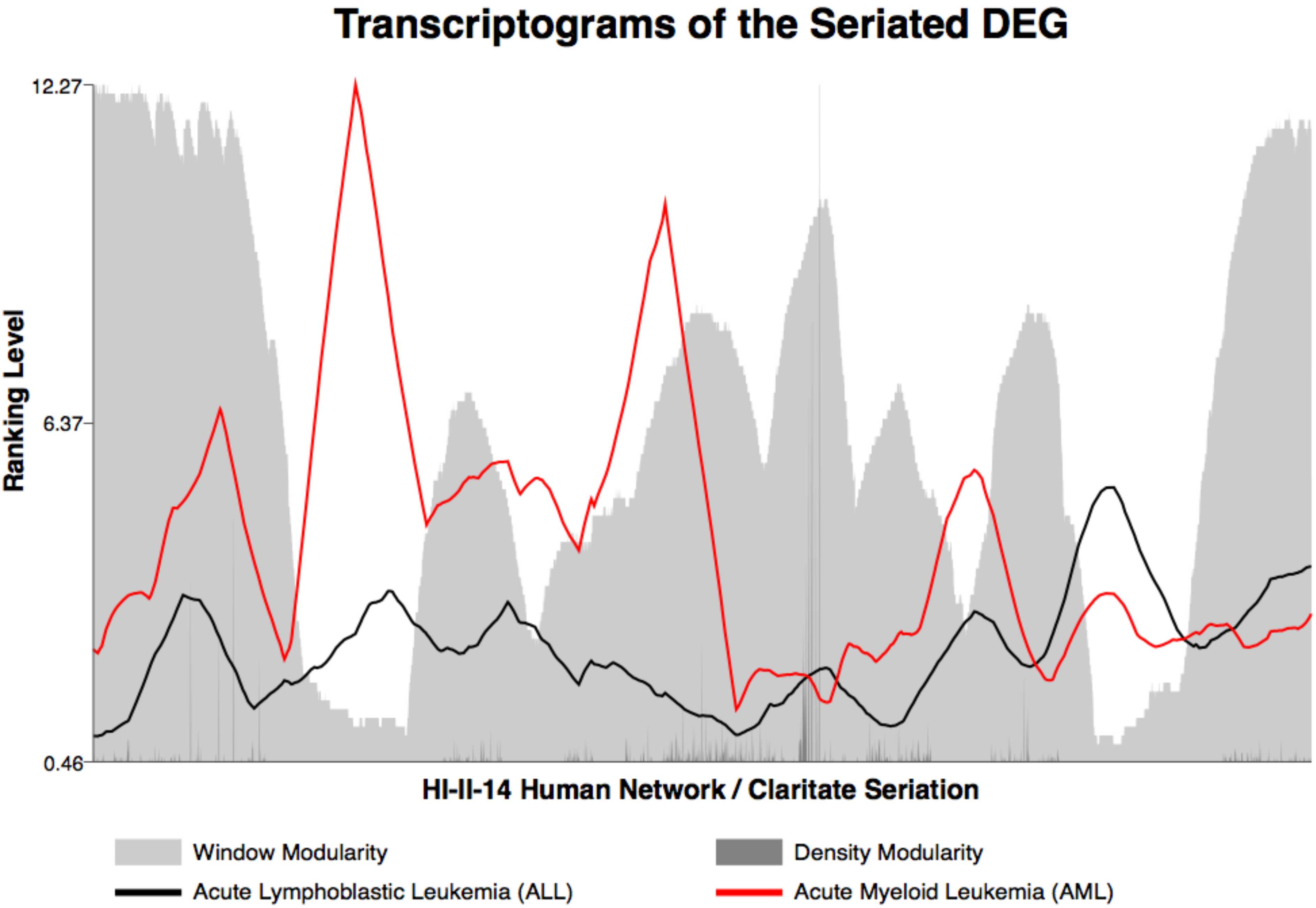
GO:0002376 immune system process
GO:0006952 defense response
GO:0070208 protein heterotrimerization

GO:0007049 cell cycle
GO:0006996 organelle organization
GO:0043933 macromolecular complex subunit organization

Step 13: Transcriptogram of the DEG

- Transcriptograms of the DEG’s results.

```
> Transcriptograma.sh "DEG_Seriated.txt" 251 >
  "Transcriptograms_DEG.txt”


> GraficoTranscriptograma.sh "WindowModularity.txt"
  "DensityModularity.txt" "Transcriptograms_DEG.txt" "0"
  "DEG_Labels.txt" "Chart_DEG_Labels.txt" "Chart_DEG.svg"
```


Transcriptograms of the Seriated DEG
Ranking Level
12.27
6.37
0.46
HI-II-14 Human Network / Claritate Seriation
Window Modularity
Density Modularity
Acute Lymphoblastic Leukemia (ALL)
Acute Myeloid Leukemia (AML)

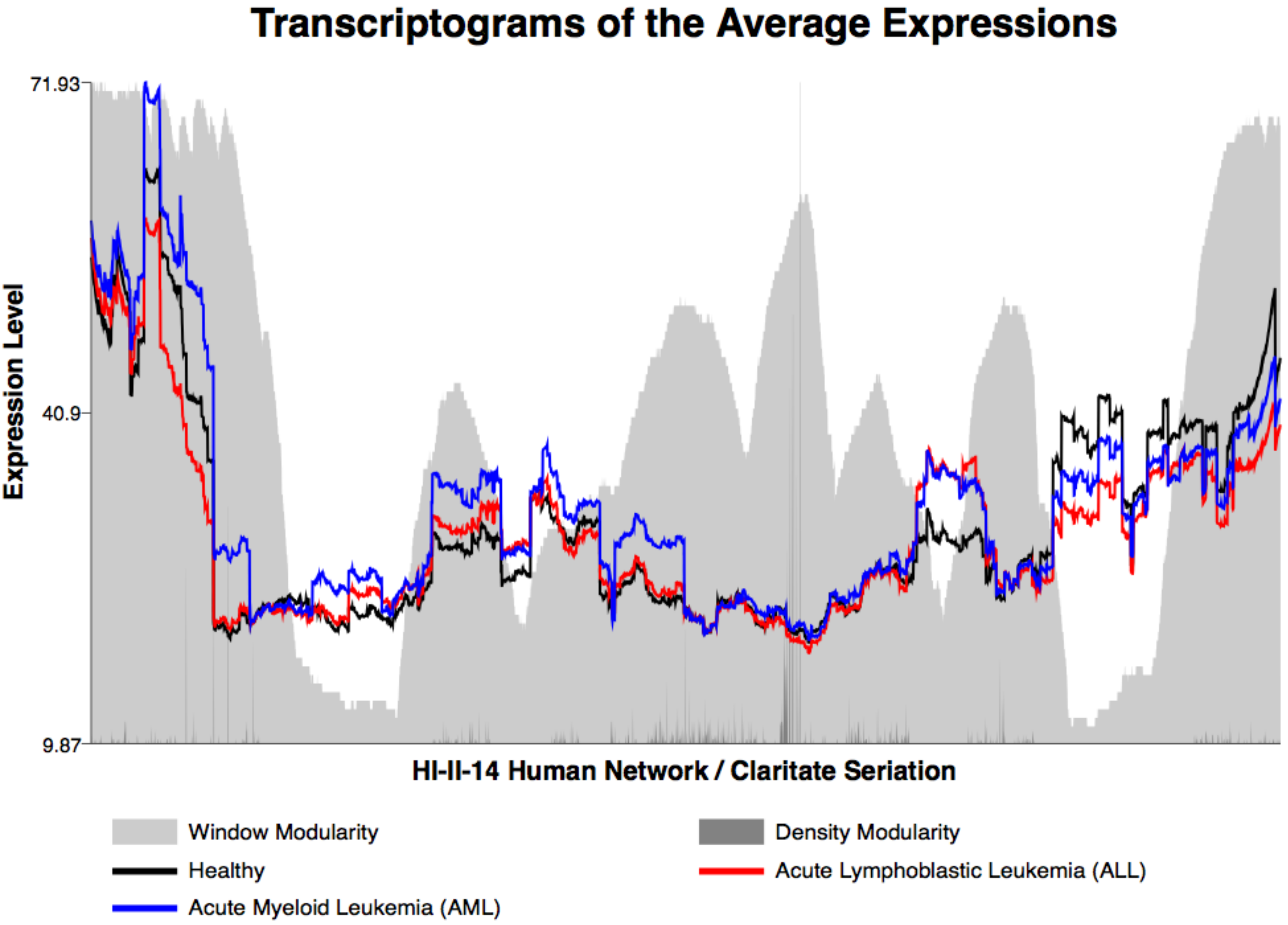
Step 14[INPUT]: Average Expression per Group

- Calculate, for each gene, the average expression per group: Healthy, ALL and AML.

```
> MediaPerfis.sh "Expressions_Seriated.txt" "1-17" "18-29"
  "30-72" > "Expressions_Average_Patients.txt”


> Transcriptograma.sh "Expressions_Average_Patients.txt"
  251 > "Transcriptograms_Average_Patients.txt"
> GraficoTranscriptograma.sh "WindowModularity.txt"
  "DensityModularity.txt"
  "Transcriptograms_Average_Patients.txt" "0"
  "Expressions_Average_Patients_Labels.txt"
  "Chart_Average_Patients_Labels.txt"
  "Chart_Average_Patients.svg"
```

Step 14[OUTPUT]: Average Expression per Group

```
GENE    HEALTHY             ALL                 AML

MAPK7   8.257422352941177   4.4434458333333335  4.733830232558139
UBE2C   4.7770247058823525  15.6066675          8.882123255813951
ASTE1   1.6349376470588235  3.3167900000000006  3.283910232558139
LIG4    3.201185882352941   8.675902500000001   3.4670174418604636
XRCC4   1.7636164705882351  2.7732858333333326  5.0195030232558135
ATP5A1  228.45422647058822  194.88605833333335  209.48216651162795
ASPH    3.9695064705882346  1.317643333333333   6.527040232558141
DRG2    18.303418235294117  19.32307            16.82014674418605
...
```


Transcriptograms of the Average Expressions
71.93
40.9
9.87
Expression Level
HI-II-14 Human Network / Claritate Seriation
Window Modularity
Density Modularity
Healthy
Acute Lymphoblastic Leukemia (ALL)
Acute Myeloid Leukemia (AML)

Step 15[INPUT]: Average Expression per Module

- Calculate the average expression per module of the “Window Modularity”, “DEG ALL” and “DEG AML”.

```
> MediaModulos.sh "Expressions_Average_Patients.txt"
  "WindowModularity_Borders.txt" >
  "Expressions_Average_Patients_WindowModules.txt"
> MediaModulos.sh "Expressions_Average_Patients.txt"
  "DEG_ALL_Borders.txt" >
  "Expressions_Average_Patients_ALLModules.txt"
> MediaModulos.sh "Expressions_Average_Patients.txt"
  "DEG_AML_Borders.txt" >
  "Expressions_Average_Patients_AMLModules.txt"
```

Step 15[OUTPUT]: Average Expression per Module

- Average expression of the detected modules in the Window Modularity.

```
MOD HEALTHY             ALL                 AML

M1  49.198809995978685  47.99625778945623   52.872376891489324
M2  40.66683248181029   38.831559164403515  44.004647697876806
M3  89.75433565582885   60.02517477660731   99.06158404247908
M4  16.53933628272068   16.53747410623043   37.10427718146631
M5  13.609250737456074  14.315416817116182  15.259160548386612
M6  19.83442048891097   21.808060232280287  24.111098659150215
M7  17.956391222271026  18.28605997955249   20.89736886164982
M8  13.818872900111504  12.818230305482638  14.051765608581094
M9  17.13925064570132   18.837447466342365  19.17593607016014
M10 21.226944115629006  23.093612777179622  23.197743093858982
M11 35.76070972723684   29.38175253159843   32.085415419105146
```