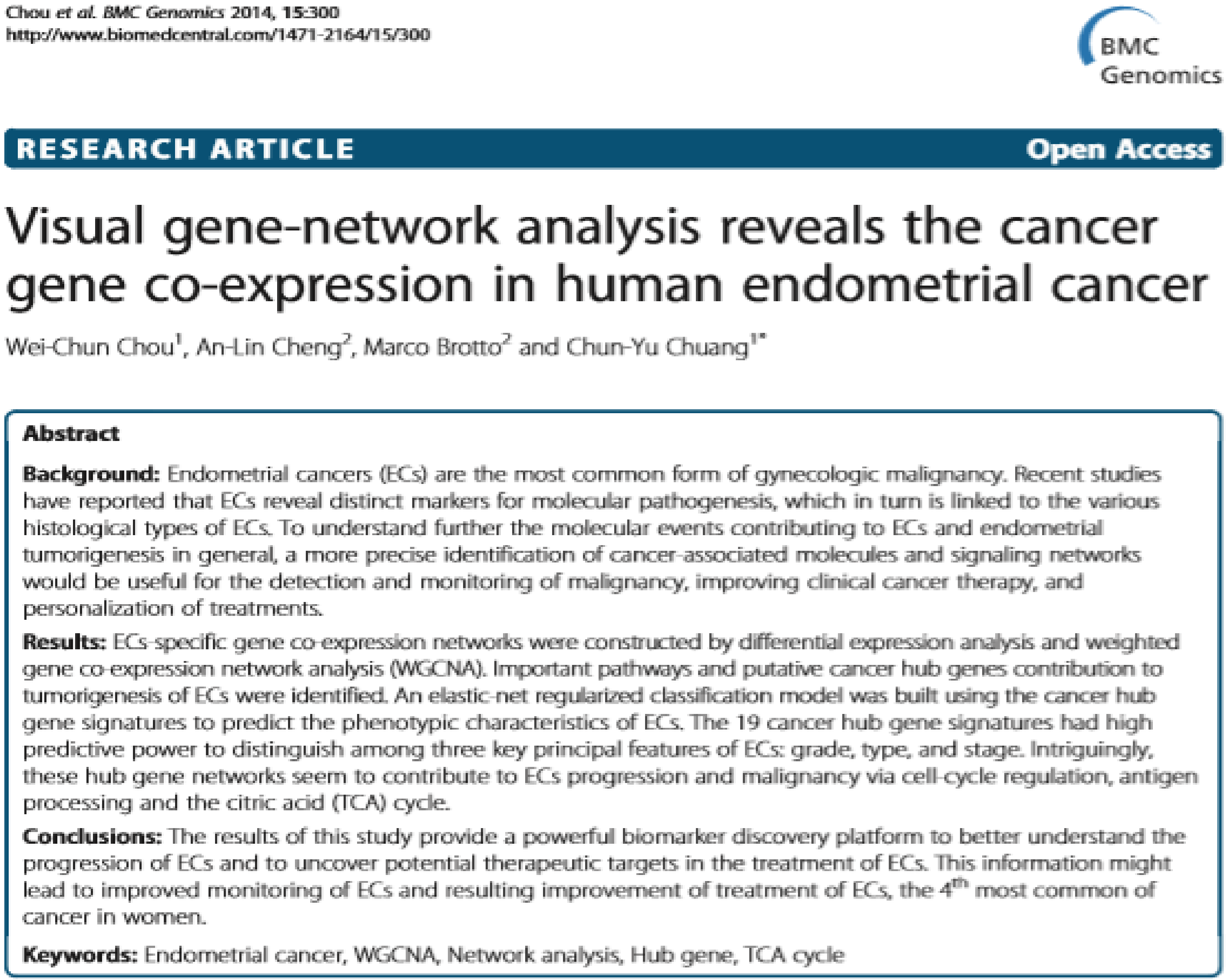
Co-expression (WGCNA) based Network analysis

GBIO0002

Archana Bhardwaj

University of Liege

Co expression Networks in cancer science



Chou *et al. BMC Genomics* 2014, **15**:300
http://www.biomedcentral.com/1471-2164/15/300

BMC Genomics

RESEARCH ARTICLE Open Access

Visual gene-network analysis reveals the cancer gene co-expression in human endometrial cancer

Wei-Chun Chou[1], An-Lin Cheng[2], Marco Brotto[2] and Chun-Yu Chuang[1*]

Abstract

Background: Endometrial cancers (ECs) are the most common form of gynecologic malignancy. Recent studies have reported that ECs reveal distinct markers for molecular pathogenesis, which in turn is linked to the various histological types of ECs. To understand further the molecular events contributing to ECs and endometrial tumorigenesis in general, a more precise identification of cancer-associated molecules and signaling networks would be useful for the detection and monitoring of malignancy, improving clinical cancer therapy, and personalization of treatments.

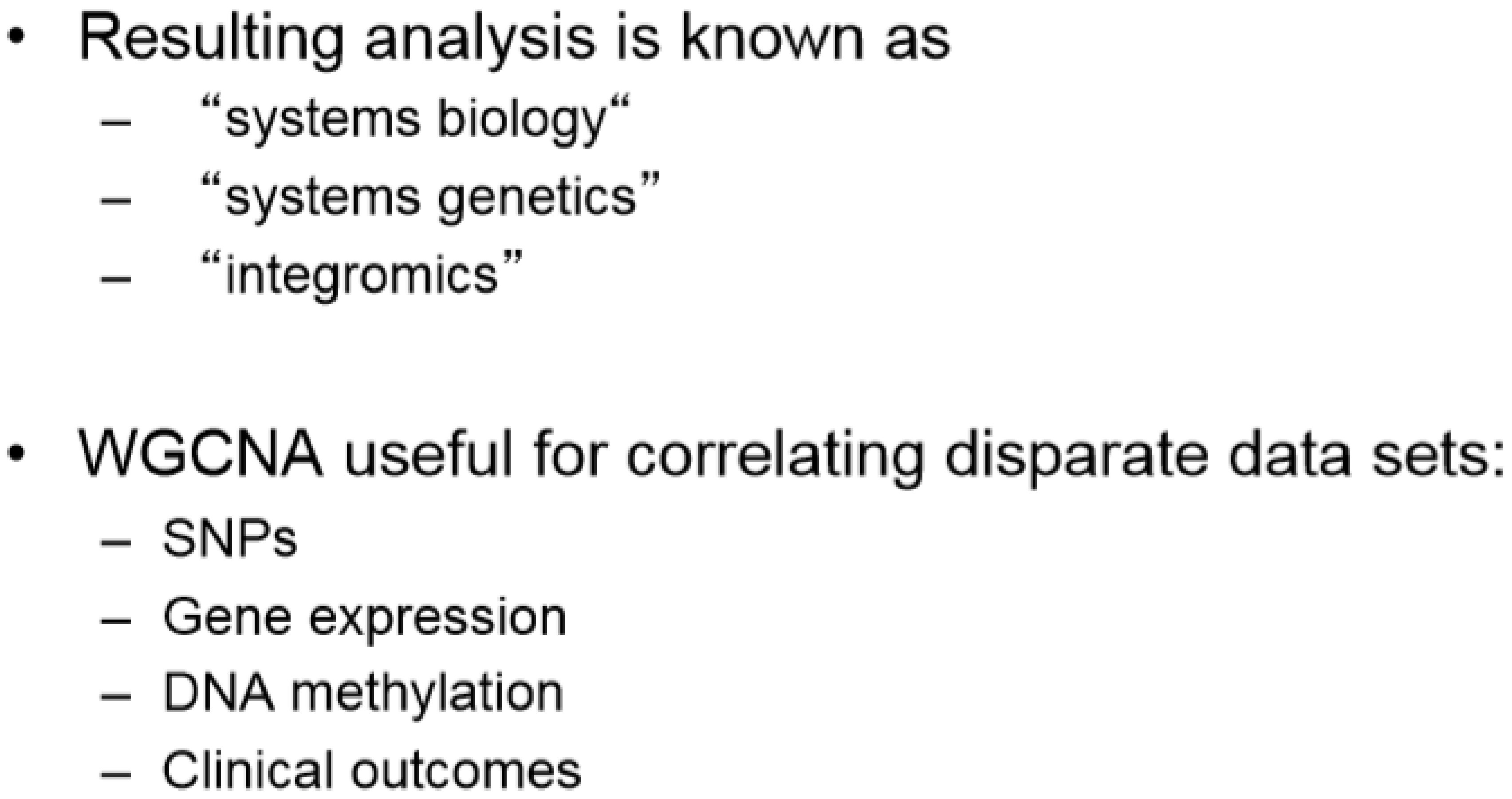
Results: ECs-specific gene co-expression networks were constructed by differential expression analysis and weighted gene co-expression network analysis (WGCNA). Important pathways and putative cancer hub genes contribution to tumorigenesis of ECs were identified. An elastic-net regularized classification model was built using the cancer hub gene signatures to predict the phenotypic characteristics of ECs. The 19 cancer hub gene signatures had high predictive power to distinguish among three key principal features of ECs: grade, type, and stage. Intriguingly, these hub gene networks seem to contribute to ECs progression and malignancy via cell-cycle regulation, antigen processing and the citric acid (TCA) cycle.

Conclusions: The results of this study provide a powerful biomarker discovery platform to better understand the progression of ECs and to uncover potential therapeutic targets in the treatment of ECs. This information might lead to improved monitoring of ECs and resulting improvement of treatment of ECs, the 4th most common of cancer in women.

Keywords: Endometrial cancer, WGCNA, Network analysis, Hub gene, TCA cycle

Weighted correlation network analysis (WGCNA)

Networks are particularly valuable for data integration



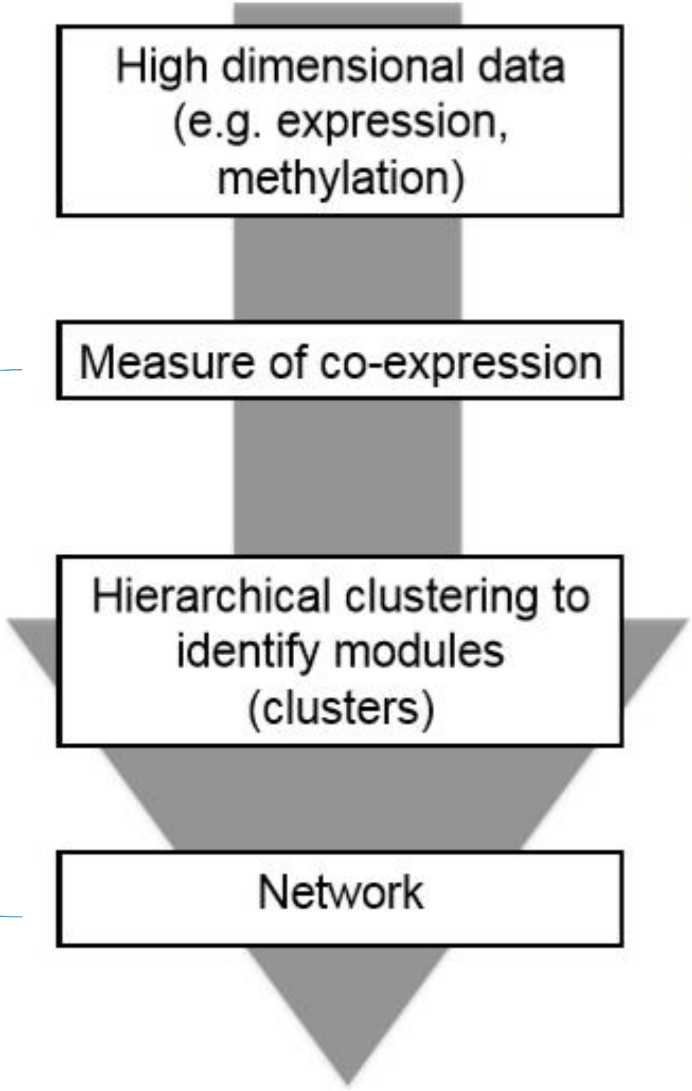
- Resulting analysis is known as
  - “systems biology“
  - “systems genetics”
  - “integromics”

- WGCNA useful for correlating disparate data sets:
  - SNPs
  - Gene expression
  - DNA methylation
  - Clinical outcomes

Constructing co-expression networks

Session I

Session II

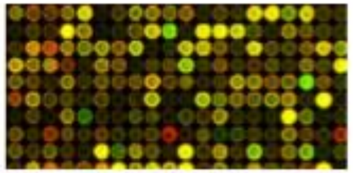


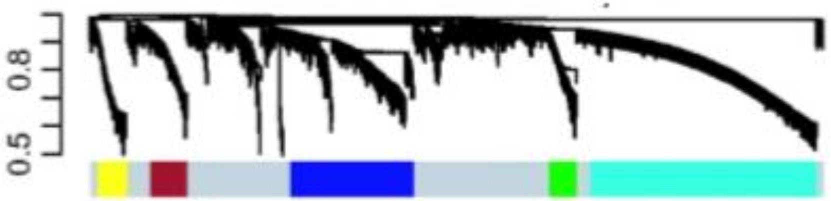
High dimensional data (e.g. expression, methylation)

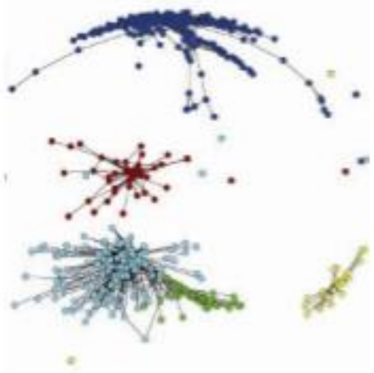
Measure of co-expression

Hierarchical clustering to identify modules (clusters)

Network







Session II : How to detect network modules?

Objective : Lets create own modules and check how genes are interacting with each other in 3D network

Module Definition

- ❑Based on the resulting cluster tree, we define modules as branches
- ❑Modules are either labeled by integers (1,2,3…) or equivalently by colors (turquoise, blue, brown, etc)
- ❑We often use average linkage hierarchical clustering coupled with the topological overlap dissimilarity measure.
- ❑Next we use the dynamic tree cutting method to define clusters. Langfelder et al 2007

Data Preparation

❑ **Set directory**

```
 workingDir = ".";
setwd(workingDir);
options(stringsAsFactors = FALSE);
```

❑ **Read sample data in csv format**

```
femData = read.csv("LiverFemale3600.csv");

dim(femData);
[1] 3600  143
```

← Check data dimensions

❑ **Prepare dataframe to extract expression data**

```
datExpr0 = as.data.frame(t(femData[, -c(1:8)]));
names(datExpr0) = femData$substanceBXH;
rownames(datExpr0) = names(femData)[-c(1:8)]
```

Checking data for excessive missing values and identification of outlier samples

```
gsg = goodSamplesGenes(datExpr0, verbose = 3);

gsg$allOK
```

If the last statement returns TRUE, all genes have passed the cuts. If not, we remove the offending genes and samples from the data

Samples clustering

❑ **clustering using hclust**

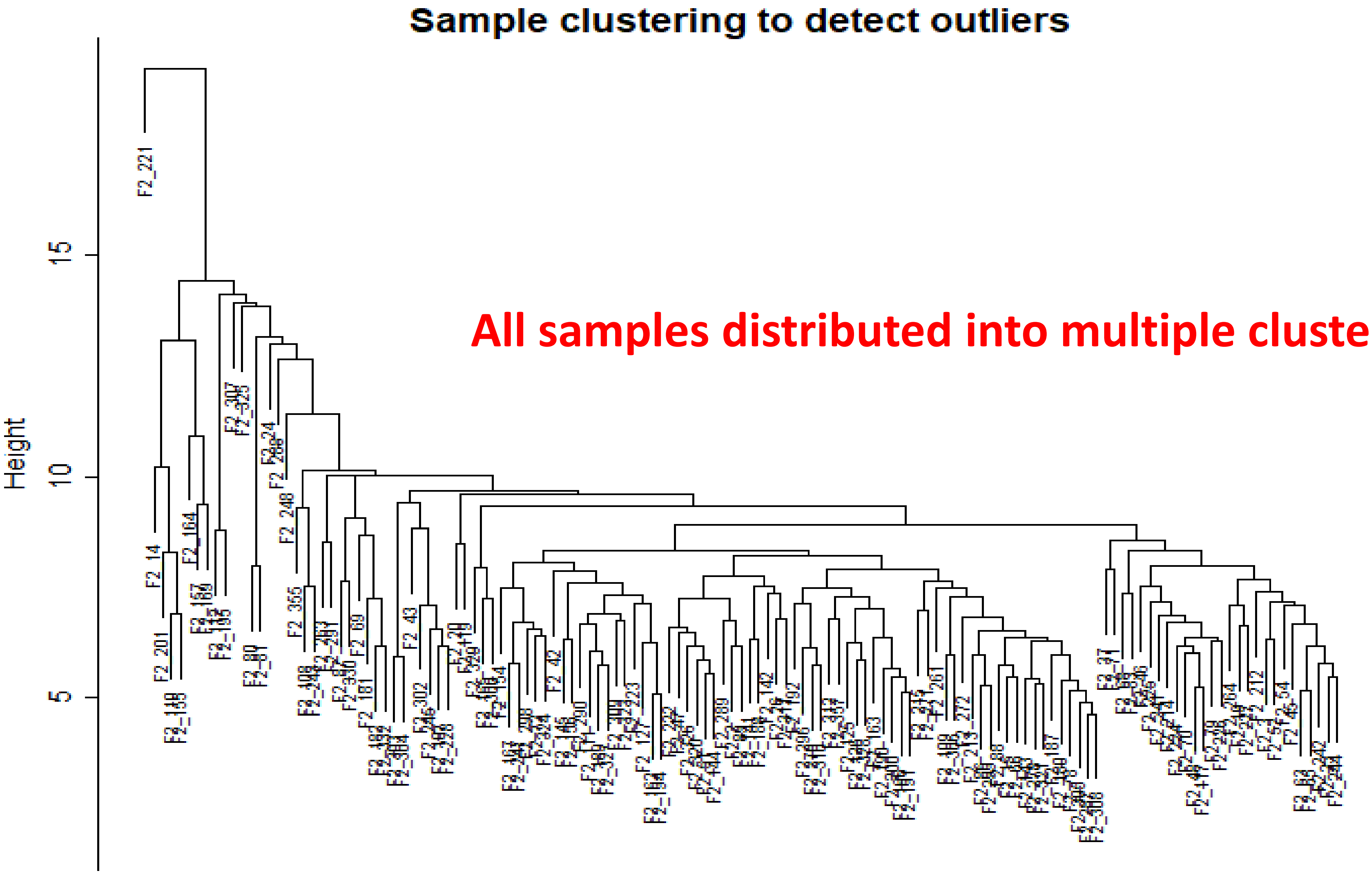
```
sampleTree = hclust(dist(datExpr0), method = "average");
```

❑ **Open a graphic output window of size 12 by 9 inches**

```
   sizeGrWindow(12,9)
par(cex = 0.6); par(mar = c(0,4,2,0))
```

❑ **Plot the sample tree**

```
plot(sampleTree, main = "Sample clustering to detect outliers", sub="", xlab="", cex.lab = 1.5, cex.axis = 1.5, cex.main = 2)
```



All samples distributed into multiple clusters

Remove outliers from data

- **Plot a line to show the cut**

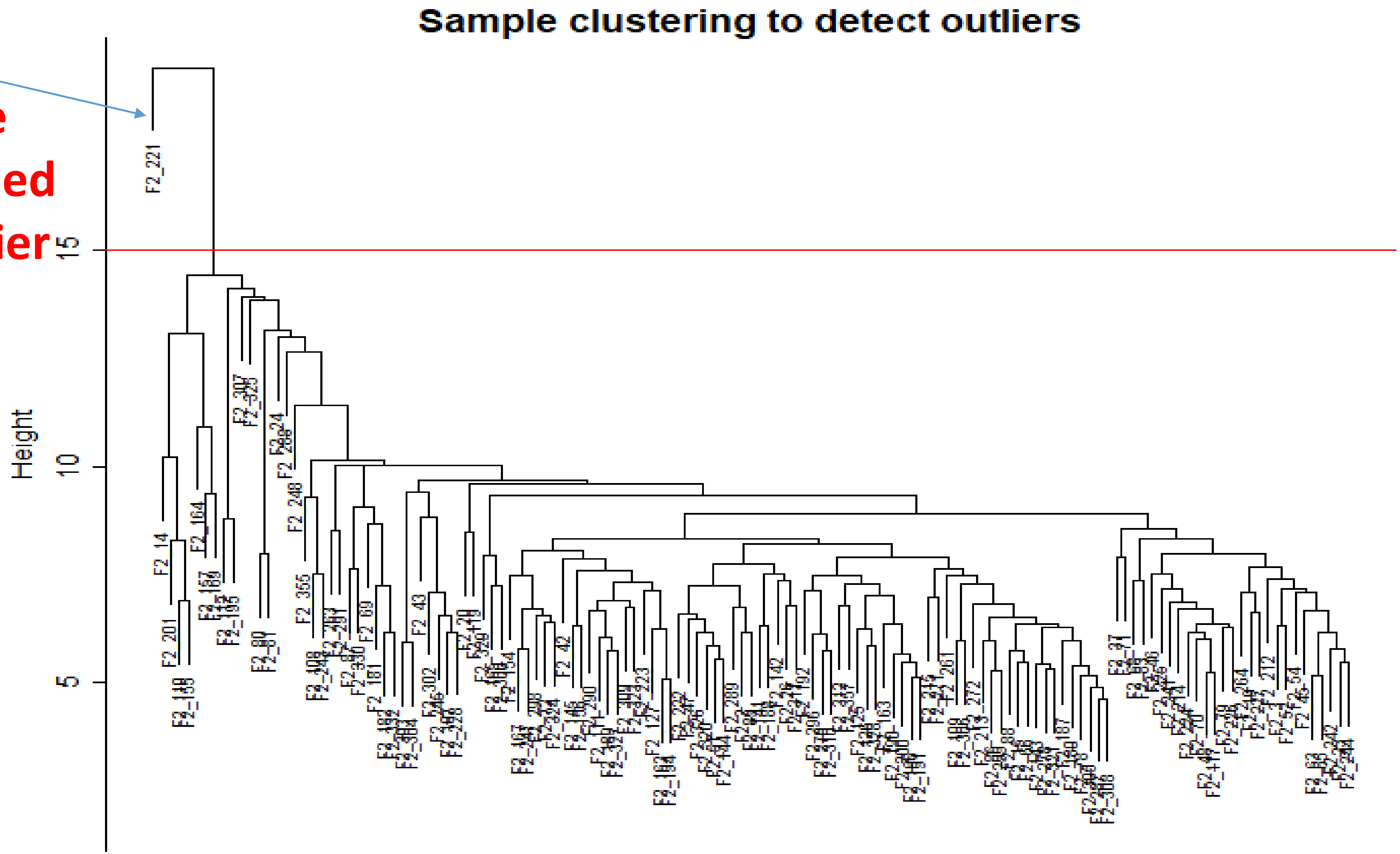
```
abline(h = 15, col = "red");
```

- **Determine cluster under the line**

```
clust = cutreeStatic(sampleTree, cutHeight = 15, minSize = 10)
table(clust)
```

- **clust 1 contains the samples we want to keep**

```
keepSamples = (clust==1)
datExpr = datExpr0[keepSamples, ]
nGenes = ncol(datExpr)
 nSamples = nrow(datExpr)
```



Sample clustering to detect outliers

Sample identified as outlier

Selection of soft thresholding power

- ❑ **Choose a set of soft-thresholding powers**

```
powers = c(c(1:10), seq(from = 12, to=20, by=2))
```

- ❑ **Call the network topology analysis function**

```
sft = pickSoftThreshold(datExpr, powerVector = powers, verbose = 5)
```

- ❑ **Plot the results: sizeGrWindow(9, 5)**

```
par(mfrow = c(1,2));
cex1 = 0.9;
```

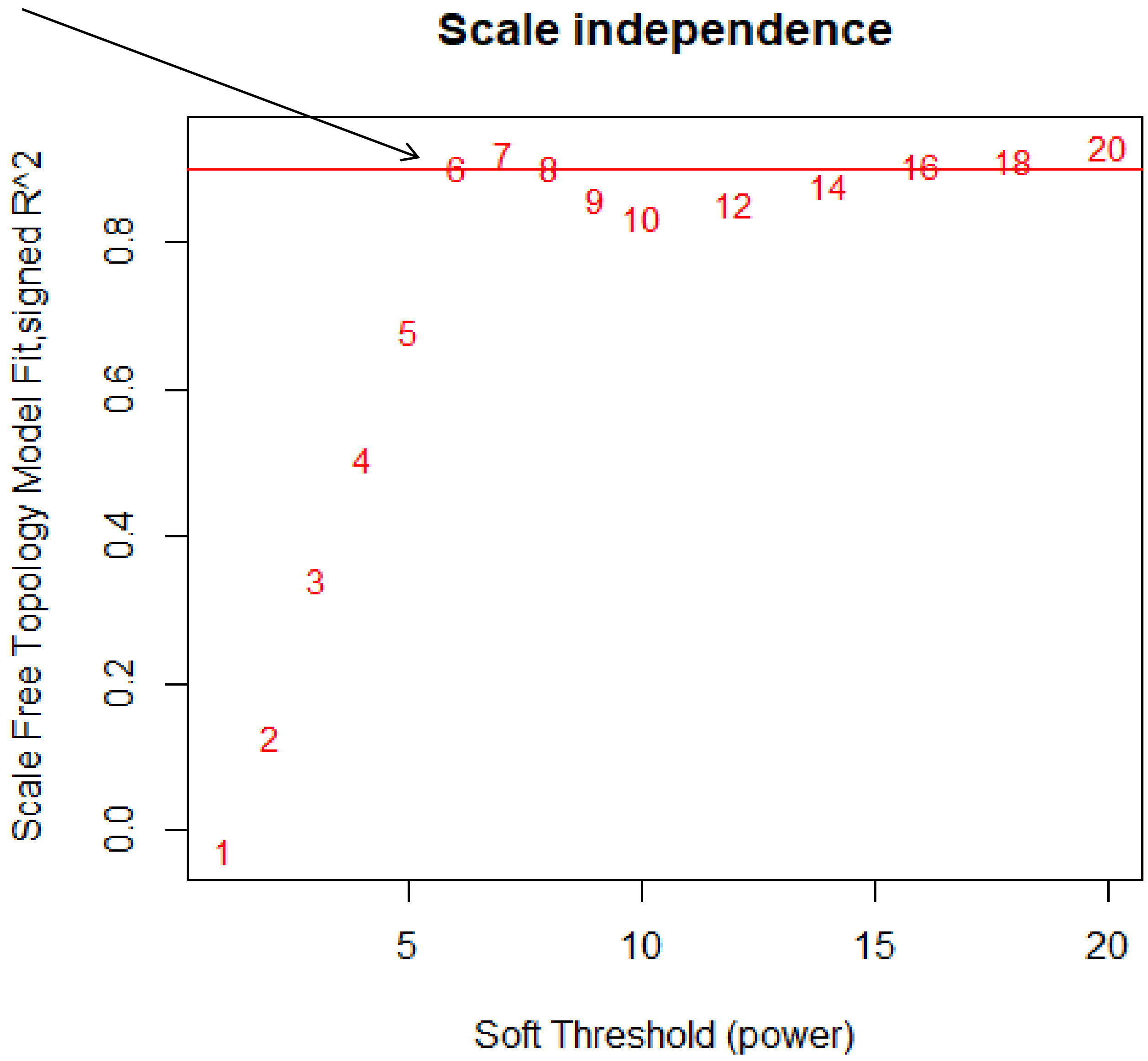
- ***Scale-free topology fit index as a function of the soft-thresholding power***

```
plot(sft$fitIndices[,1], -sign(sft$fitIndices[,3])*sft$fitIndices[,2], xlab="Soft Threshold
(power)",ylab="Scale Free Topology Model Fit,signed R^2",type="n", main = paste("Scale
independence")); text(sft$fitIndices[,1], -sign(sft$fitIndices[,3])*sft$fitIndices[,2],
labels=powers,cex=cex1,col="red");
```

- ***this line corresponds to using an R^2 cut-off of h***

```
abline(h=0.90,col="red")
```

We will choose Power 6, reached threshold r^2 > 0.9



Network Construction and Module Detection

❑ **Use function to identify modules**

```
net = blockwiseModules(datExpr, power = 6, TOMType = "unsigned",
minModuleSize = 30, reassignThreshold = 0, mergeCutHeight = 0.25, numericLabels
= TRUE, pamRespectsDendro = FALSE, saveTOMs = TRUE, saveTOMFileBase =
"femaleMouseTOM", verbose = 3)
```

❑ ***Check number of modules***

```
table(net$colors)
```

```
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18
99 609 460 409 316 312 221 211 157 123 106 100 94 91 77 76 58 47 34
```

As a result, we have 19 identified modules in this dataset

Plotting of identified Module

❑***open a graphics window***

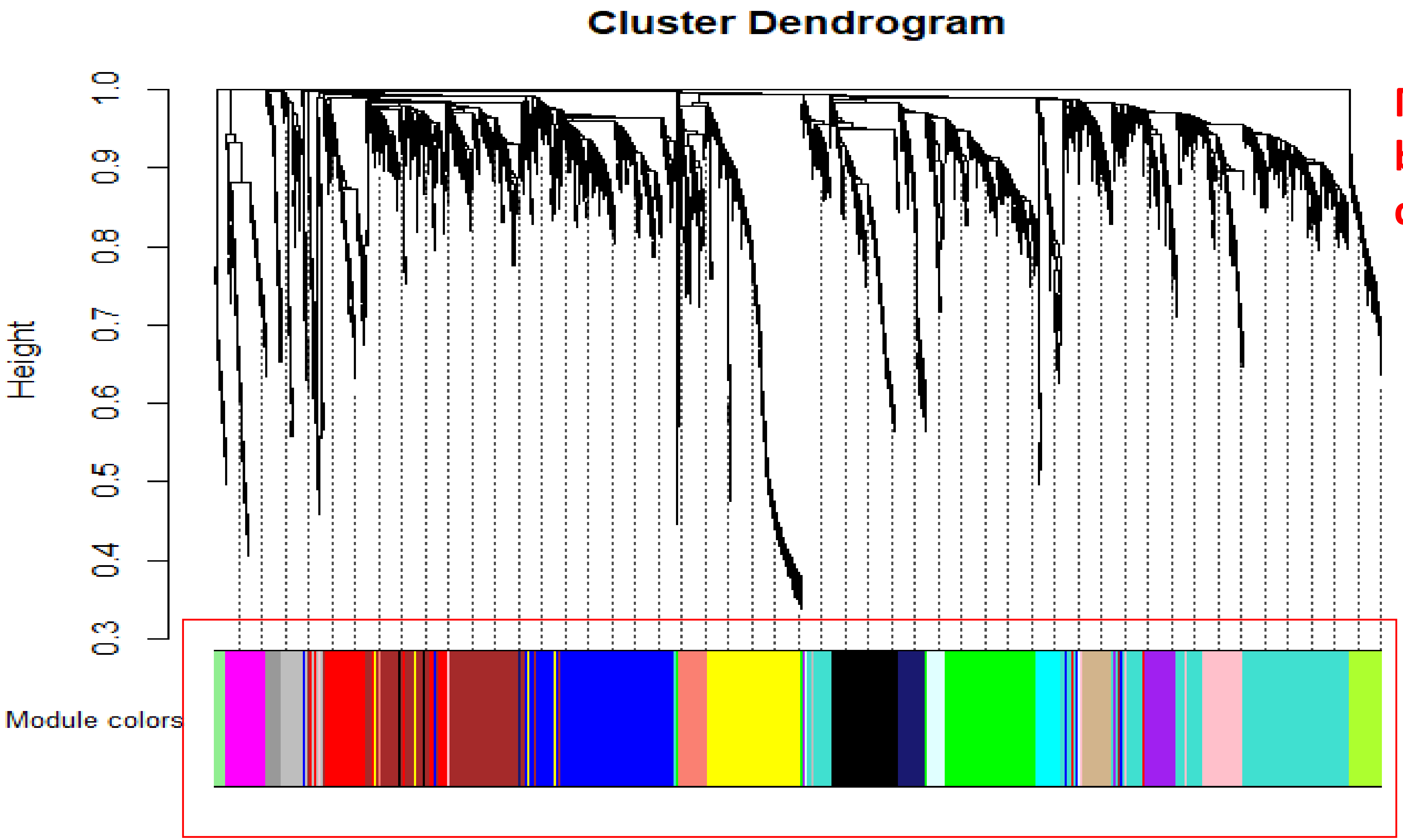
```
sizeGrWindow(12, 9)
```

❑ ***Convert labels to colors for plotting***

```
mergedColors = labels2colors(net$colors)
```

❑ ***Plot the dendrogram and the module colors underneath***

```
plotDendroAndColors(net$dendrograms[[1]], mergedColors[net$blockGenes[[1]]],
"Module colors", dendroLabels = FALSE, hang = 0.03, addGuide = TRUE, guideHang =
0.05)
```



Module= branch of a cluster tree

Each Modules indicated as different color

How does one summarize the expression profiles in a module?

□ Math answer: module eigengene
= first principal component

□ Network answer: the most highly connected intramodular hub gene

Module Eigengenes are very useful

- **1) They allow one to relate modules to each other**
  - **Allows one to determine whether modules should be merged**
- **2) They allow one to relate modules to clinical traits (HD status) and genetic variation (e.g. CAG tri-nucleotide repeat length)**

  -> avoids multiple comparison problem

- **3) They allow one to define a measure of module membership: kME=cor(x,ME)**
  - **Can be used for finding centrally located hub genes**
  - **Can be used to define gene lists for GO enrichment**

Visualizing the gene network

❑ **Get all the module labels**

```
moduleLabels = net$colors
```

❑ ***get all the module colors***

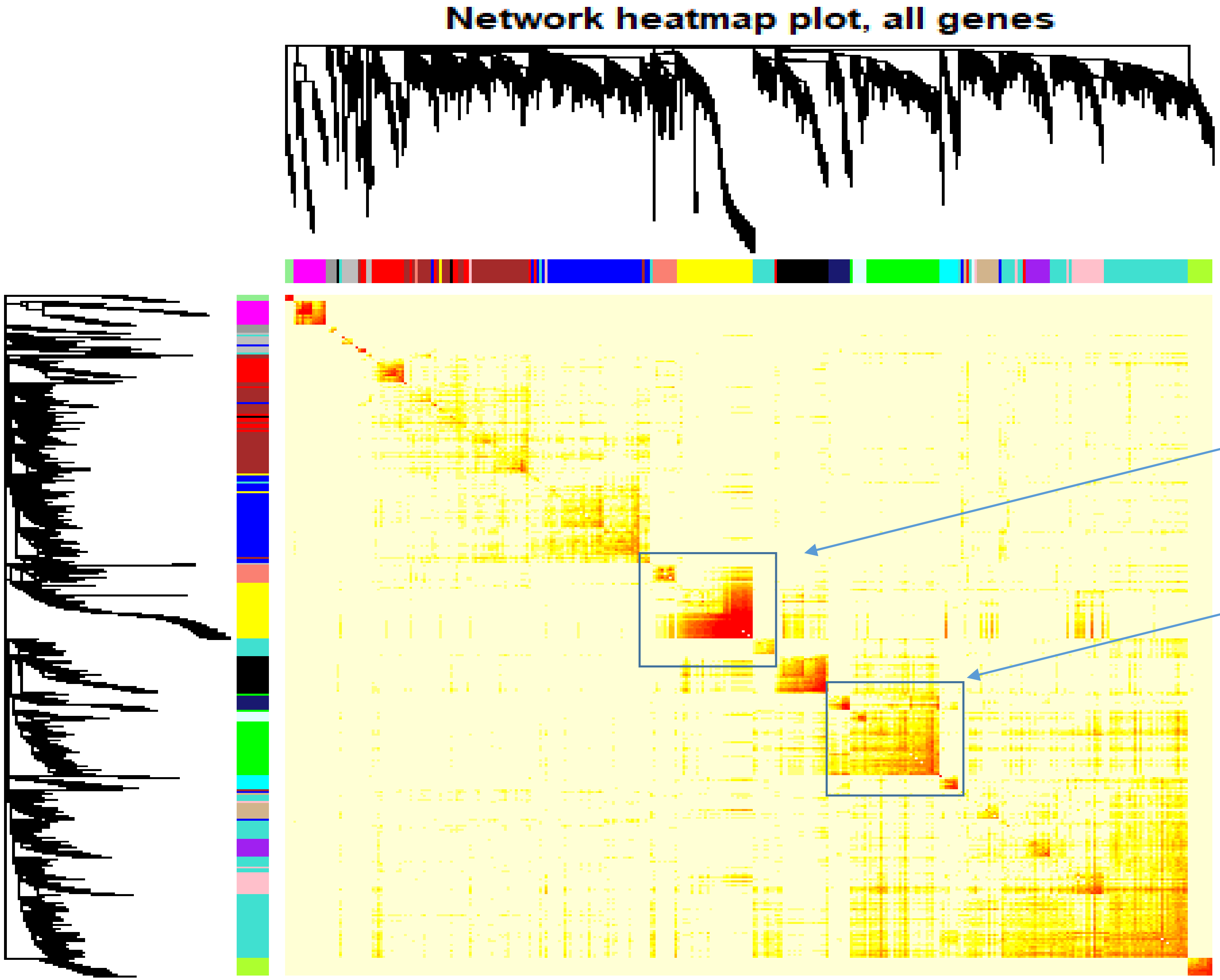
```
moduleColors = labels2colors(net$colors)
```

❑ ***get all the MEs important for next analysis***

```
MEs = net$MEs;
```

❑ ***Lets plot coexpression network in heatmap***

```
dissTOM = 1-TOMsimilarityFromExpr(datExpr, power = 6);
geneTree = net$dendrograms[[1]];
plotTOM = dissTOM^7;
diag(plotTOM) = NA;
TOMplot(plotTOM, geneTree, moduleColors, main = "Network heatmap plot, all genes")
```



Genes in these modules must have higher order interactions (higher color intensity -> stronger interactions)

Export nodes and edges detail into Cytoscape format

❑ **Recalculate topological overlap if needed**

```
TOM = TOMsimilarityFromExpr(datExpr, power = 6);
```

❑ ***Read in the gene annotation file***

```
annot = read.csv(file = "GeneAnnotation.csv");
```

❑ ***Select modules***

```
modules = c("brown", "red");
```

❑Select module probes

```
 probes = names(datExpr)
inModule = is.finite(match(moduleColors, modules));
modProbes = probes[inModule];
modGenes = annot$gene_symbol[match(modProbes, annot$substanceBXH)];
```

```
# Select the corresponding Topological Overlap

modTOM = TOM[inModule, inModule];

dimnames(modTOM) = list(modProbes, modProbes)

dim(modTOM)

modTOM = TOM[inModule, inModule];

dimnames(modTOM) = list(modProbes, modProbes)
```

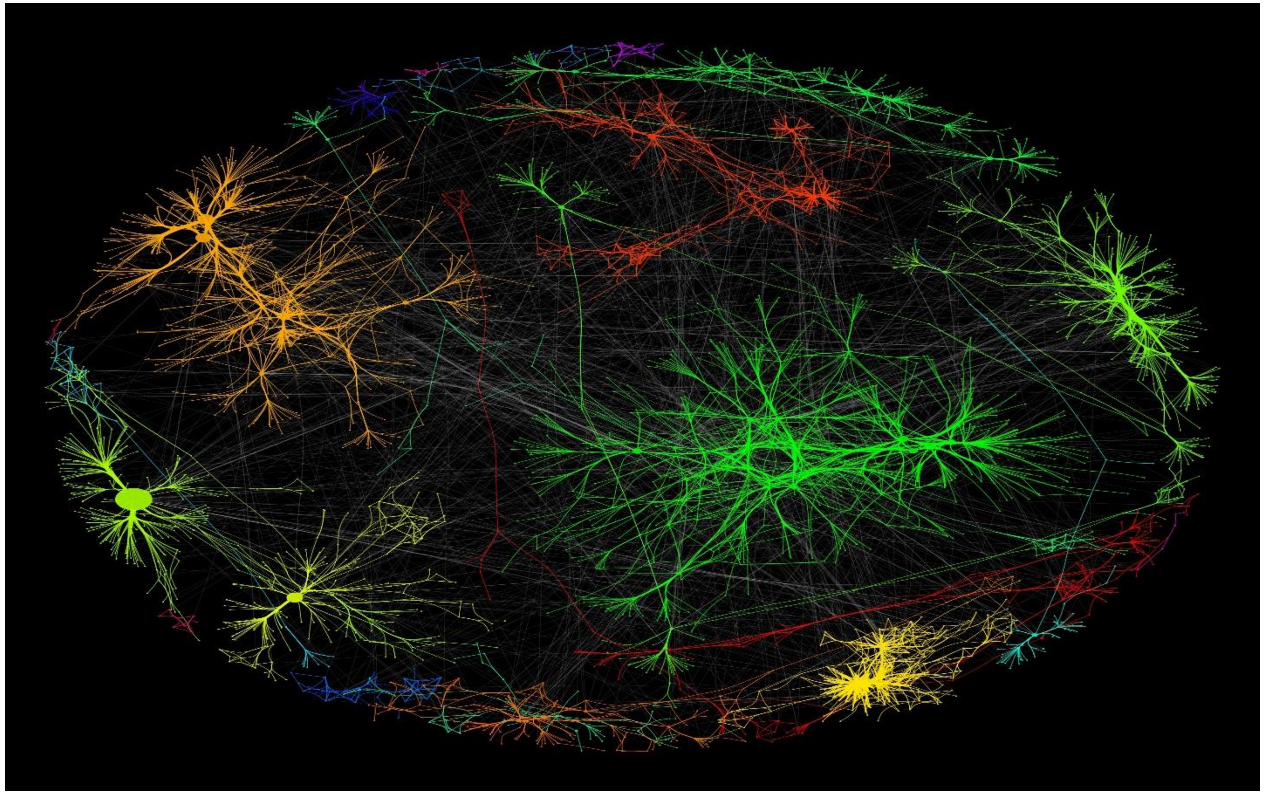
❑ Use of function exportNetworkTocytoscape

```
cyt = exportNetworkToCytoscape(modTOM, edgeFile = paste("CytoscapeInput-edges-", paste(modules, collapse="-"), ".txt", sep=""), nodeFile = paste("CytoscapeInput-nodes-", paste(modules, collapse="-"), ".txt", sep=""), weighted = TRUE, threshold = 0.02, nodeNames = modProbes, altNodeNames = modGenes, nodeAttr = moduleColors[inModule]);
```

Lets visualize interaction in Cystoscope

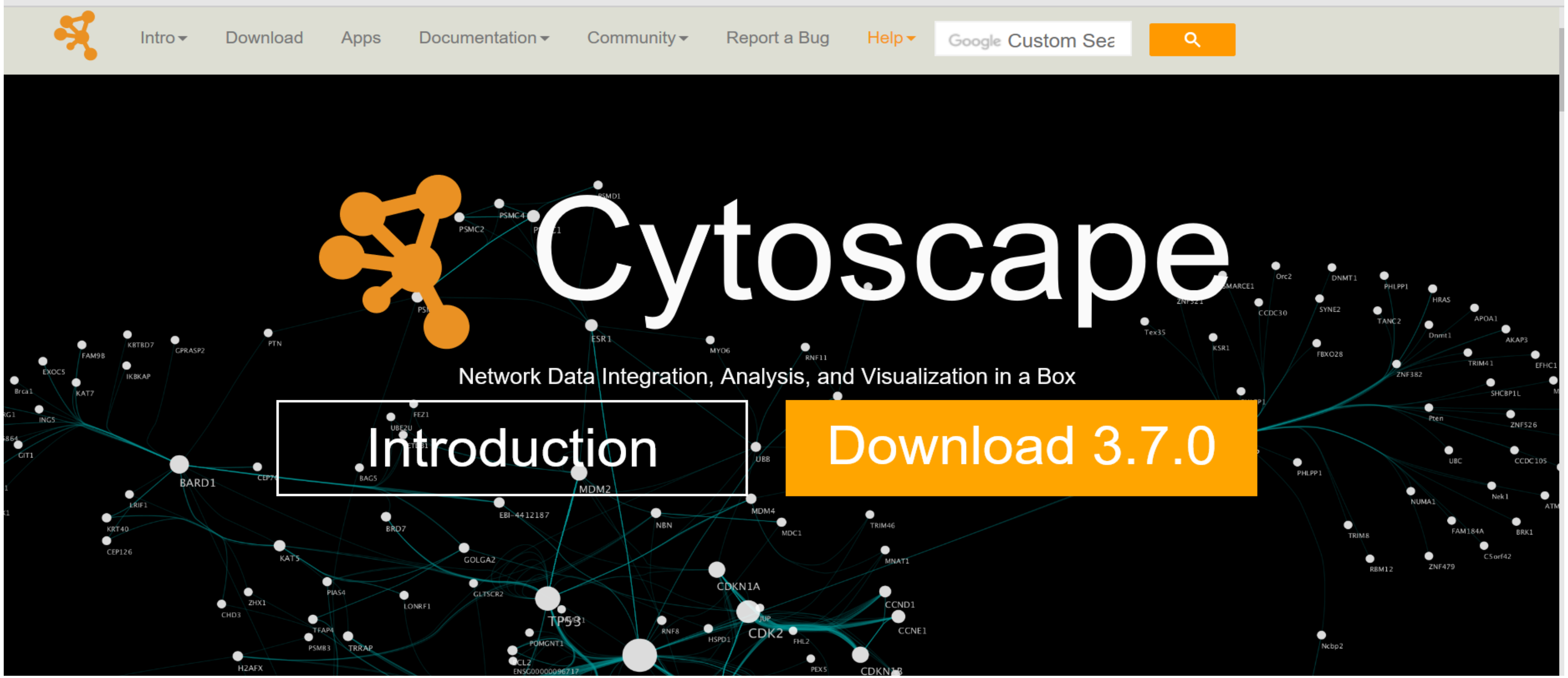
What is Cytoscape ??

Cytoscape is an open source software platform for *visualizing* molecular interaction networks and biological pathways and *integrating* these networks with annotations, gene expression profiles and other state data.

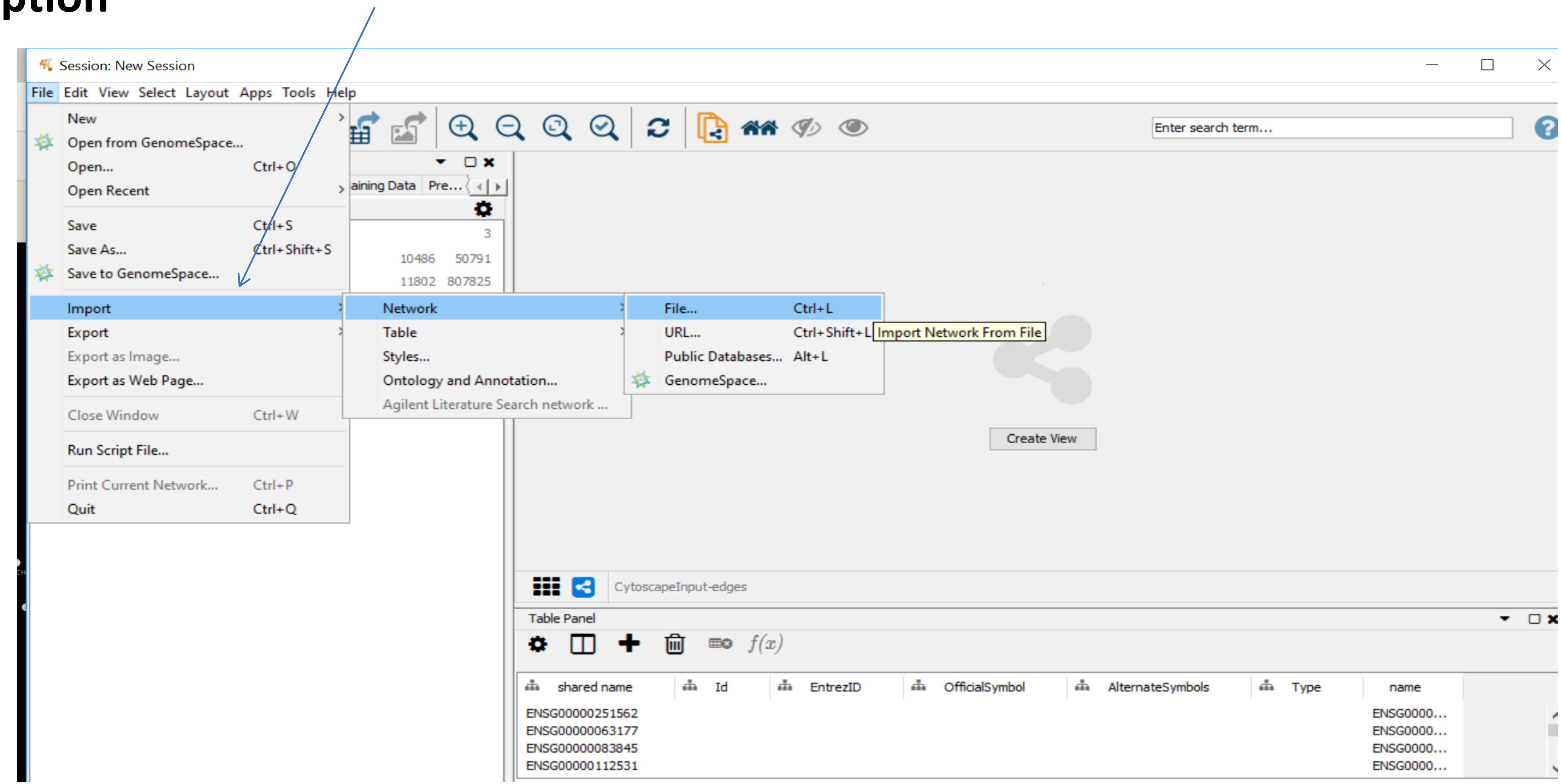


Cytoscape

Go to link : https://cytoscape.org/ and download

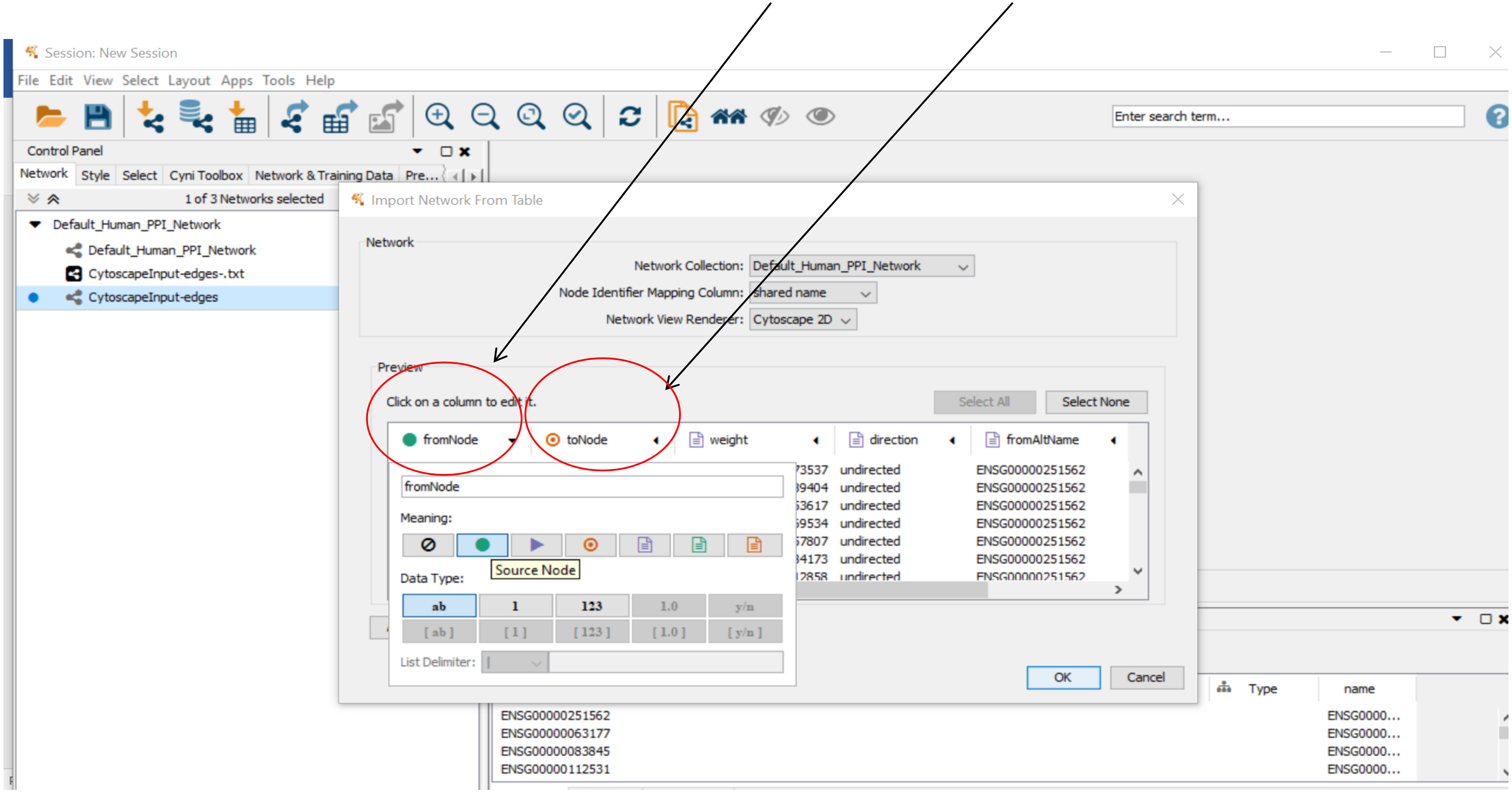


Import edge interaction file using IMPORT/Network/File from menubar file option



shared name	Id	EntrezID	OfficialSymbol	AlternateSymbols	Type	name
ENSG00000251562						ENSG0000...
ENSG00000063177						ENSG0000...
ENSG00000083845						ENSG0000...
ENSG00000112531						ENSG0000...

Change first and second column to source and Target variables



Session: New Session
File Edit View Select Layout Apps Tools Help
Enter search term...
Control Panel
Network Style Select Cyni Toolbox Network & Training Data Pre...
1 of 3 Networks selected
Default_Human_PPI_Network
Default_Human_PPI_Network
CytoscapeInput-edges-.txt
CytoscapeInput-edges
Import Network From Table
Select All
Select None
Source Node
OK
Cancel
Type
name

Network

Network Collection: Default_Human_PPI_Network

Node Identifier Mapping Column: shared name

Network View Renderer: Cytoscape 2D

Preview

Click on a column to edit it.

fromNode	toNode	weight	direction	fromAltName
		73537	undirected	ENSG00000251562
		39404	undirected	ENSG00000251562
		53617	undirected	ENSG00000251562
		59534	undirected	ENSG00000251562
		57807	undirected	ENSG00000251562
		34173	undirected	ENSG00000251562
		12858	undirected	ENSG00000251562

fromNode

Meaning: ⊘ ● ▶ ◎

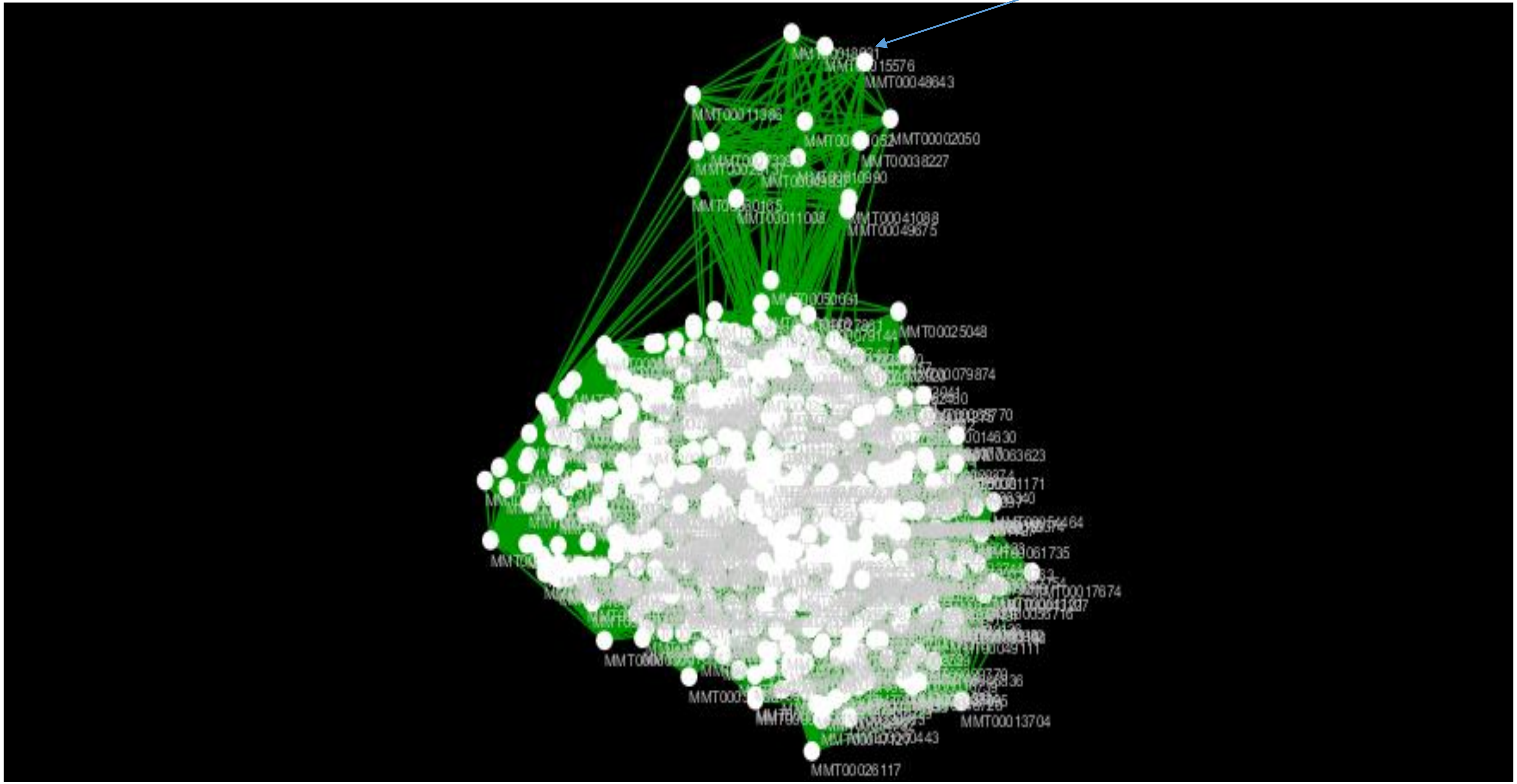
Data Type: ab 1 123 1.0 y/n [ab] [1] [123] [1.0] [y/n]

List Delimiter: |

ENSG00000251562
ENSG00000063177
ENSG00000083845
ENSG00000112531

ENSG0000...
ENSG0000...
ENSG0000...
ENSG0000...

Gene-Gene interaction view of two identified modules from WGCNA



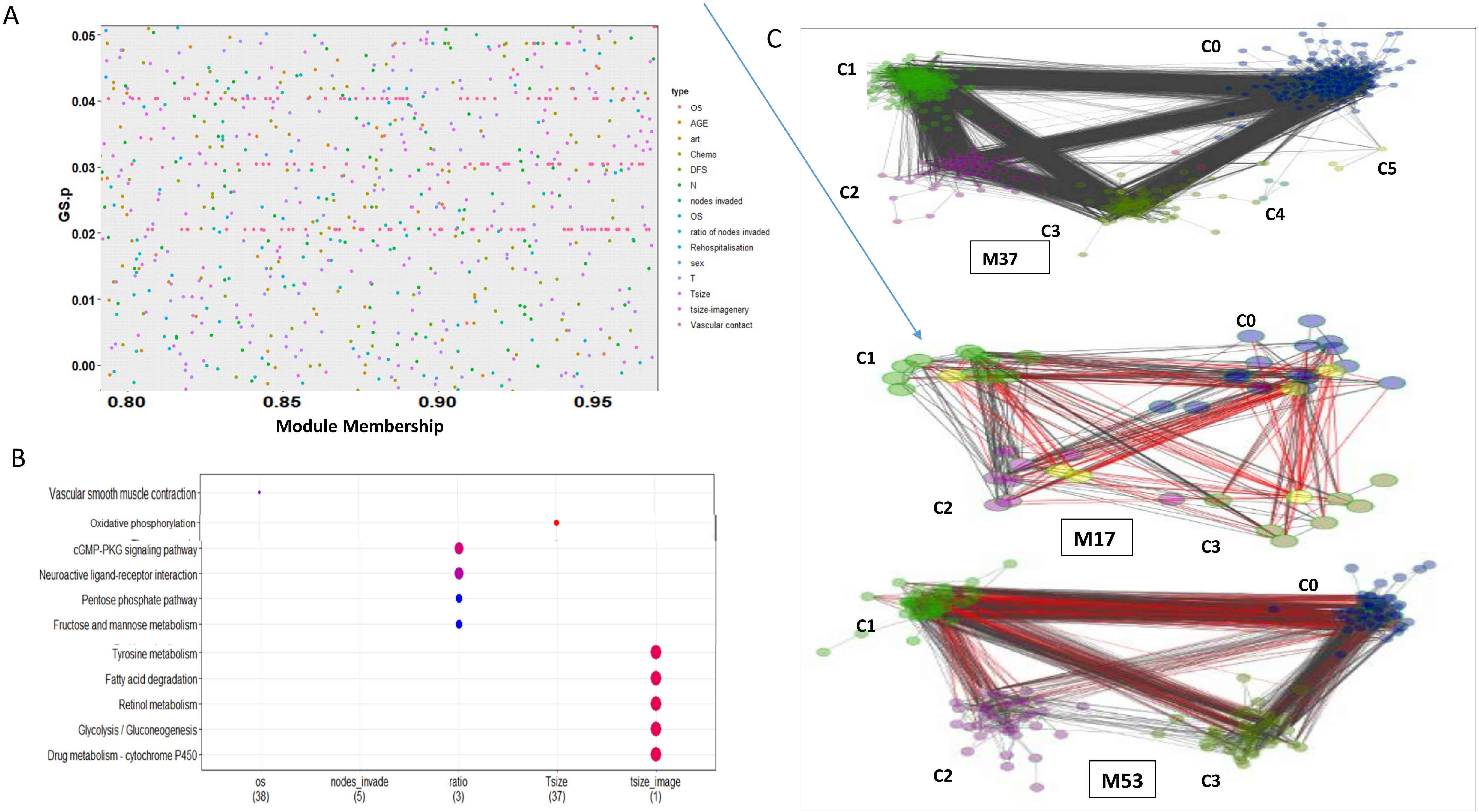
Gene as nodes

What is next after gene-gene Network ??

- ❑ **Sub clusters in networks**

- ❑ **Hub genes in networks and their functional relevance**

Survival associated Hub genes (GIGA - PDAC study)


A
GS.p
Module Membership
type
OS
AGE
art
Chemo
DFS
N
nodes invaded
OS
ratio of nodes invaded
Rehospitalisation
sex
T
Tsize
tsize-imagenery
Vascular contact
B
Vascular smooth muscle contraction
Oxidative phosphorylation
cGMP-PKG signaling pathway
Neuroactive ligand-receptor interaction
Pentose phosphate pathway
Fructose and mannose metabolism
Tyrosine metabolism
Fatty acid degradation
Retinol metabolism
Glycolysis / Gluconeogenesis
Drug metabolism - cytochrome P450
os
(38)
nodes_invade
(5)
ratio
(3)
Tsize
(37)
tsize_image
(1)
C
C0
C1
C2
C3
C4
C5
M37
C0
C1
C2
C3
M17
C0
C1
C2
C3
M53