Multiscale Embedded Gene Co-expression Network Analysis (MEGENA)

Brayan Gutierrez

MEGENA Flow Chart

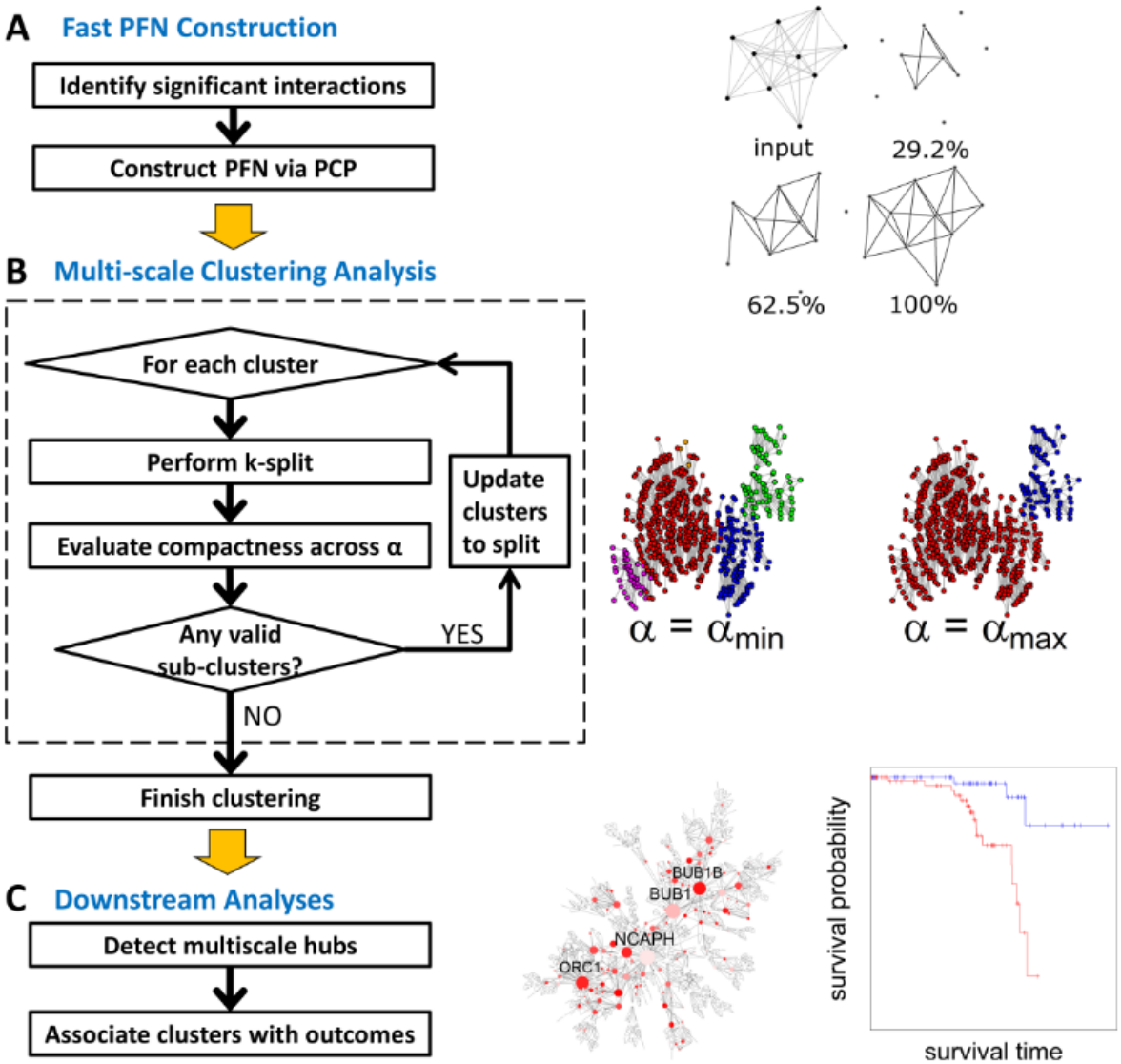


Fig 1. Flow chart of MEGENA. A) Fast planar filtered network construction. Significant interactions are first identified and then embedded on topological surface via a parallelized screening procedure described in the text. On the right, a toy example is illustrated to show construction of PFN from a thresholded network by FDR (top left), and gradual construction of PFN with number of included links and screened pairs shown on the top of each. B) Multi-scale clustering: Beginning from connected components of the initial PFN as the parent clusters, clustering is performed for each parent cluster and compactness of the sub-clusters are evaluated. These steps are described in the dotted box. The clustering is performed iteratively until there remains no further parent

Multiscale Embedded Gene Co-expression Network Analysis

Won-Min Song, Bin Zhang*

Department of Genetics and Genomic Sciences, Icahn Institute of Genomics and Multiscale Biology, Ic
School of Medicine at Mount Sinai, New York, New York, United States of America

MEGENA vs. Other Methods

- Existing methods:
  - Require predefined parameters
  - Inability to reproduce scale-free properties
  - Lack support for multiscale organization
- MEGENA:
  - Developed to address issues above
  - Parallelized network construction
  - Efficiently handles large-scale genomic data
  - Novel clustering technique that can uncover hierarchy of co-expressed gene clusters

The MEGENA Framework

- Four-step methodology for inferring and analyzing gene co-expression networks
- Overcome computational and conceptual shortcomings of previous methods (Planar Maximally Filtered Graph (PMFG), Weighted Gene Co-expression Network Analysis (WGCNA), etc.)

Fast Planar Network Construction (FPFNC)

- First major step
- Takes in raw gene expression data $\rightarrow$ Compute pairwise similarities between genes
- Variety of similarity measures including:
  - Pearson's Correlation Coefficient (PCC)
  - Mutual Information (MI)
  - Euclidean Distance
- Includes optional step to remove insignificant interactions (False Discovery Rate)

FPFNC Cont.

- The core of FPFNC is the ability to construct a Planar Filtered Network (PFN)
  - Embeds gene pairs onto a topological sphere, ensuring no links cross each other
- To make the process scalable, parallelized screening procedure (PCP) is introduced
  - Efficiently identifies subset of gene pairs that are highly likely to be embedded
  - Reduces computational time required compared to existing serial algorithms

Multiscale Clustering Analysis (MCA)

- Can identify gene co-expression modules (aka "communities" or "clusters")
- **Modules:** groups of genes whose expression profiles are tightly intertwined, suggesting they are involved in similar biological processes or disease states
- Clustering performed by the novel MCA procedure developed within MEGENA framework

MCA Cont. 1

- Uses hierarchical divisive approach to dissect complex interactions within PFN
- Turns this into coherent, multi-scale clusters
- Incorporates three distinct criteria to identify clusters:
  - **Shortest Path Distances (SPD):** Optimizes for within-cluster compactness
  - **Local Path Index (LPI):** Optimizes for local clustering structure, which is particularly effective in PFNs due to their abundance of 3- and 4-cliques
  - **Overall Modularity (Q):** A measure used to identify an optimal partition of the network

MCA Cont. 2

- Uses a compactness measure, $v(\alpha)$, a function of resolution parameter α
- Used to identify clusters at different scales
- Smaller α value, the more compact the cluster
- Multiscale capability allows the framework to identify both coarse-grained and compact clusters that can coexist within a single network
- Addresses a key limitation of many other clustering methods

Hub Genes and Significance

- MEGENA provides a formal method for identifying most influential genes within modules: the hub genes
- **Hub Genes:** "highly connected members" of the co-expression network
- Identification of these hubs is performed by a dedicated procedure called Multiscale Hub Analysis (MHA)
- MHA works by:

Grouping the different scales that show similar connectivity patterns $\rightarrow$ Identify significant hubs within each scale $\rightarrow$ Identify multiscale hubs by combining significance scores for each node across all different scales

Innovations

Method	Key Innovations	Purpose
FPFNC	Parallelized screening procedure (PCP); quality control of co-expression similarities	Construct a robust, scalable network efficiently
MCA	Hierarchical divisive clustering; uses SPD, LPI, and Q; compactness measure with a resolution parameter (α)	Identify multiscale clusters with varying degrees of compactness
MHA	Grouping of scales; combining significance scores across scales	Identify robust, highly connected hub genes at single and multiple scales

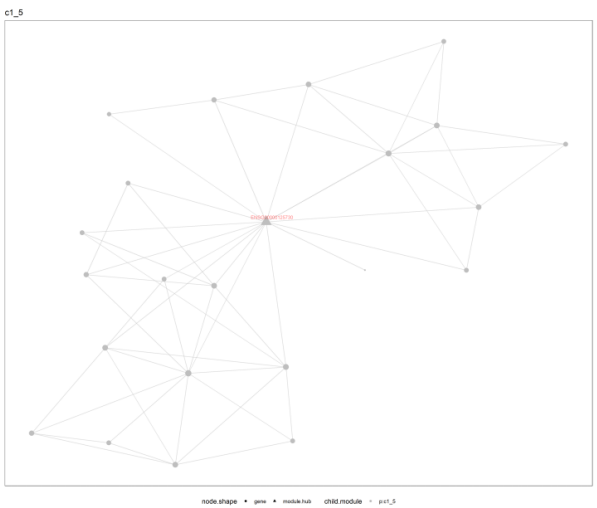
Our Data: aak100_cpmdat.csv

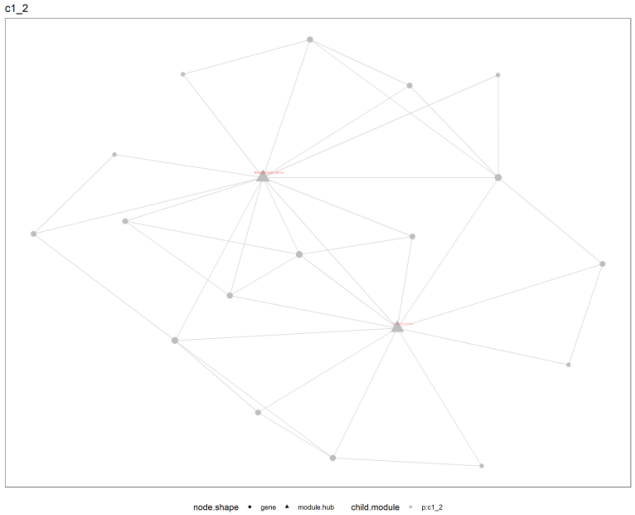
```
library(MEGENA)
library(igraph)

# 1. Load data
genes = read.csv("C:/Users/Brayan Gutierrez/Desktop/RNAseq-AMD/Dataset/aak100_cpmdat.csv")

# Remove non-expression column(s)
expr = genes[, !(colnames(genes) %in% c("mgs_level"))]

# Convert to numeric
expr_num = apply(expr, 2, as.numeric)
expr_num = matrix(expr_num, nrow = nrow(expr), ncol = ncol(expr))
rownames(expr_num) = expr$X
colnames(expr_num) = colnames(expr)

expr_num = expr_num[,-1]

# Transpose
expr_num = t(expr_num)

# 2. Calculate correlation
corr = calculate.correlation(expr_num)

# 3. Construct Planar Filtered Network (PFN)
pfn = calculate.PFN(corr)

pfn_g = graph.data.frame(pfn, directed = F)
```

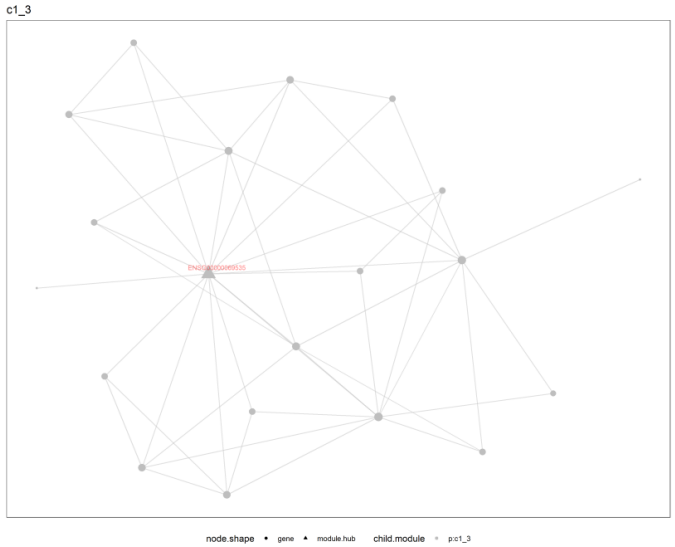
Our Data: aak100_cpmdat.csv

```
library(RColorBrewer)

# 5. Run MEGENA on igraph
meg = do.MEGENA(
    g = pfn_g,
    mod.pval = 0.05,
    hub.pval = 0.05,
    min.size = 10
)

n = 20

plot_module(MEGENA.ModuleSummary(meg), pfn_g, col.names = brewer.pal(n, 'RdYlGn'))
```

Initial Results: aak100_cpmdat.csv







Our Data: aak500_cpmdat.csv

```
library(MEGENA)
library(igraph)

# 1. Load data
genes = read.csv("C:/Users/Brayan Gutierrez/Desktop/RNAseq-AMD/Dataset/aak500_cpmdat.csv")

# Remove non-expression column(s)
expr = genes[, !(colnames(genes) %in% c("mgs_level"))]

# Convert to numeric
expr_num = apply(expr, 2, as.numeric)
expr_num = matrix(expr_num, nrow = nrow(expr), ncol = ncol(expr))
rownames(expr_num) = expr$X
colnames(expr_num) = colnames(expr)

expr_num = expr_num[,-1]

# Transpose
expr_num = t(expr_num)

# 2. Calculate correlation
corr = calculate.correlation(expr_num)

# 3. Construct Planar Filtered Network (PFN)
pfn = calculate.PFN(corr)

pfn_g = graph.data.frame(pfn, directed = F)
```

Our Data: aak500_cpmdat.csv

```
library(RColorBrewer)

# 5. Run MEGENA on igraph
meg = do.MEGENA(
    g = pfn_g,
    mod.pval = 0.05,
    hub.pval = 0.05,
    min.size = 10
)

n = 20

plot_module(MEGENA.ModuleSummary(meg), pfn_g, col.names = brewer.pal(n, 'RdYlGn'))
```

Initial Results: aak500_cpmdat.csv

SEEN IN THE `modulePlot` FILE!!

References

Song, W.-M., & Zhang, B. (2015). *Multiscale Embedded Gene Co-expression Network Analysis.* PLoS Computational Biology, 11(11), e1004574.