

# Flaviane Lopes Ferreira



Mestrado em Botânica, Instituto de Biociências, USP.

[exec](#)

## Trabalho Final

### Proposta A

Os processos biológicos são frequentemente representados sob a forma de redes, tais como a rede de interação proteína-proteína e vias metabólicas. O estudo das redes biológicas, sua modelagem, análise e visualização são tarefas importantes nas ciências da vida hoje. Um entendimento destas redes é essencial para saber o sentido biológico de grande parte dos dados complexos que está sendo gerado. Esta crescente importância das redes biológicas é também evidenciado pelo rápido aumento de publicações sobre temas relacionados com a rede e o número crescente de grupos de pesquisa relacionadas com esta área. A visualização da rede é um método fundamental que ajuda cientistas na compreensão das redes biológicas e em descobrir propriedades importantes dos processos bioquímicos subjacentes (Bachmaier et al., 2013). A rede é composta por nós e links. Em redes biológicas, os nós representam genes, proteínas ou metabólitos, enquanto os links representam as interações entre eles. As redes de co-expressão gênica visa entender a associação ou correlação entre os genes através de dados advindos por exemplo do RNA-Seq ou microarranjo.

Assim, a primeira proposta visa criar uma função que gere redes de co-expressão de genes a partir do conjunto de dados do RNA-Seq. A função usará um dataframe contendo os genes e seus respectivos valores de expressão. A partir destes dados será gerado uma matriz de correlação, podendo ter como argumento a correlação de pearson ou spearman. Depois é criado uma matriz de adjacência (0 e 1). Após este procedimento, será possível estabelecer um p-value para retirar somente as correlações que são significantes dentro da rede. A função retornará uma janela gráfica com a rede de co-expressão de genes.

### Referência

Bachmaier C, Brandes U, Schreiber F. Biological Networks. 2013. In: Tamassia R, ed. Handbook of Graph Drawing and Visualization. CRC Press, pp.621-651.

Junker Björn H. Networks in Biology. 2008. In: Junker Björn H, Schreiber Falk, eds. Analysis of Biological Networks. John Wiley & Sons, New Jersey, pp.3-14.

### Proposta B

A segunda proposta visa utilizar um conjunto de dados de atividade enzimática de fungos filamentosos para uma análise exploratória dos dados. A função terá como entrada um dataframe, e no final a função retornará um conjunto de gráficos como boxplot, barplot, histograma e quantil-quantil para avaliar os dados.

## COMENTÁRIOS PROPOSTA MELINA

em 28/abr/16

A proposta A é simples, razoável e plausível. O monitor Diogo sugeriu você dar uma olhada no pacote `igraph` para fazer a figura da rede. Seria bom a função retornar objetos com os resultados das correlações, caso o usuário queira guardar essa informação. Também sugiro dar uma olhada na função `pairs` e diversas variantes dela existentes no R, talvez seja útil mas não necessariamente.

A proposta B é ruim, sem sentido de ser colocada em uma função. Sugiro fazer a proposta A mesmo.

### Código da Função

```
==== Função co.expr.network ====

co.expr.network<-function(x,method="pearson", p.value=0.05,
node.color="orange1",link.color=c("black","gray 50"),
link.line=c("solid","dotted"), plot=T) ## Argumentos da função
{
  ## Transpõe os dados para que cada coluna seja um gene
  expr<- t(x[,-1])
  ## Coloca os nomes nas colunas e linhas da tabela de expressão
  colnames(expr)<-x[,1]
  ## Cria uma matrix de correlação par-a-par de cada gene para o método de
  correlação escolhido - "pearson" ou "spearman"
  corr<-cor(expr,method= method)
  ## Coloca 0 nas diagonais
  diag(corr)<-0
  ## Cria um arquivo com as correlações entre os genes
  write.csv(corr,file="correlação.csv")
  # Indica condição para plotar os scatterplots. Aqui o usuário tem a opção
  de fazer ou não os scatterplots, T = indica para fazê-los e F = indica para
  não fazê-los.
  if(plot==T){
    ## Faz um loop para poder criar os scatterplots da matriz de correlação
    entre os genes. Aqui o usuário precisa mudar para o número de genes que ele
    tem.
    for(i in c(0,10,20)){
      ## Abre uma janela gráfica
      X11()
      ##Faz os scatterplots da correlação entre os genes através da matriz de
      correlação
      pairs(corr[(i+1):(i+10),(i+1):(i+10)],lower.panel=panel.smooth,
upper.panel=panel.smooth, pch=1,cex.labels =1.0, cex.axis=1.0, font.labels
=1.0)
    }
  }
  ## Constrói uma matriz vazia para p-valores com os dados de expressão dos
```

```
genes
pcorr<-matrix(NA,ncol(expr),ncol(expr))
## Faz um loop para percorrer a matriz
for(i in 1:ncol(expr)){
  ## Faz um loop para percorrer a matriz
  for(j in 1:ncol(expr))
    ## Calcula do p-valor para correlação entre dois genes
    pcorr[i,j]<-cor.test(expr[,i],expr[,j])$p.value
}
## Coloca o nome dos genes nas colunas da tabela de p-valor
colnames(pcorr)<-x[,1]
## Coloca o nome dos genes nas linhas da tabela de p-valor
rownames(pcorr)<-x[,1]
## Cria um arquivo do p-valor entre os genes
write.csv(pcorr,file="p-valor.csv")
## Verifica o p-valor abaixo do estabelecido para cada gene, e assim até
alcançar o término da dimensão da matrix
for(i in 1:dim(corr)[1]){
  ## Verifica o p-valor abaixo do estabelecido para cada gene,e assim até
alcançar o término da dimensão da matrix
  for(j in 1:dim(corr)[1]){
    ## Se na tabela de pcorr,os p-valores de um par de genes for maior ao
p-valor estabelecido ou quando os valores são missing values, coloca zero na
tabela de correlação. Removendo assim o link que existiria entre dois genes.
    if(pcorr[i,j]>=p.value | is.na(pcorr[i,j])) corr[i,j]=0
  }
}

## Carrega o igraph
library(igraph)
## Constrói a rede a partir de uma matriz de adjacência
g<- graph.adjacency(corr,mode="undirected",weighted=T)

##Cria uma lista de links entre os genes a partir da matriz de adjacência
(g)
a<-as_edgelist(g)
## Cria um objeto c com NA dentro
c<-NA
## Faz um loop da lista entre os genes da matriz de adjacência (g)
for(i in 1:nrow(a)){
  ##Verifica o p-valor para estes gene e coloca no objeto c
  c[i]<-pcorr[which(colnames(corr)==a[i,1]),which(colnames(corr)==a[i,2])]
}

## Adiciona o p-valor para o objeto c
E(g)$pvalue<-c

## Adiciona o nome dos genes
colnames(pcorr)

##Deixa a correlação positiva e negativa na lista de genes da matriz de
```

```
adjacência (g)
E(g)$peso<-E(g)$weight
# Deixa a correlação em valores absolutos
E(g)$weight<-abs(E(g)$weight)

## Cria uma lista de links entre os genes
links<-get.edgelist(g, names=TRUE)
## Adiciona a correlação com sinal
cor<- E(g)$peso
## Adiciona o p-valor < 0.05
pvalue<-E(g)$pvalue
## Liga os diferentes objetos e coloca em uma lista
lista<-cbind(links,cor,pvalue)
## Cria um arquivo da rede dos genes com a correlação e o p-valor < 0.05
write.csv(lista,file="lista_rede.csv")

##Estabelece o layout do gráfico da rede. Este layout é o default da função.
layout <- layout.kamada.kawai(g)
## Dá as coordenadas do layout de x e y
layout<- norm_coords(layout, ymin=-1, ymax=1, xmin=-1, xmax=1)
## Estabelece a cor dos nós do gráfico da rede
V(g)$color<-node.color

## Faz um loop da matriz de correlação
for(i in 1:length(E(g)$peso)){
  ## Se a correlação for > 0, coloca a cor preta.
  if(E(g)$peso[i]>0) E(g)$color[i]<-link.color[1]
  ## Caso a correlação seja < 0, adiciona a cor cinza.
  else E(g)$color[i]<-link.color[2]
}

## Faz um loop da matriz de correlação
for(i in 1:length(E(g)$peso)){
  ## Se a correlação for > 0, coloca uma linha sólida entre a correlação
entre os genes.
  if(E(g)$peso[i]>0) E(g)$lty[i]<-link.line[1]
  ## Caso a correlação seja < 0, coloca uma linha pontilhada entre a
correlação entre os genes.
  else E(g)$lty[i]<-link.line[2]
}

## Abre uma janela gráfica
X11()
## Faz o gráfico da rede com genes que tem o p-valor < 0.05
plot.igraph(g,vertex.size=20,vertex.label.color="black",
vertex.label.cex=0.75, rescale=F, layout=layout*0.97, edge.width=2)

## Retorna a lista de genes com os valores de correlação e p-valor da rede
toda,e uma lista da rede com as correlações no qual o p-valor é < 0.05
return(list(correlacao=corr,p.valor=pcorr,lista=lista))
```

}

## Arquivo da Função

[funcao.r](#)

## Página Help

co.expr.network

package:unknown

R Documentation

Cria uma rede de co-expressão de genes a partir de uma conjunto dados do RNA-Seq.

### Description:

Esta função cria uma rede de co-expressão a partir de um dataframe que contém os valores de expressão de cada gene. Esses valores de expressão gênica é usado para criar uma matriz de correlação que pode ter como método a correlação de Pearson ou Spearman. Além disso, é possível estabelecer um p-valor para selecionar as correlações que são mais significantes na rede. A função retorna para o usuário os arquivos da correlação e p-valor de todos genes, scatterplots da matriz de correlação, uma lista de ligações entre os genes para o p-valor estabelecido, bem como uma janela gráfica da rede.

### Usage:

```
co.expr.network<-function(x=,method="pearson", p.value=0.05,  
node.color="orange1",link.color=c("black","gray 50"),  
link.line=c("solid","dotted"), plot=T)
```

### Arguments:

x                   É um dataframe.

method              Método de correlação usado para análise dos dados. A correlação de Pearson é o padrão, mas pode ser usada a correlação de Spearman.

p.value             Valor numérico para selecionar as correlações mais significantes na rede.

node.color          Define a cor dos nós na rede.

link. color         Define a cor dos links entre os nós.

link.line           Define o tipo de linha do link entre os nós.

plot                Define se os scatterplots da função pairs será feito ou não.

### Details:

É necessário o download do pacote igraph.

### Value:

Quando o dataframe é inserido na função, é retornado ao usuário a matriz e

os scatterplots da correlação, a matriz com o p-valor, uma lista de ligações entre os genes - onde o p-valor seja abaixo de 0.05, e uma janela gráfica da rede.

Author(s):

Flaviane Lopes Ferreira

flalopesferreira@hotmail.com

Example:

```
gene.expression <- read.csv("~/Desktop/Funcao/Genes_expression.csv", header = F)
result <- co.expr.network(gene.expression, method="pearson", p.value=0.05, node.color="orange1", link.color=c("black", "gray 50"), link.line=c("solid", "dotted"), plot=T)
```

## Arquivo para Teste

[genes\\_expression.csv](#)

From:

<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:

[http://ecor.ib.usp.br/doku.php?id=05\\_curso\\_antigo:r2016:alunos:trabalho\\_final:flalopesferreira:start](http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2016:alunos:trabalho_final:flalopesferreira:start) 

Last update: **2020/08/12 06:04**