Gene expression data is very important in experimental molecular biology (Brazma and Vilo 2000), especially for cancer study (Fehrmann et al. 2015). The large-scale microarray data and RNA-seq data provide good opportunity to do the gene co-expression analyses and identify co-expressed gene modules; and the effective and efficient algorithms are needed to implement such analysis. Substantial efforts have been made in this field, such as Cheng and Church (2000), Plaid (Lazzeroni, Owen, and others 2002), Bayesian Biclustering (BCC, Gu and Liu 2008), among them Cheng and Church and Plaid has the R package implementation. It is worth noting that our in-house biclustering algorithm, QUBIC (Li et al. 2009), is reviewed as one of the best programs in terms of their prediction performance on benchmark datasets. Most importantly, it is reviewed as the best one for large-scale real biological data (Eren et al. 2012).

Until now, QUBIC has been cited over 120 times (via Google Scholar) and its web server, QServer, was developed in 2012 to facilitate the users without comprehensive computational background (Zhou et al. 2012). In the past five years, the cost of RNA-sequencing decreased dramatically, and the amount of gene expression data keeps increasing. Upon requests from users and collaborators, we developed this R package of QUBIC to void submitting large data to a webserver.

The unique features of our R package (Y. Zhang et al. 2017) include (1) updated and more stable back-end resource code (re-written by C++), which has better memory control and is more efficient than the one published in 2009. For an input dataset in Arabidopsis, with 25,698 genes and 208 samples, we observed more than 40% time saving; and (2) comprehensive functions and examples, including discretize function, heatmap drawing and network analysis.

**How to cite**

citation("QUBIC")

Please cite the QUBIC package in your work, whenever you use it:

Yu Zhang, Juan Xie, Jinyu Yang, Anne Fennell, Chi Zhang, Qin Ma; QUBIC: a bioconductor package for qualitative biclustering analysis of gene co-expression data. Bioinformatics, 2017; 33 (3): 450-452. doi: 10.1093/bioinformatics/btw635

A BibTeX entry for LaTeX users is

```bibtex
@Article{,
  title = {{QUBIC}: a bioconductor package for qualitative biclustering analysis of gene co-expression data},
  author = {Yu Zhang and Juan Xie and Jinyu Yang and Anne Fennell and Chi Zhang and Qin Ma},
  journal = {Bioinformatics},
  year = {2017},
  volume = {33},
  number = {3},
  pages = {450--452},
  doi = {10.1093/bioinformatics/btw635},
}  ```
Other languages

If R is not your thing, there is also a C version of QUBIC.

Help

If you are having trouble with this R package, contact the maintainer, Yu Zhang.

Install and load

Stable version from BioConductor

```r
source("https://bioconductor.org/biocLite.R")
biocLite("QUBIC")
```

Or development version from GitHub

```r
install.packages("devtools")
develtools::install_github("zy26/QUBIC")
```

Load QUBIC

```r
library("QUBIC")
```

Functions

There are nine functions provided by QUBIC package.

- `qudiscretize()` creates a discrete matrix for a given gene expression matrix;
- `BCQU()` performs a qualitative biclustering for real matrix;
- `BCQUD()` performs a qualitative biclustering for discretized matrix;
- `quheatmap()` can draw heatmap for single bicluster or overlapped biclusters;
- `qunetwork()` can automatically create co-expression networks based on the identified biclusters by QUBIC;
- `qunet2xml()` can convert the constructed co-expression networks into XGMML format for further network analysis in Cytoscape, Biomax and JNets;
- *query-based biclustering* allows users to input additional biological information to guide the biclustering progress;
- `bicluster` expanding expands existing biclusters under specified consistency level;
- `biclusters comparison` compares biclusters obtained via different algorithms or parameters.

The following examples illustrate how these functions work.

Example of a random matrix with two different embedded biclusters

```r
library(QUBIC)
set.seed(1)
# Create a random matrix
test <- matrix(rnorm(10000), 100, 100)
```
# Discretization

```r
matrix1 <- test[1:7, 1:4]
```

```r
##
##   cond_1 cond_2 cond_3 cond_4
## gene_1  0.6264538 -0.62036668  0.4094018  0.89367370
## gene_2  0.1836433  0.04211587  1.6888733 -1.04729815
## gene_3 -0.8366286 -0.91092165  1.5865884  1.97133739
## gene_4  1.5952808  0.15802877  2.4976616  1.51221269
## gene_5  0.3295078 -0.65458464 -2.2852355  1.65414530
## gene_6 -0.8204684  1.76728727  2.4976616  0.66706620
## gene_7  0.4874291  0.71670748  0.6670662  0.08296573
```

```r
matrix2 <- qudiscretize(matrix1)
```

```r
##
##   cond_1 cond_2 cond_3 cond_4
## gene_1   1     0     0     1
## gene_2     0     0     1    -1
## gene_3     0    -1     0     1
## gene_4     1     0     0    -1
## gene_5     0     0    -1     1
## gene_6    -1     0     1     0
## gene_7     0     1     0    -1
```

# Fill bicluster blocks

```r
t1 <- runif(10, 0.8, 1)
t2 <- runif(10, 0.8, 1) * (-1)
t3 <- runif(10, 0.8, 1) * sample(c(-1, 1), 10, replace = TRUE)
test[11:20, 11:20] <- t(rep(t1, 10) * rnorm(100, 3, 0.3))
test[31:40, 31:40] <- t(rep(t2, 10) * rnorm(100, 3, 0.3))
test[51:60, 51:60] <- t(rep(t3, 10) * rnorm(100, 3, 0.3))
```

# QUBIC

```r
res <- biclust::biclust(test, method = BCQU())
```

```r
summary(res)
```

```r
## An object of class Biclust
##
## Call:
biclust::biclust(x = test, method = BCQU())
##
## Number of Clusters found: 39
##
## Cluster sizes:
##
## Number of Rows: 10 9 9 9 10 5 3 2 3 3 3 2 2 2
## Number of Columns: 9 9 8 7 5 3 5 6 4 4 6 6 6
## Number of Rows: 15 16 17 18 19 20 21 22 23 24 25 26 27
## Number of Columns: 6 6 6 6 6 5 5 5 5 5 5 5 5
## Number of Rows: 2 2 2 2 2 2 2 2 2 2 2 2 2
## Number of Columns: 6 6 6 6 6 5 5 5 5 5 5 5 5
## Number of Rows: 2 2 2 2 2 2 2 2 2 2 2 2 2
```
Figure 1: Heatmap for two overlapped biclusters in the simulated matrix

### Number of Columns:  5 5 5 5 5 5 5 5 5 5 5 5

# Show heatmap
hmcols <- colorRampPalette(rev(c("#D73027", "#FC8D59", "#FEE090", "#FFFFBF",
"#E0F3F8", "#91BFDB", "#4575B4")))(100)

# Specify colors
par(mar = c(4, 5, 3, 5) + 0.1)
quheatmap(test, res, number = c(1, 3), col = hmcols, showlabel = TRUE)

## [1] "yto 0"
## [1] "xlo 0"

**Example of Saccharomyces cerevisiae**

library(QUBIC)
data(BicatYeast)

# Discretization
matrix1 <- BicatYeast[1:7, 1:4]
matrix1

## cold_green_6h cold_green_24h cold_roots_6h cold_roots_24h
matrix2 <- qudiscretize(matrix1)

## QUBIC
x <- BicatYeast
system.time(res <- biclust::biclust(x, method = BCQU()))

summary(res)

# An object of class Biclust
#
# call:
# biclust::biclust(x = x, method = BCQU())
#
# Number of Clusters found: 77
#
# Cluster sizes:
# Number of Rows: 72 53 37 80 56 98 47 26 45 35 29 34 42 33
# Number of Columns: 4 5 7 3 4 2 4 7 4 5 6 5 4 5
# Number of Rows: 41 41 32 53 39 38 74 29 47 23 27 22 33
# Number of Columns: 4 4 5 3 4 4 2 5 3 6 5 6 4
# Number of Rows: 41 40 19 51 20 16 32 31 23 30 44 44 41
# Number of Columns: 3 3 6 2 5 6 3 3 4 3 2 2 2
# Number of Rows: 13 39 26 39 19 25 25 25 25 8 36 9 8
# Number of Columns: 6 2 3 2 4 3 3 3 3 9 2 8 9
# Number of Rows: 54 55 56 57 58 59 60 61 62 63 64 65 66
# Number of Columns: 6 4 3 4 2 2 3 2 4 5 3 3 3
# Number of Rows: 24 24 9 14 14 20 12 9 17 8 3
# Number of Columns: 2 2 5 3 3 2 3 4 2 3 5
Figure 2: Heatmap for the second bicluster identified in the *Saccharomyces cerevisiae* data. The bicluster consists of 53 genes and 5 conditions.

We can draw heatmap for single bicluster.

```r
# Draw heatmap for the second bicluster identified in Saccharomyces cerevisiae data
library(RColorBrewer)
pal <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
par(mar = c(5, 4, 3, 5) + 0.1, mgp = c(0, 1, 0), cex.lab = 1.1, cex.axis = 0.5, cex.main = 1.1)
quheatmap(x, res, number = 2, showlabel = TRUE, col = pal)
```

We can draw heatmap for overlapped biclusters.

```r
# Draw for the second and third biclusters identified in Saccharomyces cerevisiae data
par(mar = c(5, 5, 5, 5), cex.lab = 1.1, cex.axis = 0.5, cex.main = 1.1)
pal <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
quheatmap(x, res, number = c(2, 3), showlabel = TRUE, col = pal)
```

We can draw network for single bicluster.

```r
# Construct the network for the second identified bicluster in Saccharomyces cerevisiae
net <- qunetwork(x, res, number = 2, group = 2, method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
  qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", minimum = 0.6,
```
Figure 3: Heatmap for the second and third biclusters identified in the *Saccharomyces cerevisiae* data. Bicluster #2 (topleft) consists of 53 genes and 5 conditions, and bicluster #3 (bottom right) consists of 37 genes and 7 conditions.
Figure 4: Network for the second bicluster identified in the *Saccharomyces cerevisiae* data.

```r
color = cbind(rainbow(length(net[[2]]) - 1), "gray"), edge.label = FALSE)
```

We can also draw network for overlapped biclusters.

```r
net <- qunetwork(x, res, number = c(2, 3), group = c(2, 3), method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
  qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", minimum = 0.6,
                legend.cex = 0.5, color = c("red", "blue", "gold", "gray"), edge.label = FALSE)
```

We can use Cytoscape, Biomax or JNets open file named 'tempnetworkresult.gr'

Example of *Escherichia coli* data

The *Escherichia coli* data consists of 4,297 genes and 466 conditions.
Figure 5: Network for the second and third biclusters identified in the *Saccharomyces cerevisiae* data.

```r
library(QUBIC)
library(QUBICdata)
data("ecoli", package = "QUBICdata")

# Discretisation
matrix1 <- ecoli[1:7, 1:4]

## dinI_U_N0025 dinP_U_N0025 lexA_U_N0025 lon_U_N0025
## b3241 7.122300 7.195453 7.051193 7.124200
## b3240 7.184417 7.336610 7.283377 7.188263
## b3243 7.902090 7.963167 7.847747 7.943650
## b3242 6.801900 6.843213 6.795007 6.889897
## b0885 9.057120 8.918723 8.985483 9.002663

matrix2 <- qudiscretize(matrix1)

## dinI_U_N0025 dinP_U_N0025 lexA_U_N0025 lon_U_N0025
## b4634 -1 1 0 0
## b3241 0 1 -1 0
## b3240 -1 1 0 0
## b3243 0 1 -1 0
## b3242 0 0 -1 1
## b2836 -1 0 0 1
## b0885 1 -1 0 0
```
# QUBIC

```r
res <- biclust::biclust(ecoli, method = BCQU(), r = 1, q = 0.06, c = 0.95, o = 100,
                        f = 0.25, k = max(ncol(ecoli)/%/%20, 2))

system.time(res <- biclust::biclust(ecoli, method = BCQU(), r = 1, q = 0.06, c = 0.95,
o = 100, f = 0.25, k = max(ncol(ecoli)/%/%20, 2)))
```

## user system elapsed
## 13.332 0.044  6.409

```r
summary(res)
```

## An object of class Biclust

## call:
## biclust::biclust(x = ecoli, method = BCQU(), r = 1, q = 0.06, c = 0.95, o = 100,
## f = 0.25, k = max(ncol(ecoli)/%/%20, 2))

## Number of Clusters found:  20

## Cluster sizes:
## BC 1 BC 2 BC 3 BC 4 BC 5 BC 6 BC 7 BC 8 BC 9 BC 10 BC 11 BC 12 BC 13 BC 14
## Number of Rows:  437 121  51 108 103 65 41 26 27 20 25 23 17 18
## Number of Columns:  29 45 94 44 38 38 31 33 31 27 19 20 23 21
## BC 15 BC 16 BC 17 BC 18 BC 19 BC 20
## Number of Rows:  14 15  13 11  5  6
## Number of Columns:  26 20 22 25 32 23

# Draw heatmap for the 5th bicluster identified in Escherichia coli data

```r
library(RColorBrewer)
paleta <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
par(mar = c(5, 4, 3, 5), mgp = c(0, 1, 0), cex.lab = 1.1, cex.axis = 0.5,
     cex.main = 1.1)
quheatmap(ecoli, res, number = 5, showlabel = TRUE, col = paleta)
```

```r
library(RColorBrewer)
paleta <- colorRampPalette(rev(brewer.pal(11, "RdYlBu")))(11)
par(mar = c(5, 4, 3, 5), cex.lab = 1.1, cex.axis = 0.5, cex.main = 1.1)
quheatmap(ecoli, res, number = c(4, 8), showlabel = TRUE, col = paleta)
```

## [1] "yto 0"
## [1] "xlo 1"

# construct the network for the 5th identified bicluster in Escherichia coli data

```r
net <- qunetwork(ecoli, res, number = 5, group = 5, method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
  qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", minimum = 0.6,
                color = cbind(rainbow(length(net[[2]]) - 1), "gray"), edge.label = FALSE)
```

## Warning in qgraph::qgraph(net[[1]], groups = net[[2]], layout = "spring", : The following arguments are not documented and likely not arguments of qgraph and thus ignored: edge.label

10
Figure 6: Heatmap for the fifth bicluster identified in the *Escherichia coli* data. The bicluster consists of 103 genes and 38 conditions.

```r
# construct the network for the 4th and 8th identified bicluster in Escherichia coli data
net <- qunetwork(ecoli, res, number = c(4, 8), group = c(4, 8), method = "spearman")
if (requireNamespace("qgraph", quietly = TRUE))
  qgraph::qgraph(net[[1]], groups = net[[2]], legend.cex = 0.5, layout = "spring",
                minimum = 0.6, color = c("red", "blue", "gold", "gray"), edge.label = FALSE)
## Warning in qgraph::qgraph(net[[1]], groups = net[[2]], legend.cex = 0.5, : The following
## arguments
## are not documented and likely not arguments of qgraph and thus ignored: edge.label
```

**Query-based biclustering**

We can conduct a query-based biclustering by adding the weight parameter. In this example, the instance file “511145.protein.links.v10.txt” (Szklarczyk et al. 2014) was downloaded from string (http://string-db.org/download/protein.links.v10/511145.protein.links.v10.txt.gz) and decompressed and saved in working directory.

```r
# Here is an example to download and extract the weight
library(igraph)
tmp <- tempfile()
download.file(url, tmp)
graph = read.graph(gzfile(tmp), format = "ncol")
unlink(tmp)
ecoli.weight <- get.adjacency(graph, attr = "weight")
```
Figure 7: Heatmap for the fourth and eighth biclusters identified in the *Escherichia coli* data. Bicluster #4 (topleft) consists of 108 genes and 44 conditions, and bicluster #8 (bottom right) consists of 26 genes and 33 conditions.
Figure 8: Network for the fifth bicluster identified in the *Escherichia coli* data.

Figure 9: Network for the fourth and eighth biclusters identified in the *Escherichia coli* data.
library(QUBIC)
library(QUBICdata)
data("ecoli", package = "QUBICdata")
data("ecoli.weight", package = "QUBICdata")
res0 <- biclust(ecoli, method = BCQU(), verbose = FALSE)
res0

## An object of class Biclust
## call:
## biclust(x = ecoli, method = BCQU(), verbose = FALSE)
## Number of Clusters found: 100
## First 5 Cluster sizes:
## BC 1 BC 2 BC 3 BC 4 BC 5
## Number of Rows: 437 519 178 180 169
## Number of Columns: 29 22 52 50 53

res4 <- biclust(ecoli, method = BCQU(), weight = ecoli.weight, verbose = FALSE)
res4

## An object of class Biclust
## call:
## biclust(x = ecoli, method = BCQU(), weight = ecoli.weight, verbose = FALSE)
## Number of Clusters found: 100
## First 5 Cluster sizes:
## BC 1 BC 2 BC 3 BC 4 BC 5
## Number of Rows: 437 519 178 180 169
## Number of Columns: 29 22 52 50 53

Bicluster-expanding

we can expand existing biclustering results to recruit more genes according to certain consistency level:

res5 <- biclust(x = ecoli, method = BCQU(), seedbicluster = res, f = 0.25, verbose = FALSE)
summary(res5)

## An object of class Biclust
## call:
## biclust(x = ecoli, method = BCQU(), seedbicluster = res, f = 0.25, verbose = FALSE)
## Number of Clusters found: 20
## Cluster sizes:
## BC 1 BC 2 BC 3 BC 4 BC 5 BC 6 BC 7 BC 8 BC 9 BC 10 BC 11 BC 12 BC 13 BC 14
## Number of Rows: 593 151 51 110 117 68 84 27 43 20 36 30 17 19
## Number of Columns: 29 45 94 44 38 38 31 33 31 27 19 20 23 21
Biclusters comparison

We can compare the biclustering results obtained from different algorithms, or from a same algorithm with different combinations of parameter.

```r
test <- ecoli[1:50,]
res6 <- biclust(test, method = BCQU(), verbose = FALSE)
res7 <- biclust(test, method = BCCC())
res8 <- biclust(test, method = BCBimax())
showinfo (test, c(res6, res7, res8))
```

### References


