User guide of the PC-corr algorithm (version 2) for R

The PC- corr code (version 2) is available as a R function, that can be downloaded from the GitHub repository (https://github.com/biomedical-cybernetics/PC-corr_net/tree/master/R%20version). In this user guide, we show how to apply the PC-corr algorithm using as example the metagenomic dataset (available at https://github.com/biomedical-cybernetics/PC-corr_net/tree/master/R%20version).

First steps

1. The PC-corr algorithm (*PC_corr_v2.R*) is provided with the example dataset `amir4_gastricfluid_ppi.RData`: save them in the same folder, which will become the working directory.
2. Make sure that the .RData dataset contains all the necessary variables:
    a) the data matrix (numeric matrix MxN) with samples in rows and features in columns;
    b) the sample labels (character vector Mx1), identifying to which group the samples belong;
    c) the names of the features (lipids, genes, bacteria, etc.) (character vector Nx1) , necessary for the construction of the PC-corr network;
    d) the names of the samples (character vector Mx1), that, if needed, can be used to label the samples in the PCA plot.

The example dataset `amir4_gastricfluid_ppi.RData`, that was analysed in the paper, contains the following variables:

a) *x* - the data matrix (gastric fluid 16S metagenomes);
b) *sample_labels* – metadata specifying at which time point the eight patients were sampled, that is before (label: *before*) and after eight weeks of PPI treatment (label: *after*), for a total of 16 samples;
c) *feat_names* - contains the bacterial taxa;

d) *sample_names* - contains the samples' IDs.

The user can input any other –omic dataset taking care to prepare the input variables in the same way as the example dataset.

How to use the PC-corr algorithm in R

Now follow the steps below:

1. Load the data in the R workspace by typing like:

   ```
   load("amir4_gastricfluid_ppi.RData")
   ```

2. Load the PC-corr function from file by typing:

   ```
   source("PC_corr_v2.R")
   ```

3. Run the PC-corr function, giving as inputs the previously stated variables.

   Then you can call the function as

   ```
   PC_corr_v2(x,sample_labels,feat_names, sample_names)
   ```

   3.1 If you input less than 4 variables, an error will be returned, since you didn't provide all the necessary variables.

   For example, if you type by mistake

   ```
   PC_corr_v2(x,sample_labels)
   ```

   an error message will be displayed:

   ```
   Error in PC_corr_19_5_2017(x, sample_labels) :
   Not Enough Input Arguments
   ```

   3.2 Also, the analysis will not start if there are NaN values (standing for not a number values)

   They should be replaced in order to proceed:

   ```
   Error in PC_corr_v2 (x, sample_labels, feat_names, sample_names) :
   There are 2 NaN values in your data matrix.
   ```

```
Please replace them.
```

4. When the input is correct, you will be asked one question:

```
Is your data represented by
[r] ranked labels(labels that are organized according to a progressive order. e.g. different stages of a disease, where Stage 1 < Stage 2 < Stage 3)
[c] class labels (labels that are not necessary organized in a progressive order e.g. Condition A, Condition B, Condition C)? [r/c]:
```

If your data are represented by ranked labels, you type `r` (standing for ranked values) and another question will be shown:

```
Are the values of your ranked labels
[d]   discrete (Stage 1 < Stage 2 < Stage 3)
[con] continuous (different times of development of a cell line)? [d/con]:
```

In the example data, the labels are class labels (label: *before* or *after,* PPI treatment), hence you should input `c`.

5. The analysis starts, firstly by removing features that have the same values across all the samples, secondly by normalizing or not the dataset. You can either apply no normalization, one preferred single normalization (from the list reported below) or apply all the set of 11 different single normalizations (the list of normalizations is provided below) and no normalization, by typing after the question:

```
Do you want to apply:
[1] no normalization
[2] a preferred normalization
[3] automatically all the set of available normalizations? [1/2/3]
respectively 1, 2 or 3.
```

If you want to use a particular normalization, you type `2` and another question will be shown:

```
Input a preferred normalization from the following list:
```

DCS, DRS, LOG, ZSCORE, QUANTILE T, QUANTILE, ZSCORE T, PLUS(ABS(MIN)), PARETO SCALING, SQRT, MANORM

(For detailed information on the type of normalization, see the User guide)

Example: LOG

and you should input the normalization, that should be one from the list below.

List of Normalizations:

1. DRS: dividing by the row (sample) sum;
2. DCS: dividing by the column (features) sum;
3. LOG: logarithm with base 10 of each data element plus 1 (to avoid problems with 0 values). In case the data have negative values, remember to scale the minimum data value to 0 before to perform this normalization.
4. ZSCORE: z-score for each data element such that the features are centred to have mean 0 and scaled to have standard deviation 1;
5. QUANTILE T: quantile normalization over the samples;
6. QUANTILE: quantile normalization over the features;
7. ZSCORE T: z-score for each data element such that the samples are centred to have mean 0 and scaled to have standard deviation 1;
8. PLUS(ABS(MIN)): adding to each data element the minimum present in the data matrix, in absolute value.
9. PARETO SCALING: each feature is centred to have mean 0 and scaled by the square root of the standard deviation of the feature's values;
10. SQRT: square root of each data element;
11. MANORM: scaling the values in each feature, dividing by the mean of the feature.

Note: Some of the normalizations may be excluded from the analysis, in the case the normalized data contains infinite or NaN values or it is complex.

6. From the corresponding normalized dataset or the original dataset, the analysis continues by PCA analysis, either centred or not centred.

   The best discrimination in a PCA result (combination of normalization and centering) and along one dimension (principal component, PCn) can be assessed by different *evaluators*, depending on the type of labels:

   a) class labels: p-value (of Mann-Whitney U test), AUC, AUPR;

   b) discrete labels (ranked labels): p-value (of Mann-Whitney U test),), Area Under the ROC-Curve (AUC), Area Under the Precision-Recall curve (AUPR), correlation (Pearson and Spearman)

   c) continuous labels (ranked labels): correlation (Pearson and Spearman)

   In the case of more than two groups (that is more than two distinct label values, say $L>2$), the value shown will be the average of all the $\binom{L}{2}$ pairwise group comparison values.

   For the calculation of AUC and AUPR, you need to provide a positive label or, in the case of more than two groups ($L>2$), a positive label for each pairwise group comparison.

   - For two groups with same number of samples, you are asked to input the positive label:

     `Input the positive label for the calculation of AUC and AUPR values: Example: "after"`

     that for the example dataset will be *after* (inputting `"after"`).

   - In the case of more than two groups (that is more than two distinct label values, say $L>2$) that have a different number of samples, the positive class label for each pairwise comparison can be either the label of the largest/smallest sample group. In alternative, a list of ranked possible positive labels (containing $L-1$ distinct labels) can be inputted: in each pairwise comparison, the positive label will be one of the two compared labels with lowest

ranking (position) or the one that is present in the ranked list. Hence you have to type `l, s` or `r` respectively after the following question:

```
Do you want to calculate the AUC and AUPR values considering:
[s] as positive label, the label of the smallest sample group
[l] as positive label, the label of the largest sample group
[r] a ranked list of possible positive labels
```

In the case `r` is chosen, you should input the list of ranked labels, in the form shown in the example (that depends from your *sample_labels*):

```
Input the ranked list of possible positive labels for the
calculation of AUC and AUPR values:

Example: c("untr_HPpos","untr_HPneg")
```

- In the case of more than two groups (that is more than two distinct label values, say $L>2$) where at least two groups have the same number of samples, the user is asked to input a list of ranked positive label, as in the previous case.

7. Now you will be asked to choose the *evaluator*, with respect to which the results will be ordered on the screen, from the most discriminative to the least discriminative:

   a) for class labels:

```
Would you like to rank the PCA results by
[p]    P-value
[auc]  AUC
[aupr] AUPR? [p/auc/aupr]:
```

   b) for discrete labels (ranked labels):

```
Would you like to rank the PCA results by
[p]    P-value
[auc]  AUC
[aupr] AUPR
[pc]   Pearson correlation
[sc]   Spearman correlation? [p/auc/aupr/pc/sc]:
```

   c) for continuous labels (ranked labels):

```
Would you like to rank the PCA results by
[pc]   Pearson correlation
[sc]   Spearman correlation? [pc/sc]:
```

For the example data, you can rank the results by p-value, AUC or AUPR, typing respectively `p`,`auc` or `aupr`.

8. All the results are returned in an Excel table, named *result.xlsx* in the worksheet *PCA results*, automatically ranked with respect to the chosen evaluator from the most discriminative to the least discriminative

   In general, the table contains all the results for the complete set of evaluators, in consecutive columns, and reports in other separated columns:

   - the normalization of the dataset (*Norm*),
   - the centring of PCA that generated the results (*Centering*),
   - the PCA dimension (principal component PCn) (*Dim*).
   - The explained variance (*expl Var*) of the PCA dimension, which is the ratio, expressed as a percentage, of the variance accounted by the dimension over the total variance in all of the PCA dimensions.

   For more than two groups, the evaluator results are shown in multiple columns: one column contains the average values across all the pairwise group comparisons, followed by $\binom{L}{2}$ columns that have the specific values in the pairwise group comparison (specified in the header).

   Note: In the *Norm* column, - means that no normalization was applied to the data.

9. In addition, the results are also shown on the screen, in the command window, but only the best results are reported, ranked as in the Excel table, in order to assist the user in the creation of the network from the most discriminative PCn. In the case of more than two groups, only the average value (and not the values in each pairwise comparison) for each evaluator is present.

An example of results shown on the screen is presented in Figure S1, where the results of the example dataset are ordered with respect to the p-value estimator (p-value<0.05) from the lowest to the highest. For the calculation of AUC and AUPR, `"after"` was inputted as positive label.

P-value	AUC	AUPR	Norm	Centering	Dim	expl Var
0.01041181	0.875000	0.8975108	LOG	no	2	1.379534e+01
0.02066822	0.843750	0.8725852	LOG	yes	1	2.596061e+01
0.03791764	0.812500	0.8842720	QUANTILE	no	2	1.643984e+01
0.03791764	0.812500	0.8705721	SQRT	no	2	1.643409e+01
0.03791764	0.812500	0.8215436	QUANTILE T	no	3	1.227037e+01
0.03791764	0.812500	0.8301272	DCS	no	5	7.496535e+00
0.03791764	0.812500	0.8301272	MANORM	no	5	7.496535e+00
0.03791764	0.812500	0.8483860	SQRT	no	15	4.746616e-01
0.03791764	0.812500	0.8665179	QUANTILE	yes	1	2.252175e+01
0.03791764	0.812500	0.8241883	DCS	yes	3	1.122501e+01
0.03791764	0.812500	0.8241883	MANORM	yes	3	1.122501e+01
0.03791764	0.812500	0.8483860	PARETO SCALING	yes	16	3.923722e-30
0.04988345	0.796875	0.8649267	PLUS(ABS(MIN))	no	2	2.573272e+01
0.04988345	0.796875	0.8649267	-	no	2	2.573272e+01
0.04988345	0.796875	0.7953474	DCS	no	14	1.465720e+00
0.04988345	0.796875	0.7953474	MANORM	no	14	1.465720e+00
0.04988345	0.796875	0.8476555	PLUS(ABS(MIN))	yes	1	5.756417e+01
0.04988345	0.796875	0.8476555	-	yes	1	5.756417e+01
0.04988345	0.796875	0.7953474	DCS	yes	13	1.610657e+00
0.04988345	0.796875	0.7953474	MANORM	yes	13	1.610657e+00

Figure S1. Table of results shown in the command window for the example dataset, ranked with respect to the p-value (first column).

Note: For some cases, the R code can give different results from MATLAB, due to differences in the implementation of the specific built-in functions.

10. After seeing the ranked results, you can choose any combination of normalization, centering ,dimension (PCn) and cut-off for the network to obtain the PC-corr network according to your interest (or need), by replying to four questions:

a) `Select the normalization:`

`Examples: LOG or –`

`where - stands for no normalization.`

b) `Centering version?[y/n]`

[y] yes, centred PCA

[n] no, non-centred PCA

c) Select the dimension for generating the PC-corr network:

d) Select a cut-off or a set of cut-offs for generating the PC-corr network [number between 0 and 1]:

Examples: 0.6 or c(0.6, 0.65, 0.7)

Note: In the case you choose either to apply no normalization or a preferred normalization at the beginning of the analysis, only the questions in b), c) and d) will appear in the command window (excluded question in a)).

For example, if you want to create the PC-corr network from the combination that exhibits the lowest p-value of Mann-Whitney test in Figure S1 (first row in the table), you have to type after each question respectively: `LOG` (for a)) , `n` (for b)), `2` (for c)) and `0.75` (for d)).

Moreover, in the command window it will be reported the number of features that were discarded since their processed loading values didn't surpass the chosen cut-off/cut-offs, like in the following example (for the previous choices on the example dataset):

```
219 features were deleted because |V_feature|<cutoff
```

in addition to some information on the PC-corr network, similar to the following (for the previous choices on the example dataset):

```
At cut-off 0.75 the PC-corr network has 15 nodes and 25 edges.
```

Instead, if you want to visualize the network at two different cut-offs 0.7 and 0.75, you can type after question d):

```
c(0.7,0.75)
```

Note:

If the chosen cut-off removes all the edges in the PC-corr network, a message will be shown:

```
With this cut-off, there are no edges that have |PC_corr(i,j)| > cutoff.
```

and it will suggest to consider a cut-off below a certain value, similar to the following message, that prompts when choosing a 0.95 cut-off for the example dataset:

```
Try with another cut-off less than 0.85.
```

Then, it will be asked to input another cut-off, that for the example dataset can be a number between 0 and 0.85:

```
Then select another cut-off for generating the PC-corr network [number between 0 and 0.85]:
```

11. Depending on the user options, the (centred or non-centred) PCA will be automatically plotted in the two-dimensional space, where the y-axis is the chosen dimension (in our example `2)` and the x-axis represents the best discriminating dimension, on the basis of the chosen evaluator. In case the best discriminating dimension is the selected dimension, it will be the first dimension in the PCA plot and the second dimension will be the second best discriminating dimension.

    The black colour will be assigned to the sample group more on the left side of the PCA plot, while the sample group more on the right will have a red colour and, in the case of more than two groups, the colours of the other groups are random.

    In addition, the figure contains the probability density estimate for the sample groups and the evaluator values (average values if more than two groups are present) along the two axes.

    In the example data, the PCA result is shown in the (PC2, PC13) space (Figure S2).

    - **Optional**

      You can decide to label the samples on the PCA plot with their ID, by adding another argument to the PC-corr function (`'yes'` or `'no'`), like in the following example:

      ```
      PC_corr_v2(x,sample_labels,feat_names, sample_names, 'yes')
      ```

      The default is `'no'`, that is the samples will be plotted without any label.

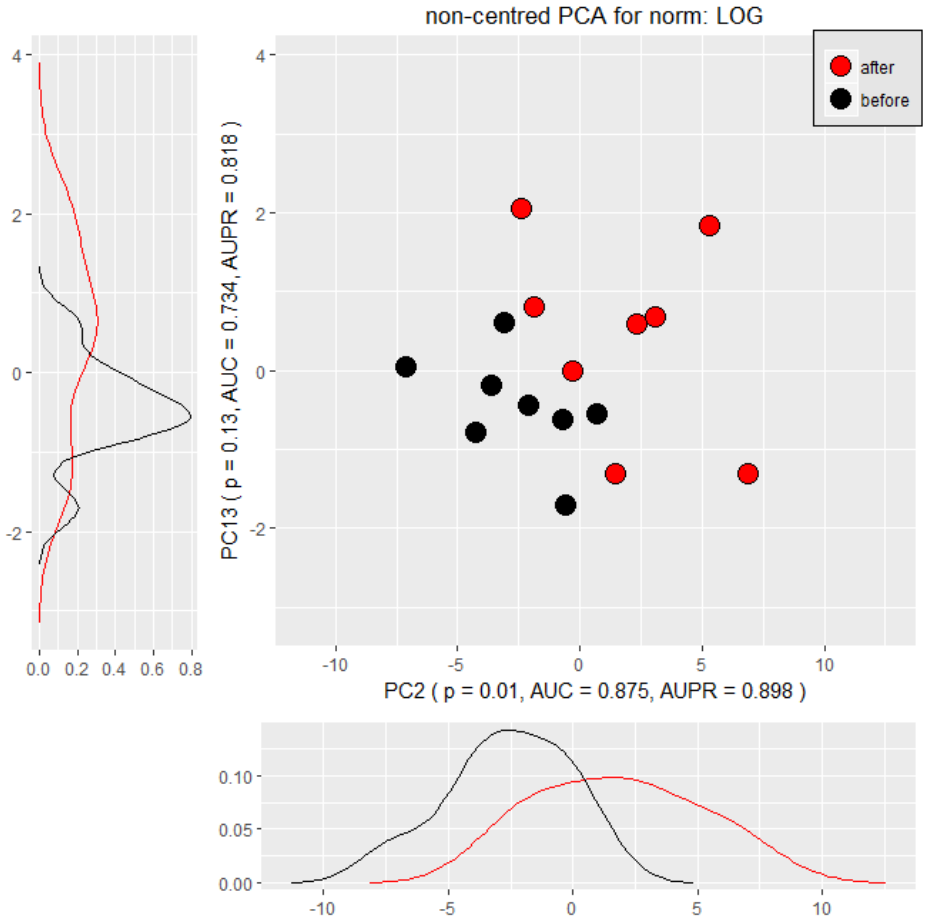

Figure S2. PCA plot for the example dataset

12. In addition, you will get another figure associated to the PCA plot, that shows on top the evaluator value for each of the PCA dimensions and on the bottom the percentage of explained variance accounted for by each principal component. Figure S3 was obtained from the example data, and corresponds to the PCA result in Figure S2. In the figure, the best discriminating PCA dimension is displayed in a different colour (sky blue) and its evaluator value is reported in red on top of its bar in the top panel of the figure.

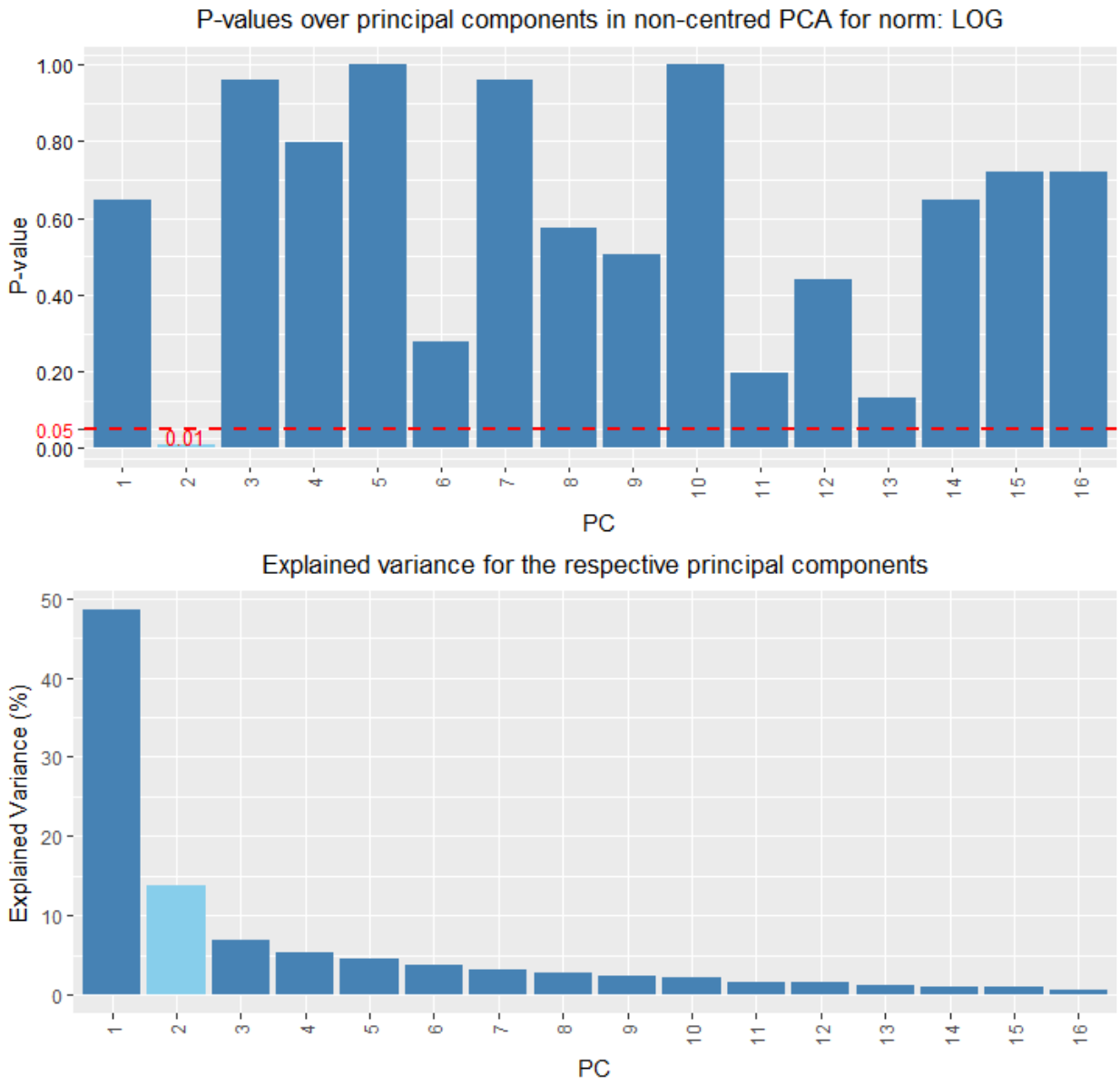

Figure S3. Example of figure showed after the PCA plot in Figure S2 (for the example dataset), displaying the P-value over all the principal components and the variance (as a percentage) explained by each dimension. Dimension two is the most discriminative in the example since the p-value is the lowest.

13. According to your choices and the selected cut-off/cut-offs, the PC-corr network is then constructed at that/those particular cut-off/cut-offs.

The tables of edge and node values of the PC-corr network are reported in an Excel file named *PC-corr_net_edges-nodes_results.xlsx* (spreadsheet named for instance *Edges – cutoff 0.75* and *Nodes – Cutoff 0.75*), so that the user can easily visualize the graph with another network visualization program, like Cytoscape (http://www.cytoscape.org/).

When more than one cut-off is chosen, the nodes and edges of each cut-off are saved in separate worksheets.

14. Additionally, you can decide to visualise the PC-corr network in another image, by replying to the following question:

`Do you want to visualize the PC-corr network? [y/n]:`

Figure S4 shows the obtained PC-corr network (by typing `y`) for the example data.

In the network the red and black edges represent positive and negative interactions respectively. On the other hand, the colour of the nodes reflects whether the features have higher values (in mean) in the red or black group of samples. The presence of grey dashed edges indicates edges under frustration (see the Result section of the article for more details), whose fraction (that is the ratio of number of edges under frustration to the total number of edges) is reported in the PC-corr network.

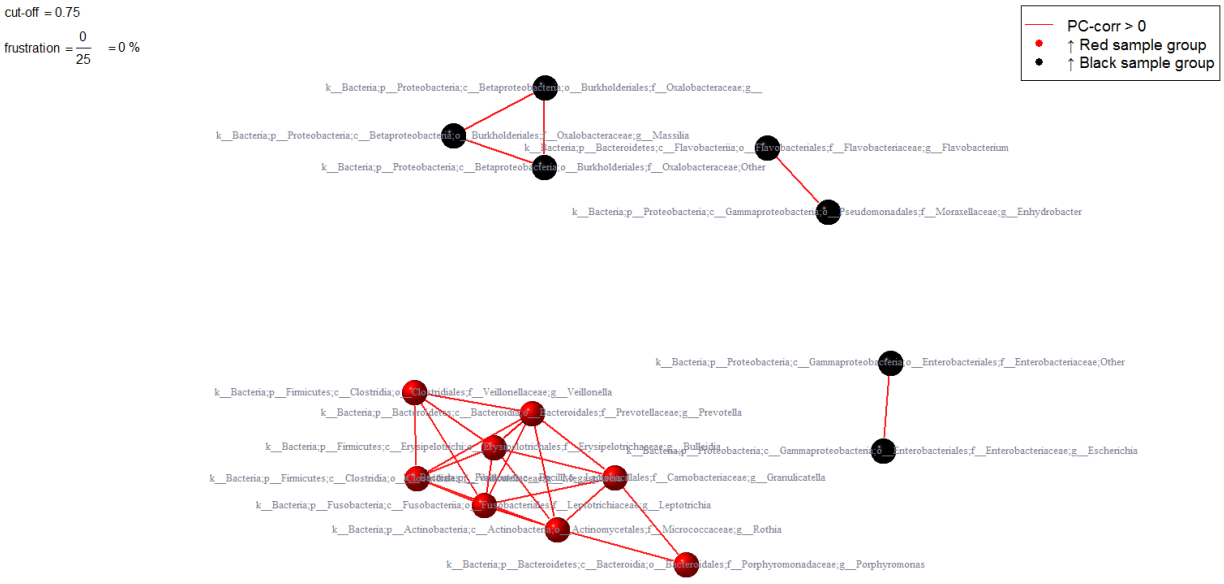

Figure S4. PC-corr network of the example dataset under cut-off =0.75.

15. Additionally, you can compute or the trustworthiness of p-value, AUC and AUPR results, that is a measure of significance of segregation. The trustworthiness (of p-value/AUC/AUPR results) evaluates if the results are discriminative because it is random (trustworthiness p-value>0.05) or it captures main variability (trustworthiness p-value<=0.05). This is especially useful for measuring the significance of segregation especially for discriminative results in PCA dimensions with low explained variance.

    You can either choose not to compute it or calculate it for either your selected case (selected normalization, dimension and centering) or for all the cases (every combination of normalization, dimension and centering), by typing respectively 1, 2 or 3 after the following question:

    ```
    Would you like to compute the trustworthiness of measure of segregation (p-value,AUC and AUPR)? [1/2/3]
    [1] no
    [2] yes, but just for the selected case (normalization, dimension and centering)
    [3] yes, for all the cases
    ```

    The trustworthiness of p-value, AUC and AUPR measures of segregation will appear in the Excel table of results (*result.xlsx*, in worksheet *PCA results)* in three separate columns (*Trustworthiness(p-value), Trustworthiness(AUC), Trustworthiness(AUPR)*) .

16. If you would like to compute the trustworthiness of the measure of segregation (p-value, AUC, AUPR) for each case (option 3 in the previous question), an additional question is asked for confirmation:

    ```
    Are you sure? Do you want to compute for all? It will take some time (hours). [y/n]
    (We suggest running it over night)
    [y] yes, I want to compute for all the cases
    [n] no, I don't want to compute for all the cases
    ```

    In the case the user is sure to compute the trustworthiness for each case (y is typed), the computation will take some hours.

Note: since the computation may take some time, a text progress bar will indicate the percentage of computation complete

17. Finally, the function returns two outputs for the inferred PC-corr network:
    a. The interacting features (*Nodes*),
    b. The interactions between the features (*Edges*).

- For a single chosen cut-off, *Nodes* and *Edges* contain respectively (in data frames) the node and edge tables for the PC-corr network at that cut-off, that were saved in a Excel file.
- When you choose multiple cut-offs, *Nodes* and *Edges* will contain the imputed cut-offs in the first column and in the second column you will find the respective node and edge tables (in matrix of lists) for the PC-corr network cut at those thresholds, that were also saved in an Excel file.

Recommendation

We recommend checking PC-corr networks that you obtain using different normalizations that are significant, because different normalizations can put a different focus on the original features, providing PC-corr networks that emphasizes group of interacting features that emerges at different levels of abundance. An example is shown in Figure S5, where panel A and B show the PC-corr network at 0.6 cut-off obtained from a lipidomic datasets using two different normalizations that are significant (LOG, panel A, and ZSCORE, panel B). Since ceramides (Cer) are low abundant species, they do not emerge with the LOG normalization in the PC-corr network (even at lower cut-off, for example at 0.5 cut-off), while with ZSCORE (that means all the lipid species are centred to have mean 0 and standard deviation 1) these lipid species are present in the network.

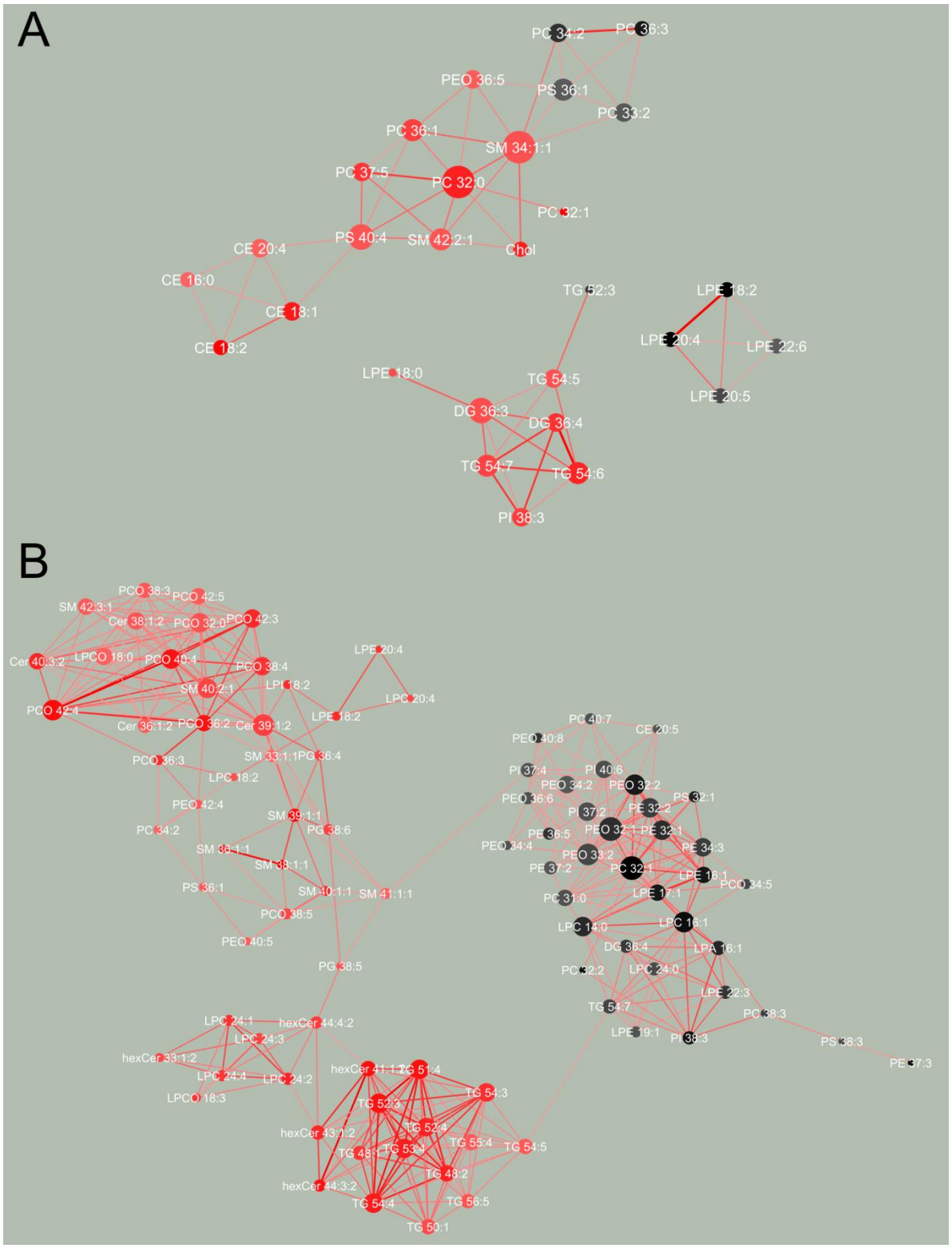

Figure S5. PC-corr network (cut-off=0.6) constructed from a lipidomic dataset using two different normalizations. (A) For LOG normalization. (B) For ZSCORE normalization.