User guide of the PC-corr algorithm for MATLAB

The PC- corr code is available as a MATLAB function (for MATLAB 2014b or later versions), that can be downloaded from the github repository (https://github.com/biomedical-cybernetics/PC-corr_net/tree/master/MATLAB%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/MATLAB%20version).

First steps

1. The PC-corr algorithm (*PC_corr.m*) is provided with the example dataset `amir4_gastricfluid_ppi.mat`: save them in the same folder, which will become the working directory.
2. Make sure that the .mat 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 (cell array Mx1), identifying to which group the samples belong;
    c) the names of the features (lipids, genes, bacteria, etc.) (cell array Nx1) , necessary for the construction of the PC-corr network;
    d) the names of the samples (cell array Mx1), that, if needed, can be used to label the samples in the PCA plot.

The example dataset `amir4_gastricfluid_ppi.mat`, 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 MATLAB

Now follow the steps below:

1. Load the data in the MATLAB workspace by double clicking on it in the MATLAB Current Folder window, or by typing like:

```
load amir4_gastricfluid_ppi.mat
```

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

   Then you can call the function as

```
[Edges, Nodes]=PC_corr(x,sample_labels,feat_names,sample_names)
```

   2.1 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

```
[Edges, Nodes]=PC_corr(x,sample_labels)
```

   an error message will be displayed:

```
Error using PC_corr (line 47)
Not Enough Input Arguments
```

   2.2 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 using PC_corr (line 53)
There are 2 NaN values in your data matrix.
```

```
Please replace them.
```

3. 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)
or [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`.

4. The analysis starts, firstly by removing features that have the same values across all the samples, secondly by normalizing or not the dataset with 11 different single normalizations (the list of normalizations is provided below) and then obtaining from the corresponding normalized dataset or the original dataset the PCA result, either centred or not centred.

List of Normalizations:

1. DSOR: dividing by the sum over the samples;
2. DSOC: dividing by the sum over the features;
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.

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 test), AUC, AUPR;
b) discrete labels (ranked labels): p-value (of Mann-Whitney test), AUC, 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.

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-value, AUC or AUPR [p/auc/aupr]:`

b) for discrete labels (ranked labels):

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

c) for continuous labels (ranked labels):

`Would you like to rank the PCA results by Pearson correlation or 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`.

5. All the results are returned in an Excel table, named *result.xlsx* in the spreadsheet *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).

6. 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.

'P-value'	'AUC'	'AUPR'	'Norm'	'Centering'	'Dim'	'expl Var'
[0.0104]	[0.8750]	[0.7725]	'LOG'	'no'	[2]	[13.7953]
[0.0207]	[0.8438]	[0.6680]	'LOG'	'yes'	[1]	[25.9606]
[0.0379]	[0.8125]	[0.7593]	'QUANTILE'	'no'	[2]	[16.4398]
[0.0379]	[0.8125]	[0.7456]	'SQRT'	'no'	[2]	[16.4341]
[0.0379]	[0.8125]	[0.6605]	'QUANTILE T'	'no'	[3]	[12.2704]
[0.0379]	[0.8125]	[0.7051]	'DSOR'	'no'	[5]	[7.4965]
[0.0379]	[0.8125]	[0.7051]	'MANORM'	'no'	[5]	[7.4965]
[0.0379]	[0.8125]	[0.7234]	'SQRT'	'no'	[15]	[0.4747]
[0.0379]	[0.8125]	[0.7415]	'QUANTILE'	'yes'	[1]	[22.5217]
[0.0379]	[0.8125]	[0.6992]	'DSOR'	'yes'	[3]	[11.2250]
[0.0379]	[0.8125]	[0.6992]	'MANORM'	'yes'	[3]	[11.2250]
[0.0499]	[0.7969]	[0.7399]	'PLUS(ABS(MIN))'	'no'	[2]	[25.7327]
[0.0499]	[0.7969]	[0.7399]	'-'	'no'	[2]	[25.7327]
[0.0499]	[0.7969]	[0.6703]	'DSOR'	'no'	[14]	[1.4657]
[0.0499]	[0.7969]	[0.6703]	'MANORM'	'no'	[14]	[1.4657]
[0.0499]	[0.7969]	[0.6524]	'PLUS(ABS(MIN))'	'yes'	[1]	[57.5642]
[0.0499]	[0.7969]	[0.6524]	'-'	'yes'	[1]	[57.5642]
[0.0499]	[0.7969]	[0.6569]	'DSOR'	'yes'	[13]	[1.6107]
[0.0499]	[0.7969]	[0.6569]	'MANORM'	'yes'	[13]	[1.6107]

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

7. After seeing the ranked results, you can choose any combination of normalization, centring ,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:`

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

   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]:`

   `Example: [0.6 0.65 0.7]`

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)).

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

```
[0.7 0.75]
```

Note:

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

```
Warning: 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 warning, that prompts when choosing a 0.95 cut-off for the example dataset:

```
Warning: Try with another cut-off lower than 0.85836.
```

8. 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:

```
[Edges,Nodes]=PC_corr(x,sample_labels,feat_names, sample_names, 'yes')
```

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

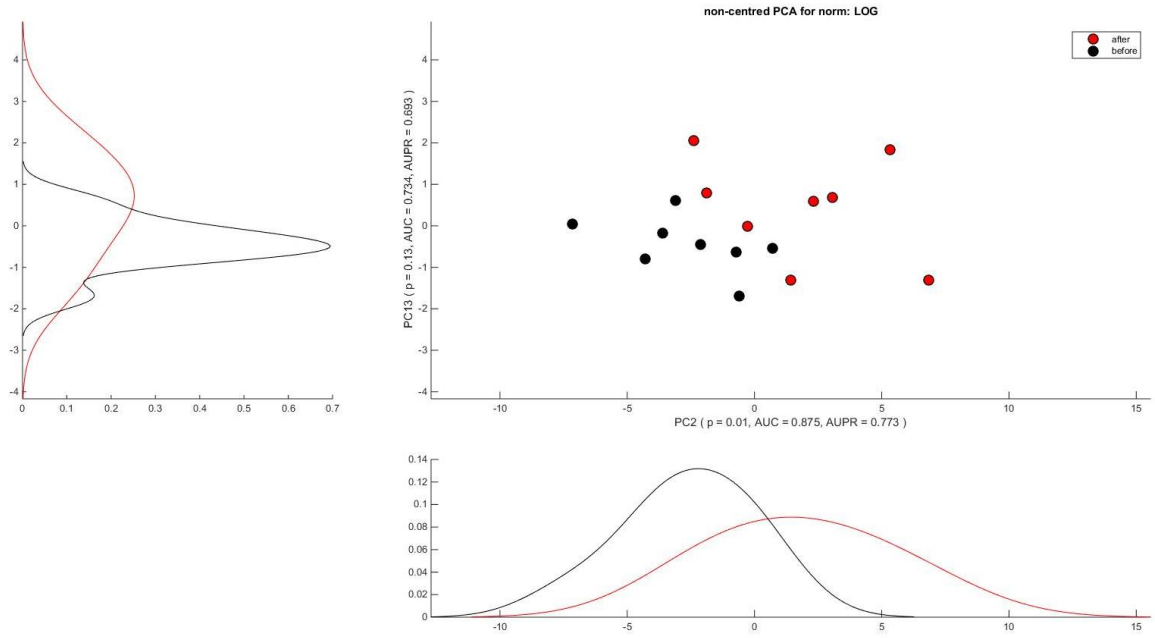

Figure S2. PCA plot for the example dataset

9. 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.

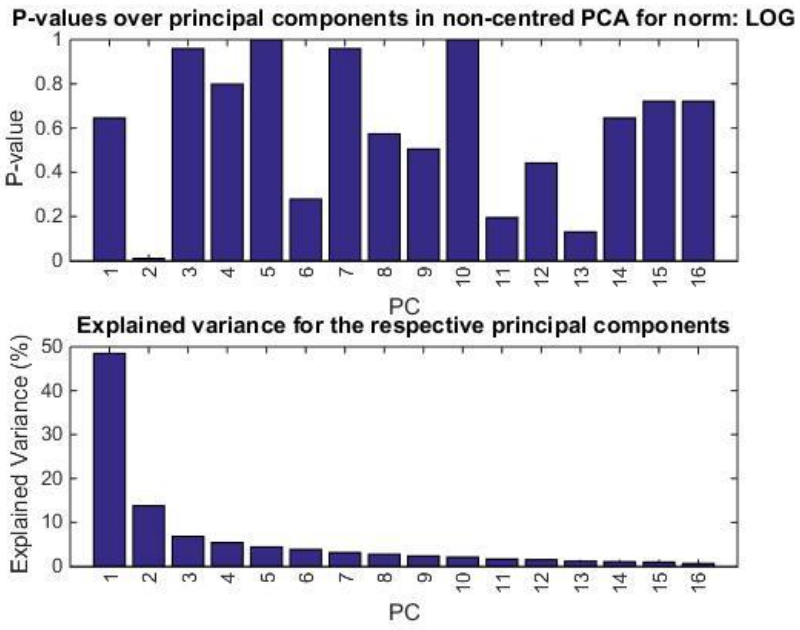

Figure S3. Example of figure showed after the PCA plot in Figure S2 (for the example dataset), displaying the P-value over all the principal components and the variance (as a percentage) explained by each dimension. Dimension two is the most discriminative in the example since the p-value is the lowest.

10. 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.

In the command window you can see the list of inferred associations between features (edges) and edge values (Figure S4), as well as the interacting features (nodes), their corresponding colour and loading value (Figure S5), and also visualise the PC-corr network (Figure S6) in another image. 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.

In addition, 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 *Edges* and *Nodes*), 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 printed on the screen, and concurrently saved in separate sheets (named for instance *Edges – cutoff 0.75* and *Nodes – Cutoff 0.75*).

edges =

'Node i'	'Node j'	'PC-corr(i,j)'
'k__Bacteria;p__Actinobacteria;c__A...'	'k__Bacteria;p__Bacteroidetes;c__Ba...'	[0.7545]
'k__Bacteria;p__Actinobacteria;c__A...'	'k__Bacteria;p__Bacteroidetes;c__Ba...'	[0.8407]
'k__Bacteria;p__Actinobacteria;c__A...'	'k__Bacteria;p__Firmicutes;c__Bacil...'	[0.7596]
'k__Bacteria;p__Actinobacteria;c__A...'	'k__Bacteria;p__Firmicutes;c__Clost...'	[0.7555]
'k__Bacteria;p__Actinobacteria;c__A...'	'k__Bacteria;p__Firmicutes;c__Erysi...'	[0.7687]
'k__Bacteria;p__Actinobacteria;c__A...'	'k__Bacteria;p__Fusobacteria;c__Fus...'	[0.8198]
'k__Bacteria;p__Bacteroidetes;c__Ba...'	'k__Bacteria;p__Firmicutes;c__Bacil...'	[0.7545]
'k__Bacteria;p__Bacteroidetes;c__Ba...'	'k__Bacteria;p__Firmicutes;c__Bacil...'	[0.7596]
'k__Bacteria;p__Bacteroidetes;c__Ba...'	'k__Bacteria;p__Firmicutes;c__Clost...'	[0.7555]
'k__Bacteria;p__Bacteroidetes;c__Ba...'	'k__Bacteria;p__Firmicutes;c__Clost...'	[0.7808]
'k__Bacteria;p__Bacteroidetes;c__Ba...'	'k__Bacteria;p__Firmicutes;c__Erysi...'	[0.8584]
'k__Bacteria;p__Bacteroidetes;c__Ba...'	'k__Bacteria;p__Fusobacteria;c__Fus...'	[0.8354]
'k__Bacteria;p__Bacteroidetes;c__Fl...'	'k__Bacteria;p__Proteobacteria;c__G...'	[0.7634]
'k__Bacteria;p__Firmicutes;c__Bacil...'	'k__Bacteria;p__Firmicutes;c__Erysi...'	[0.7596]
'k__Bacteria;p__Firmicutes;c__Bacil...'	'k__Bacteria;p__Fusobacteria;c__Fus...'	[0.7596]
'k__Bacteria;p__Firmicutes;c__Clost...'	'k__Bacteria;p__Firmicutes;c__Clost...'	[0.7555]
'k__Bacteria;p__Firmicutes;c__Clost...'	'k__Bacteria;p__Firmicutes;c__Erysi...'	[0.7555]
'k__Bacteria;p__Firmicutes;c__Clost...'	'k__Bacteria;p__Fusobacteria;c__Fus...'	[0.7555]
'k__Bacteria;p__Firmicutes;c__Clost...'	'k__Bacteria;p__Firmicutes;c__Erysi...'	[0.7812]
'k__Bacteria;p__Firmicutes;c__Clost...'	'k__Bacteria;p__Fusobacteria;c__Fus...'	[0.7930]
'k__Bacteria;p__Firmicutes;c__Erysi...'	'k__Bacteria;p__Fusobacteria;c__Fus...'	[0.8354]
'k__Bacteria;p__Proteobacteria;c__B...'	'k__Bacteria;p__Proteobacteria;c__B...'	[0.8000]
'k__Bacteria;p__Proteobacteria;c__B...'	'k__Bacteria;p__Proteobacteria;c__B...'	[0.7877]
'k__Bacteria;p__Proteobacteria;c__B...'	'k__Bacteria;p__Proteobacteria;c__B...'	[0.7877]
'k__Bacteria;p__Proteobacteria;c__G...'	'k__Bacteria;p__Proteobacteria;c__G...'	[0.7684]

Figure S4. The edges in the PC-corr network of the exampledataset at 0.75 cut-off.

```
nodes =

    'Node'                                  'Colour'    'Loading (V)'
    'k__Bacteria;p__Actinobacteria;c__...'  'Red'       [     0.8407]
    'k__Bacteria;p__Bacteroidetes;c__...'   'Red'       [     0.7545]
    'k__Bacteria;p__Bacteroidetes;c__...'   'Red'       [     0.9922]
    'k__Bacteria;p__Bacteroidetes;c__...'   'Black'     [    -0.8856]
    'k__Bacteria;p__Firmicutes;c__Bac...'   'Red'       [     0.7596]
    'k__Bacteria;p__Firmicutes;c__Clo...'   'Red'       [     0.7555]
    'k__Bacteria;p__Firmicutes;c__Clo...'   'Red'       [     0.8872]
    'k__Bacteria;p__Firmicutes;c__Ery...'   'Red'       [     0.8584]
    'k__Bacteria;p__Fusobacteria;c__F...'   'Red'       [     0.8354]
    'k__Bacteria;p__Proteobacteria;c_...'   'Black'     [    -0.8000]
    'k__Bacteria;p__Proteobacteria;c_...'   'Black'     [    -0.9564]
    'k__Bacteria;p__Proteobacteria;c_...'   'Black'     [    -0.7877]
    'k__Bacteria;p__Proteobacteria;c_...'   'Black'     [         -1]
    'k__Bacteria;p__Proteobacteria;c_...'   'Black'     [    -0.7684]
    'k__Bacteria;p__Proteobacteria;c_...'   'Black'     [    -0.7634]
```

Figure S5. The nodes in the PC-corr network of the example dataset at 0.75 cut-off.

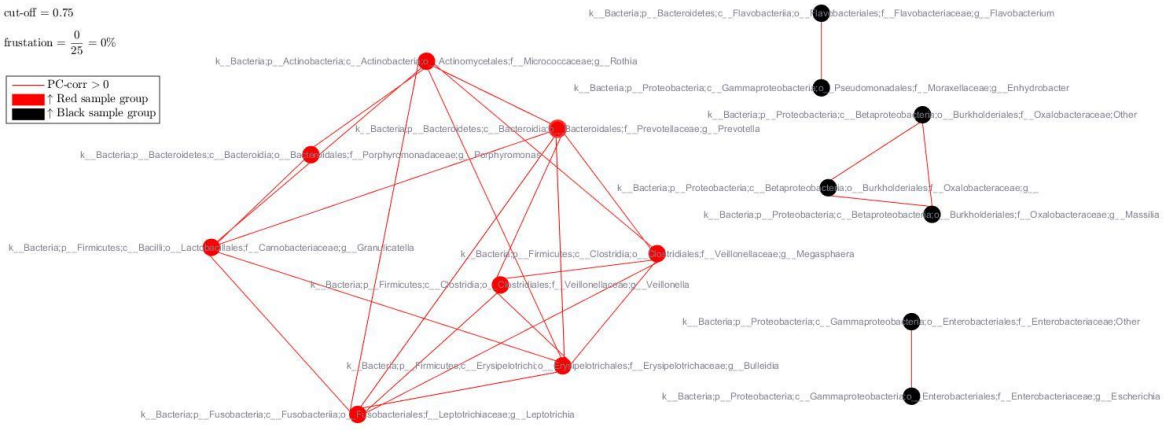

Figure S6. PC-corr network of the example dataset under cut-off =0.75.

11. 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 cell arrays) the node and edge tables for the PC-corr network at that cut-off, that were shown on the screen and saved in 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 cell

arrays) for the PC-corr network cut at those thresholds, that were also shown on the screen and saved in Excel files.