MultiPEN

MultiPEN uses a penalised logistic regression approach to find a subset of features (genes and/or metabolites) that hold more discriminant power to separate two classes, namely control and cases [1]. Such approach uses a molecular interaction network (e.g., protein-protein interaction network or metabolic network) to find the largest connected component that best separates the two conditions (for details on the logistic regression program to be optimised refer to [1]).

**TABLE OF CONTENTS**

Getting Started 1

Cross Validation 2

Syntax 2

Description 2

Cross Validation Output Files 3

Example - OS 3

Feature Selection 5

Syntax 5

Description 5

Feature Selection Output Files 5

Example - OS 6

Hierarchical Clustering 6

Syntax 6

Description 6

Output Files 6

Example - OS 6

Module Name 7

Syntax 7

Description 7

Output Files 7

Example - OS 7

References 7

# Getting Started

MultiPEN is shared as a MATLAB stand-alone application, which requires the installation of the MATLAB Runtime for R2015b in your system.

1. Download and save MATLAB Runtime for R2015b for your operative system which can be found from:

<http://www.mathworks.com/products/compiler/mcr/index.html>

1. Double click the installer and follow the instructions in the installation wizard.

# Cross Validation

A common practice in the machine learning community is to first solve for the λ parameter that optimises the logistic regression problem in equation 1 for your specific data. In MultiPEN the module to do precisely that is *crossValidation*.

## Syntax

*MultiPEN* **crossValidation** *OutputDirectory* *ExpressionData* *Interactions SampleClass lambdas Folds NumIterations*

## Description

MultiPEN

This is the path to the binary executable of MultiPEN, i.e., binary-OS/MultiPEN\_v001\_OS/.

OutputDirectory

Specify directory for output files.

ExpressionData

The expression data is in tabular format where the rows are the features (genes and/or metabolites) and the columns are the samples. An example of a file containing expression data is shown in Figure 1 c).

Interactions

The interaction matrix where the ith interaction (row) is represented as:

*A* *B* *score*

where *A* and *B* are indexes of the connected nodes and score is a number in the range [0,1] representing the interaction confidence score (where 1 corresponds to the maximum level of confidence). An example is shown in Figure 1 b).

SampleClass

For each sample specify if control (0) or case (1). An example of this file is shown in Figure 1 a) where each row contains the class for one sample.

lambdas

Set of lambdas to test for cross validation

Folds

For cross validation.

NumIterations

Maximum number of iterations for the optimisation solver. Default value is 100.

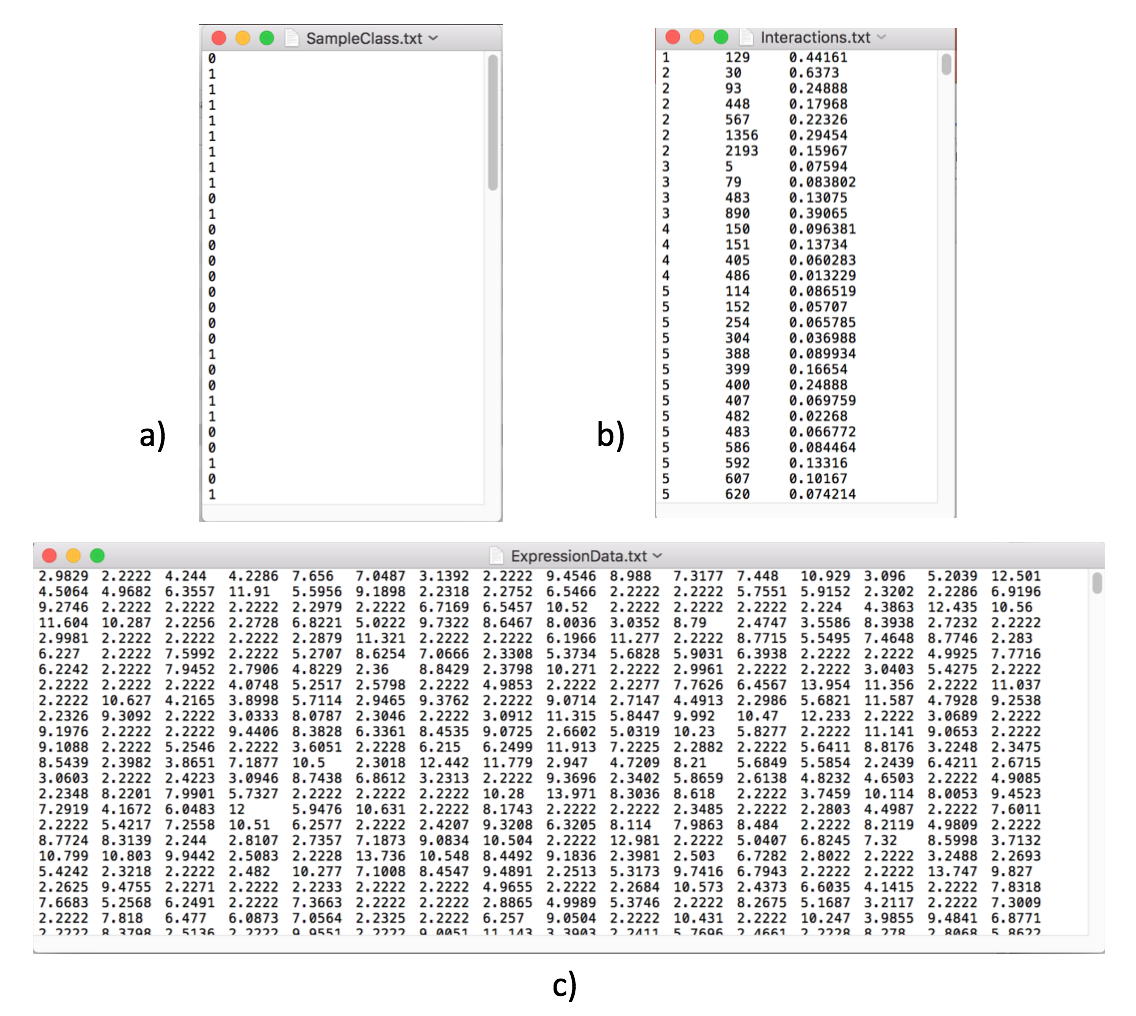


Figure . Cross Validation: Example of input files

## Cross Validation Output Files

cross-validation\_statst.txt

Statistics for tests which include, for each lambda, the size of the largest connected component (LCC), the standard deviation of the largest connected component (std\_LCC), the number of selected features (selected, i.e., features which weights are different to zero), area under the curve (AUC), and the standard deviation of the area under the curve (std\_AUC).

## Example - OS

To run MultiPEN open a terminal. Navigate to the folder where the binary for MultiPEN is located, i.e., binary-OS/MultiPEN\_v001\_OS/. Then create variables for the paths to stand-alone application, output directory and input files by typing:

MultiPEN="MultiPEN.app/Contents/MacOS/applauncher"

OutputDirectory="ExampleOutputs/"

ExpressionData="ExampleInputs/X.txt"

Interactions="ExampleInputs/E.txt"

SampleClass="ExampleInputs/Y.txt"

lambda=0.001

Folds=3

NumIter=100

Note that in this example we are using the example files provided with the application. All the files used as example as located in the folder: *ExampleInputs/*.

Next, run Cross Validation with the following command:

$MultiPEN **crossValidation** $OutputDirectory $ExpressionData $Interactions $SampleClass $lambda $Folds $NumIter

# Feature Selection

After selecting the best λ parameter to optimise the logistic regression problem, *featureSelection* performs the ranking of all features (both genes and metabolites or a combination of genes and metabolites) based on their expression (genes) and/or levels (metabolites).

## Syntax

*MultiPEN* **featureSelection** *OutputDirectory* *ExpressionData* *Interactions SampleClass Lambdas FeatureNames SampleNames*

## Description

MultiPEN

This is the path to the binary executable of MultiPEN, i.e., binary-OS/MultiPEN\_v001\_OS/.

OutputDirectory

Specify the directoy for the output files.

ExpressionData

The expression data in tabular format where the rows are the features (genes and/or metabolites) and the columns are the samples.

Interactions

The interaction matrix where the ith interaction (row) is represented as:

*A* *B* *score*

where *A* and *B* are indexes of the connected nodes and score is a number in the range [0,1] representing the interaction confidence score (where 1 corresponds to the maximum level of confidence).

SampleClass

For each sample specify if control (0) or case (1).

lambda

For the optimisation problem

FeatureNames

One name per feature and in the same order as the corresponding rows in the expression matrix.

SampleNames

Sample name in the same order as the columns in the expression matrix.

## Feature Selection Output Files

MultiPEN-Rankings\_lambda*X*(.mat/.txt) <insert 2016-09-27\_example-output\_rankings.png>

Ranking of features for the corresponding lambda *X*.

MultiPEN-Rankings\_lambda0.001\_genes-higher-in-cases.txt

Ranking of features which includes only features with higher expression in cases samples.

MultiPEN-Rankings\_lambda0.001\_genes-higher-in-control.txt

Ranking of features which includes only features with higher expression in control samples.

MultiPEN-vts\_lambda0.001.txt

Intercept term (logistic regression model)

## Example - OS

To run MultiPEN open a terminal. Navigate to the folder where the binary for MultiPEN is located, i.e., binary-OS/MultiPEN\_v001\_OS/. Create variables for the paths to stand-alone application, output path and all (example) input files.

MultiPEN="MultiPEN.app/Contents/MacOS/applauncher"

OutputDirectory="ExampleOutputs/"

ExpressionData="ExampleInputs/X.txt"

Interactions="ExampleInputs/E.txt"

SampleClass="ExampleInputs/Y.txt"

lambda=0.001

FeatureNames="ExampleInputs/features.txt"

SampleNames="ExampleInputs/samples.txt"

Run feature selection

$MultiPEN **featureSelection** $OutputDirectory $ExpressionData $Interactions $SampleClass $lambda $FeatureNames $SampleNames

# Hierarchical Clustering

## Syntax

## Description

## Output Files

Default Output folder:

output\_MultiPEN/stats/hierarchical\_clustering.png

## Example - OS

$MultiPEN **hierarchicalClustering** $OutputDirectory $ExpressionData $SampleNames $FeatureNames $Threshold $PlotTitle

Threshold to filter expression is optional, which is exclusive (>).

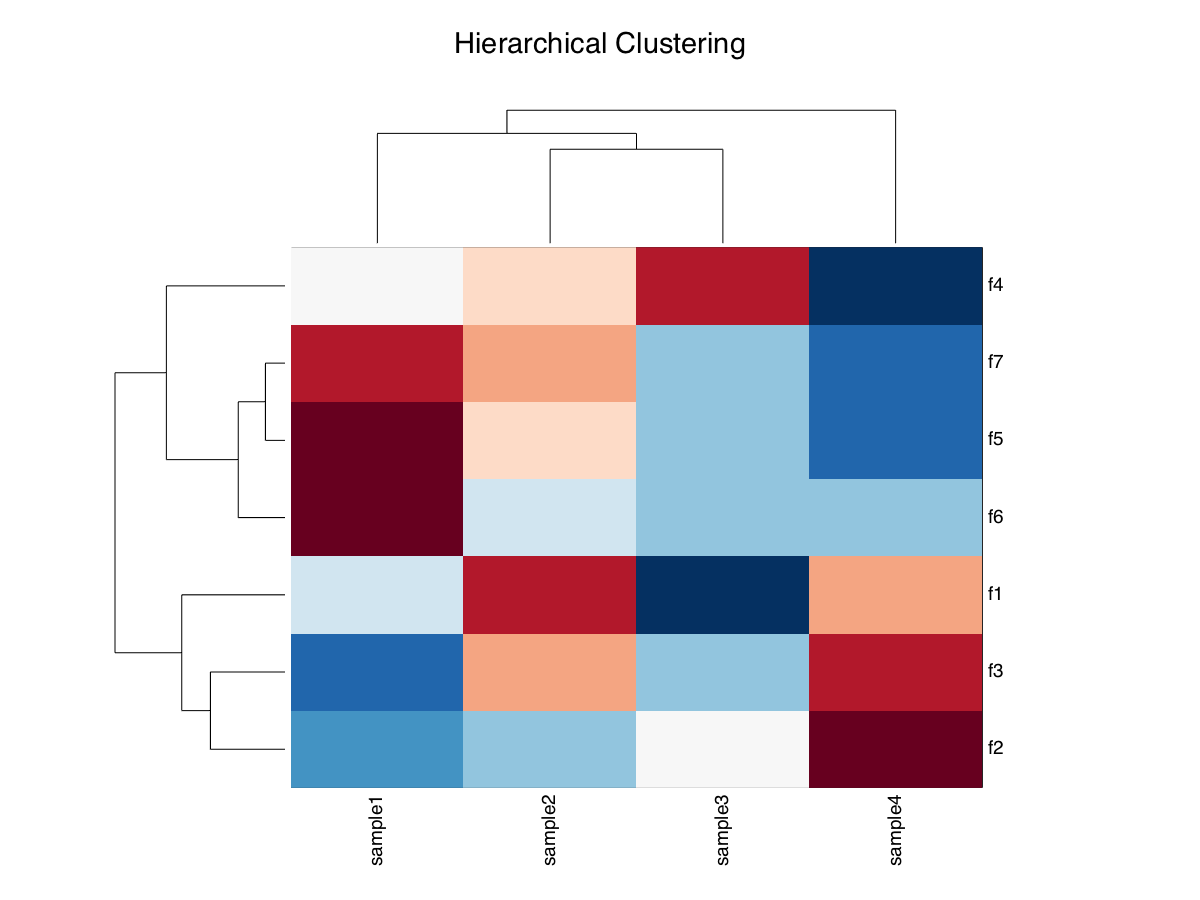


Figure 2: Example of a figure for hierarchical clustering

# Module Name

## Syntax

## Description

## Output Files

## Example - OS

# References

[1] GenePEN: analysis of network activity alterations in complex diseases via the pairwise elastic net., Vlassis N, Glaab E., Stat Appl Genet Mol Biol. 2015 Apr;14(2):221-4. doi: 10.1515/sagmb-2014-0045.