How to perform PIDC using the R package *minet*

What is PIDC?

PIDC is a fast algorithm that uses partial information decomposition (PID) to identify regulatory relationships between genes. Heterogeneity in single-cell transcriptomic data carries information about gene-gene interactions. Use multivariate information theory to explore the statistical dependencies/relationships between triplets of genes in single-cell gene expression datasets.

Introduction:

The R package *minet* stands for Mutual Information network inference. The package provides a set of functions to infer mutual information networks from a dataset. The package returns a network where nodes denote genes, edges model statistical dependencies between genes, and the weight of an edge quantifies the statistical evidence of a specific (e.g. transcriptional) gene to gene interaction. The 4 different entropy estimators are empirical, Miller-Madow, Schurmann-Grassberger and shrink; and the 4 different inference methods are relevance networks, ARACNE, CLR, and MRNET. The package also integrates accuracy assessment tools, like F-scores, PR-curves, and ROC-curves in order to compare the inferred network with a reference one.

What is Mutual Information?

Mutual information is a non-linear measure of dependency. With mutual information generalized correlation networks (relevance networks) and conditional independence graphs (e.g. ARACNE) can be built. Mutual information calculates dependencies between two discrete random variables.

Datasets used:

The dataset is from an scRNA-seq experiment on induced pluripotent stem cells (iPSCs) in two-dimensional culture differentiating to hepatocyte-like cells. The datasets contain 425 scRNA-seq measurements from multiple time points: days 0 (iPSCs),6,8,14 and 21 (mature hepatocyte-like). Each row contains a microarray experiment/sample and each column contains a gene.

Microarray data

It obtains raw data in the form of a matrix. Hence, data are required to be pre-processed data to analyze similarities and differences of the genes expressed in the experiment. The microarray data is also known as gene expression profile. Gene expression means the amount of a gene expressed in given conditions. Usually the gene expression matrix is composed of rows and columns where genes are present in the row while the columns represent conditions such as an array, gene-chip, or experiments. Experiments usually correspond to timepoints for time-series dataset, and to conditions or treatments for steady-state dataset.

Four steps:

In order to infer a network with the *minet* package, 4 steps are required: data discretization, MIM computation, network inference, normalization of the network (optional)

The final step of the *minet* function is the normalization using the *norm(net)* function. This step normalizes all the weights of the inferred adjacency matrix between 0 and 1. Hence, the *minet* function

returns the inferred network as a weighted adjacency matrix with values ranging from 0 to 1 where the higher a weight, the higher is the evidence that a gene-gene interaction exists.

Details of the method:

Discretion Algorithms: In order to use the entropy estimators, continuous datasets must first be discretized. A number of algorithms exist to define the total number and boundaries of the resulting partitions (or bins). Common discretion methods are by equal width, global equal width, or equal frequency.

Implementation of Discretion methods:

discretize(dataset, disc="equalfreq", nbins=sqrt(nrow(dataset)))

where disc is a string which can take 3 values:"equalfreq","equalwidth","globalequalwidth", and nbins is the number of bins used for discretization.

Inference Methods:

1. **Relevance network**

This approach consists in inferring a genetic network where a pair of genes is linked by an edge if the mutual information is larger than a threshold.

2. **CLR Algorithm – Context Likelihood of Relatedness**

Computes the mutual information for each pair of genes and derives a score related to the empirical distribution of the MI values. Modifies the MI score based on the empirical distribution of all MI scores.

3. **ARACNE**---**The Algorithm for the Reconstruction of Accurate Cellular Networks**

Filters out indirect interactions from triplets of genes with the Data Processing Inequality. Starts by assigning to each pair of nodes a weight equal to the mutual information.

4. **MRNET – Minimum Redundancy Networks**

Infers a network using the maximum relevance/minimum redundancy (MRMR) feature selection method.

Estimators:

1. **The Empirical estimator**: also called the maximum likelihood estimator, is the entropy of the empirical distribution.
2. **The Scchurmann-Grassberger estimator**: the multivariate generalization of the beta distribution.
3. **The shrinkage estimator:** compromises between the observed frequencies, unbiased but with a high variance, and a prior (or target) distribution, biased but with low variance. The estimate is affected by both the choice of target distribution and the weight given to the target (or shrinkage intensity). It combines two different estimators by using a weighing factor.
4. **The Miller-Madow estimator**: the empirical entropy corrected by the asymptotic bias.

Implementation of estimators in *minet*:

The mutual information matrix is estimated by using the function *build.mim(dataset, estimator).* This function returns a matrix of paired mutual information and takes two arguments:

1. The generic dataset where columns contain variables/features and rows contain outcomes/samples

2. The string mi, that denotes the routine used to perform mutual information estimator: "mi.empirical","mi.shrink","mi.sg","mi.mm"

Network Inference Performance Metrics and Comparisons:

AUROC and AUPR curves are calculated by comparing the inferred networks (which assign a score to every potential network edge) with the true network used to simulate data and identifying the numbers of correctly assigned edges as the threshold for edge inclusion is varied.

AUROC is calculated from the area under the ROC curve, which is a plot of the false-positive rate (FPR) on the axis on the x-axis versus the true-positive rate (TPR) on the y-axis.

AUPR is the area under the curve for a plot of precision (y axis) versus recall (equal to TPR) on the x axis.

1) **True positives**: it means the predicted tool has an edge between genes as well as an edge present in the golden standard.

2) **True negatives**: it means the predicted tool has no edge between genes and no edge present in the golden standard.

3) **False positives**: it means the predicted tool has an edge between genes while no edge is present in the golden standard.

4) **False negatives**: it means the predicted tool has no edge between genes and there is an edge present between genes in the golden standard.

Assessment of the network inference algorithm

A network inference problem is a binary decision problem where the inference algorithm plays the role of a classifier: for each pair of nodes, the algorithm either returns an edge or not. Each pair of nodes can thus be assigned a positive label (an edge) or a negative one (no edge).

A positive label (an edge) predicted by the algorithm is considered as a true positive (TP)or as a false positive (FP) depending on the presence or not of the corresponding edge in the underlying true network.

A negative label is considered as a true negative (TN) or a false negative (FN)depending on whether the corresponding edge is present or not in the underlying true network.

All mutual information network inference methods use a threshold value in order to delete the arcs having a too low score.

1. **ROC curves**

The false positive rate: FPR= FP/(TN+FP)

The true positive rate: TPR= TP/(TP+FN)

A Receiver Operating Characteristic (ROC) curve is a plot of the TPR (true positive rate) vs FPR (false positive rate) for a binary classifier system as the threshold is varied.

2. **PR curves**

Let the precision quantity p=TP/(TP+FP) measure the fraction of real edges among the ones classified as positive and the recall quantity r=TP/(TP+FN), also known as true positive rate(TPR), denote the fraction of real edges that are correctly inferred.

The PR curve is a diagram which plots the precision (p) vs recall (r)for different values of the threshold.

3. **F-Scores**

A compact representation of the PR diagram is returned by the maximum and/or the average of the F-score quantity F=2pr/(r+p).

Assessment functionalities in *minet*

In order to benchmark the inference methods, the package provides a number of assessment tools. The *validate(net, ref.net, steps=50)* allows to compare an inferred network net to a reference network ref.net, described by a Boolean adjacency matrix. The assessment process consists in removing the inferred edges having a score below a given threshold and in computing the related confusion matrix, for steps thresholds ranging from the minimum to the maximum value of edge weights. A resulting data frame table containing the list of all the steps confusion matrices is returned.

pr(table) returns the related precisions and recalls, *rates(table)* computes true positive and false positive rates, while *fscores(table, beta)* returns the Fb-scores. show.pr(table) and *show.roc(table)* plots the PR-curves and ROC-curves

Things to note:

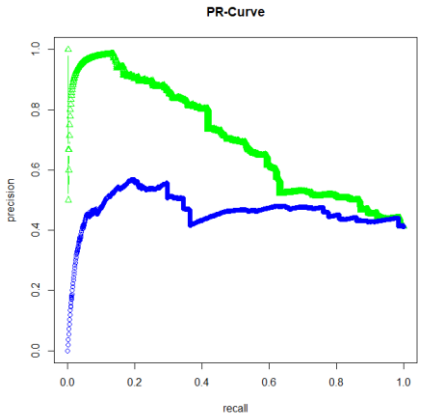
1. The *minet* package is no longer included in the R repository. To download, use the code provided in the R file.
2. Use similar code to download the Rgraphviz package.
3. A faster way of using the PIDC method is using JULIA, found here: https://github.com/Tchanders/NetworkInference.jl

References:

http://www.m2p-bioinfo.ups-tlse.fr/site/images/1/1a/Minet.vignette.pdf

https://link.springer.com/article/10.1186/1471-2105-9-461#Sec24

The Precision-Recall Curve



The ROC Curve