DAWN User Manual

Contents

1 Overview

This R implementation is based on the paper Network Assisted Analysis to Reveal the Genetic Basis of Autism (Li Liu, Jing Lei and Kathryn Roeder)[2].

DAWN (Detecting Association With Networks) is a statistical algorithm that uses a hidden Markov random field (HMRF) model to discover risk genes from gene co-expression network estimated by partial neighborhood selection (PNS) algorithm.

2 Usage

2.1 Download DAWN

2.1.1 Files

DAWN package contains 6 files: 2 R source files, and 4 sample input data files.

- **source_DAWN.R**: source file containing functions for PNS and HMRF analysis
- **run_DAWN.R**: R script for running complete DAWN analysis by calling functions defined in source_DAWN.R
- **sample_input_GLA.txt**: sample input for gene level association statistics, see details in section 2.2.1
- **sample_input_expression.txt**: sample gene expression input data, see details in section 2.2.2
- **sample_input_fixed_genes.txt**: sample input for genes to have fixed hidden states 2.2.3
- **sample_input_additional_covariates.txt**: sample input for additional covariates, se details in section 2.2.4

2.1.2 Package Dependences

DAWN requires the following pacakge(s):

- `glmnet`[1, 3]: used in HMRF analysis (`glmnet` requires package `Matrix` and `foreach`)

2.2 Prepare Inputs

2.2.1 Gene Level Association Statistics (GLA)

Gene level association statistics (GLA), such as TADA and Sherlock, should contain at least two parts: gene names, and p-values. GLA data should be processed into two vectors:

- `GLA_genenames`, a vector of gene names (one string for every gene). The gene names should be consistent with those used in gene expression data (using the same nomenclature, e.g. both use Ensembl IDs)
- `GLA_pvalue`, a vector of p-values (numeric). Its ordering should be consistent with GLA gene names.

2.2.2 Gene Expression Data

Gene expression data should contain the information about gene names, samples, and the expression value of each gene in each sample. Gene expression data should be processed into a data frame, `expdata`, where:

- each row represents a gene, and every column represents a sample;
- row and column names should be set to corresponding gene and sample names.

2.2.3 Fixed Genes

There may exist some genes that are proved to be risk genes for the disease of interest. In order to incorporate such previous knowledge, we provide the `fixedGene` variable.

`fixedGene` is a vector of gene names (one string for every gene, can be empty) whose hidden states you would like to set to 1 throughout the HMRF analysis.

Fixed genes can also be set automatically based on the magnitude of the `GLA_pvalue`. See `trimthresh` in 2.3.1 below.

The gene names should be consistent with those used in gene expression data (using the same nomenclature, e.g. both use Ensembl IDs).

To use fixed genes information, include `fixedGene = fixedGene` as a parameter when calling the function `identify_risk_genes()`.

2.2.4 Additional Covariates

DAWN can incorporate additional covariates to predict risk genes. Because the targets of one or more key transcription factors can be a meaningful covariate, we provide the `covGene` variable to take in this information.

`covGene` is a vector of gene names you believe that will affect the HMRF analysis. For example, gene names of targets of one or more transcription factors.

To use `covGene` information, include `covGene = covGene` as a parameter when calling the function `identify_risk_genes()`.

2.3 Quick Start Using `run_DAWN.R`

2.3.1 Steps

To perform a complete DAWN analysis using GLA and gene expression data, follow these steps:

1. open file `run_DAWN.R`
2. set your working directory to the directory of your DAWN code
3. load gene expression data
4. load GLA data
5. adjust parameters*
6. adjust input for function `identify_risk_genes()`**
7. save modifications on `run_DAWN.R` and source it using command `source('run_DAWN.R')`

*** Parameters:**

- `pthres_pns`: threshold for p-value in PNS algorithm, default = 0.1
- `corthres`: threshold for pairwise-correlation in PNS algorithm, default = 0.7
- `lambda`: tuning parameter for lasso, default = 0.24; lambda can affect the structure of PNS estimated graph, see 2.4.1 for more details.
- `pthres_hmrf`: threshold for p-value in HMRF analysis, default = 0.05
- `iter`: number of iterations in HMRF analysis, default = 100
- `trimthres`: threshold for Z score trimming, default = 5, where Z = normal score for the `GLA_pvalue`.
  To achieve robust estimates for the parameters in HMRF, we need to exclude some genes that have exceptionally high Z scores (or, low p-values). Also, these genes are very likely to be true risk genes according to the meaning of GLA p-value. Therefore, we provide a variable, `trimthres`, the trimming threshold for Z scores. For genes with $Z >$ `trimthres`, we fix their hidden states to 1 throughout the HMRF analysis so as to exclude them from the parameter estimation process.
  Please note, the union of genes with $Z >$ `trimthres` and genes in `fixedGene` are both treated as fixed genes in the function `DAWN_HMRF()` in the source file.
- `file_out_pns`: output file directory for estimated PNS network, default = 'pns.network'
- `file_out_hmrf`: output file directory for HMRF analysis results, default = 'DAWN_result.csv'

**** Call `identify_risk_genes()` function**

Please include `fixedGene= fixedGene` and/or `covGene= covGene` if you would like to include `fixedGene` and/or `covGene` information. For example, if you want to include both information, use

```
identify_risk_genes (GLA_pvalue = GLA_pvalue , GLA_genenames = GLA_genenames , expdata =
    expdata , covGene= covGene)
```

2.3.2 Sample code

```
##To directly run the code using sample input data
source('run_DAWN.R')
```

2.4 PNS Algorithm

2.4.1 Choose lambda

`lambda` is a important parameter in PNS algorithm, because its value directly affects the structure of estimated network (sparsity). As discussed in the paper, the network estimated by the PNS algorithm conforms the power law. Therefore, here we use the square of correlation $R^2 = (corr(logp(k), log(k)))^2$ to assess the goodness of esitmated network, where a larger R^2 means better a conformation to the power law. Therefore, the tuning parameter, `lambda` , can be chosen by visualizing the scatter plot of R^2 as a function of `lambda`.

To plot R^2 and choose a good `lambda`, please call:

```
    lambda_R2 <- choose_lambda(expdata=expdata, GLA_pvalue=GLA_pvalue, pthres_pns=pthres_pns, corthres=corthres)
```

Inputs:

- `expdata` - dataframe, expression data
- `GLA_pvalue` - dataframe, GLA p-values
- `pthres_pns` - threshold for p-value screening, default = 0.1
- `corthres` - threshold for pairwise-correlation, default = 0.7

Output:

- `lambda_R2` - a matrix where every row represents a `lambda` and `R`^{2} value pair

Then determine a reasonably good `lambda` for your PNS algorithm, and use this `lambda` in the future analyses.

2.4.2 Run PNS

If you have preprocessed your input, chosen a good lambda, and only want to run PNS algorithm, please call:

```
  graphres_PNS <- PNS_algorithm(expdata = expdata, GLA_pvalue = GLA_pvalue, pthres_pns = pthres_pns, corthres = corthres, lambda = lambda)
  graphfinal_PNS <- graphres_PNS$graphfinal  ##estimated network
  genelist_PNS <-  graphres_PNS$Gene  ##a list of genes in the estimated network
  rownames(graphfinal_PNS) <- genelist_PNS
  colnames(graphfinal_PNS) <- genelist_PNS
  write.csv(graphfinal_PNS, file_out_pns)
```

Inputs:

- `expdata` - dataframe, expression data
- `GLA_pvalue` - dataframe, GLA p-values
- `pthres_pns` - threshold for p-value screening, default = 0.1
- `corthres` - threshold for pairwise-correlation, default = 0.7
- `lambda` - tuning parameter for lasso, default = 0.24

Outputs:

- `graphres` - a list object representing the estimated PNS network
  - `graphres$Gene` - genes in the network
  - `graphres$graphfinal` - estimated unweighted, undirected network in matrix format
  - `graphres$pvfinal` - p-values of genes in the network

Please note, your expdata and GLA_pvalue should contain data for the same set of genes.

2.5 HMRF Analysis

If you have preprocessed your input, obtained a gene network (estimated by PNS or other algorithms), and only want to run HMRF analysis, please call:

```
## suppose you are using the graph estimated by PNS algorithm
result_DAWN <- DAWN_HMRF(pv = graphres_PNS$pvfinal, graph = graphres_PNS$graphfinal,
     covGene = covGene, fixedGene = fixedGene, pthres_hmrf = pthres_hmrf, iter = iter,
     trimthres = trimthres)
report <- result_report(finalposter = (1 - result_DAWN$post), genes = genelist_PNS, pv =
      graphres_PNS$pvfinal)  ## generate report
write.csv(report, file_out_hmrf)
```

Inputs:

- `pv` - vector, p-values of genes in the network
- `graph` - matrix, estimated network in matrix format
- `covGene` - vector, names of genes in the additional covariate of HMRF analysis
- `fixedGene` - vector, names of genes to have fixed hidden states 1 in HMRF analysis
- `pthres_hmrf` - threshold for p-values in HMRF initialization, default=0.05
- `iter` - number of iteration, default=100
- `trimthres` - threshold for Z score trimming, default=5

Outputs:

- `result_DAWN` - a list object representing the estimated statistics of genes
  - `result_DAWN$Iupdate` - updated I
  - `result_DAWN$post` - posterior distribution of I
  - `result_DAWN$b0` - estimated b
  - `result_DAWN$b1` - estimated c
  - `result_DAWN$b2` - estimated coefficient for additional covariate (default=0 if additional covariate is not included)
  - `result_DAWN$mu1` - estimated mean
  - `result_DAWN$sigmas` - estimated variance

Please note, your pv and graph should correspond to genes of the same ordering.

2.6 Error and Warning Message Handling

While running DAWN, you may encounter error or warning messages. Here are some explanations to these messages, and tips to handle them.

2.6.1 Error messages:

The program will stop and exit when an error occurs.

- ERROR: no common genes shared by GLA data and expression data, please check gene names again...

  This message means the intersect of genes in GLA and expression data is empty, then you will not be able to perform PNS algorithm where both GLA and expression data is needed for genes to be considered. If you see this error, please check if your two datasets contain genes in common. This error is also very possiblily caused by different naming systems you use in two datasets. Please note that shared genes are detected by directly comparing gene names, and thus is case sensitive.

- ERROR: cannot perform HMRF on empty PNS graph, please adjust your parameters.

  This message means the graph estimated by PNS is empty, and thus cannot be used in HMRF analysis. To deal with this, you may need to adjust parameters for the PNS algorithm.

2.6.2 Warning messages:

The program will keep running and return results when a warning is thrown out. However, you should be very careful about the reliability of the returned results.

- WARNING: DAWN identified a large number of risk genes. Assumptions of the model may be false. The set of risk genes likely contains many false positives.

  This message means the estimated intercept, `b0`, is too large, indicating DAWN will identify a large number of risk genes. This tends to occur when the genes are not clustered in the graph and the algorithm fails to find an acceptable solution. In this case, it might help to re-estimate your graph using different parameters.

- WARNING: Weak connectivity among risk genes in the input graph. Assumptions of the model appear to be false. The set of risk genes likely contains many false positives.

  This message means the estimated slope parameter, `b1`, is near 0, indicating the risk genes are not clustered in the graph, resulting in a high false positive rate. In this case, it might help to re-estimate your graph using different parameters.

3 Version Information

3.1 DAWN

- DAWN v1.0: Dec 09, 2015

3.2 Manual

- Manual v1.0: Dec 09, 2015

References

[1] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

[2] Li Liu, Jing Lei, and Kathryn Roeder. Network assisted analysis to reveal the genetic basis of autism. *Ann. Appl. Stat.*, 9(3):1571–1600, 09 2015.

[3] Noah Simon, Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for cox's proportional hazards model via coordinate descent. *Journal of Statistical Software*, 39(5):1–13, 2011.