MRFscRNAseq Vignette

Hongyu Li

2021-01-14

Contents

Installing MRFscRNAseq from GitHub

```
if (!requireNamespace("devtools", quietly = TRUE))
    install.packages("devtools")
library(devtools)
install_github("eddiehli/MRFscRNAseq", quiet = TRUE)
library(MRFscRNAseq)
```

Main Function

The main function is

```
?get_DE_MRF()
```

It requires three input values: (1) data: a summary statistics matrix where rows are genes and columns are cell types; (2) g_g: the gene to gene binary network matrix; and (3) c_c: the cell type dependency binary network matrix. The usage of this function will be illustrated on a simulated data set.

Example

The gg_info data is a list that contains (1) a gene to gene binary network matrix; (2) all gene names in the network matrix; (3) genes that are assumed to be DE in the network matrix.

```
gg_info = MRFscRNAseq::gg_info
gg_info$g_g[100:110, 100:110]
#>        OAS1 OAS2 OAS3 RNASEL NLRP3 PYCARD CASP1 IL1B IL18 IL33 NLRC4
#> OAS1      0    1    1      1     0      0     0    0    0    0     0
#> OAS2      1    0    1      1     0      0     0    0    0    0     0
#> OAS3      1    1    0      1     0      0     0    0    0    0     0
#> RNASEL    1    1    1      0     0      0     0    0    0    0     0
#> NLRP3     0    0    0      0     0      1     1    1    1    0     0
#> PYCARD    0    0    0      0     1      0     1    1    1    0     1
#> CASP1     0    0    0      0     1      1     0    1    1    1     1
#> IL1B      0    0    0      0     1      1     1    0    1    1     0
#> IL18      0    0    0      0     1      1     1    1    0    1     0
#> IL33      0    0    0      0     0      0     1    1    1    0     0
#> NLRC4     0    0    0      0     0      1     1    0    0    0     0
```

We use this information to simulate a data set. The number of cell types was set to be 20. We set those genes that are assumed to be DE in the first 10 cell types to be DE. The rest were set to be equally expressed. Then the counts were simulated from a Negative Binomial distribution with mean $\alpha = 2$, $\beta = 4$ and dispersion $\phi = 0.1$, and $\tau = 5$.

```
data = MRFscRNAseq::get_sim_data(gg_info,
                                 gamma_alpha = 2, gamma_beta = 4, tau = 5,
                                 random.seed = 2020)
```

First, we get p-values for the simulated data set using `ttest`, and obtain sensitivity, specificity and fdr for the results using significance level at $\alpha = 0.01$,

```
results = MRFscRNAseq::get_DE_noMRF(data, test_type = "ttest")
MRFscRNAseq::get_sens_spec_fdr(data, test_type = "ttest", results)
#> sensitivity specificity         fdr
#> 0.503273322 0.999723283 0.008054539
```

Then we use our proposed MRF models to obtain DE results, we first covert the raw expression values to z scores using two-sample t-tests,

```
zz = MRFscRNAseq::get_z_scores(data)
g_g = data$g_g
c_c = data$c_c
```

Then we ran the MRF model. Note that this step might require high performance computing,

```
results = get_DE_MRF(zz, g_g, c_c)
```

After we have the results from the MRF procedure, we can evaluate it by sensitivity, specificity and fdr

```
MRFscRNAseq::get_sens_spec_fdr(data, test_type = "MRF", results)
```