miRNAtap example use

Maciej Pajak, Ian Simpson

October 30, 2018

Contents

1 Introduction **2**

2 Installation **2**

3 Workflow **3**

4 Session Information **5**

References **6**

1 Introduction

`miRNAtap` package is designed to facilitate implementation of workflows requiring miRNA prediction. Aggregation of commonly used prediction algorithm outputs in a way that improves on performance of every single one of them on their own when compared against experimentally derived targets. microRNA (miRNA) is a 18-22nt long single strand that binds with RISC (RNA induced silencing complex) and targets mRNAs effectively reducing their translation rates.

Targets are aggregated from 5 most commonly cited prediction algorithms: DIANA (Maragkakis et al., 2011), Miranda (Enright et al., 2003), PicTar (Lall et al., 2006), TargetScan (Friedman et al., 2009), and miRDB (Wong and Wang, 2015).

Programmatic access to sources of data is crucial when streamlining the workflow of our analysis, this way we can run similar analysis for multiple input miRNAs or any other parameters. Not only does it allow us to obtain predictions from multiple sources straight into R but also through aggregation of sources it improves the quality of predictions.

Finally, although direct predictions from all sources are only available for *Homo sapiens* and *Mus musculus*, this package includes an algorithm that allows to translate target genes to other speices (currently only *Rattus norvegicus*) using homology information where direct targets are not available.

2 Installation

This section briefly describes the necessary steps to get `miRNAtap` running on your system. We assume that the user has the R program (see the R project at http://www.r-project.org) already installed and is familiar with it. You will need to have R 3.2.0 or later to be able to install and run `miRNAtap`. The miRNAtap package is available from the Bioconductor repository at http://www.bioconductor.org To be able to install the package one needs first to install the core Bioconductor packages. If you have already installed Bioconductor packages on your system then you can skip the two lines below.

```
> if (!requireNamespace("BiocManager", quietly=TRUE))
+     install.packages("BiocManager")
> BiocManager::install()
```

Once the core Bioconductor packages are installed, we can install the miRNAtap and accompanying database `miRNAtap.db` package by

```
> if (!requireNamespace("BiocManager", quietly=TRUE))
+     install.packages("BiocManager")
> BiocManager::install("miRNAtap")
> BiocManager::install("miRNAtap.db")
```

3 Workflow

This section explains how `miRNAtap` package can be integrated in the workflow aimed at predicting which processes can be regulated by a given microRNA.

In this example workflow we'll use `miRNAtap` as well as another Bioconductor package `topGO` together with Gene Ontology (GO) annotations. In case we don't have `topGO` or GO annotations on our machine we need to install them first:

```
> if (!requireNamespace("BiocManager", quietly=TRUE))
+     install.packages("BiocManager")
> BiocManager::install("topGO")
> BiocManager::install("org.Hs.eg.db")
```

Then, let's load the required libraries

```
> library(miRNAtap)
> library(topGO)
> library(org.Hs.eg.db)
```

Now we can start the analysis. First, we will obtain predicted targets for human miRNA *miR-10b*

```
> mir = 'miR-10b'
> predictions = getPredictedTargets(mir, species = 'hsa',
+                                      method = 'geom', min_src = 2)
```

Let's inspect the top of the prediction list.

```
> head(predictions)
```

```
       source_1 source_2 source_3 source_4 source_5 rank_product rank_final
627         103     10.0      1.0       NA        1     1.416281          1
79741        NA       NA      8.0        2       NA     2.000000          2
6095          5      2.5     73.5       NA        5     2.058173          3
348980       NA      2.5     20.0       NA       NA     3.535534          4
51365        NA     53.0      3.0       12       27     3.766392          5
7022         88     17.5      5.0      149        3     4.058725          6
```

We are using *geometric mean* aggregation method as it proves to perform best when tested against experimental data from MirBase (Griffiths-Jones et al., 2008).

We can compare it to the top of the list of the output of *mimumum* method:

```
> predictions_min = getPredictedTargets(mir, species = 'hsa',
+                                      method = 'min', min_src = 2)
> head(predictions_min)
```

```
      source_1 source_2 source_3 source_4 source_5 rank_product rank_final
627        103       10      1.0       NA        1            1        2.0
8897         1      183    282.0       NA       NA            1        2.0
79042       NA      107     99.5        1       NA            1        2.0
7182         2       NA       NA       NA      106            2        5.5
10739       NA       42      2.0       NA       NA            2        5.5
79741       NA       NA      8.0        2       NA            2        5.5
```

Where predictions for rat genes are not available we can obtain predictions for mouse genes and translate them into rat genes through homology. The operation happens automatically if we specify species as `rno` (for *Rattus norvegicus*)

```
> predictions_rat = getPredictedTargets(mir, species = 'rno',
+                                       method = 'geom', min_src = 2)
```

Now we can use the ranked results as input to GO enrichment analysis. For that we will use our initial prediction for human *miR-10b*

```
> rankedGenes = predictions[,'rank_product']
> selection = function(x) TRUE
> # we do not want to impose a cut off, instead we are using rank information
> allGO2genes = annFUN.org(whichOnto='BP', feasibleGenes = NULL,
+                 mapping="org.Hs.eg.db", ID = "entrez")
> GOdata =  new('topGOdata', ontology = 'BP', allGenes = rankedGenes,
+              annot = annFUN.GO2genes, GO2genes = allGO2genes,
+              geneSel = selection, nodeSize=10)
```

In order to make use of the rank information we will use Kolomonogorov Smirnov (K-S) test instead of Fisher exact test which is based only on counts.

```
> results.ks = runTest(GOdata, algorithm = "classic", statistic = "ks")
> results.ks

Description:
Ontology: BP
'classic' algorithm with the 'ks' test
630 GO terms scored: 5 terms with p < 0.01
Annotation data:
    Annotated genes: 346
    Significant genes: 346
    Min. no. of genes annotated to a GO: 10
    Nontrivial nodes: 630
```

We can view the most enriched GO terms (and potentially feed them to further steps in our workflow)

```
> allRes = GenTable(GOdata, KS = results.ks, orderBy = "KS", topNodes = 20)
> allRes[,c('GO.ID','Term','KS')]
```

```
        GO.ID                                        Term     KS
1  GO:0050789          regulation of biological process 0.0015
2  GO:0065007                     biological regulation 0.0038
3  GO:0044087 regulation of cellular component biogene... 0.0061
4  GO:0042692              muscle cell differentiation 0.0067
5  GO:0048518 positive regulation of biological proces... 0.0095
6  GO:0043254     regulation of protein complex assembly 0.0112
7  GO:0050794            regulation of cellular process 0.0116
8  GO:0044089 positive regulation of cellular componen... 0.0173
9  GO:0061061               muscle structure development 0.0211
10 GO:0051146       striated muscle cell differentiation 0.0215
11 GO:0006352     DNA-templated transcription, initiation 0.0222
12 GO:1902680 positive regulation of RNA biosynthetic ... 0.0253
13 GO:1903508 positive regulation of nucleic acid-temp... 0.0253
14 GO:0014070         response to organic cyclic compound 0.0333
15 GO:0016032                             viral process 0.0344
16 GO:0008219                                cell death 0.0351
17 GO:0006367 transcription initiation from RNA polyme... 0.0382
18 GO:0043547     positive regulation of GTPase activity 0.0390
19 GO:0006915                         apoptotic process 0.0417
20 GO:0012501                     programmed cell death 0.0417
```

For more details about GO analysis refer to topGO package vignette (Alexa and Rahnenfuhrer, 2010).

Finally, we can use our predictions in a similar way for pathway enrichment analysis based on KEGG (Kanehisa and Goto, 2000), for example using Bioconductor's KEGGprofile (Zhao, 2012).

4 Session Information

- R version 3.5.1 Patched (2018-07-12 r74967), x86_64-apple-darwin15.6.0
- Locale: C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
- Running under: OS X El Capitan 10.11.6
- Matrix products: default
- BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
- LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils

- Other packages: AnnotationDbi 1.44.0, Biobase 2.42.0, BiocGenerics 0.28.0, GO.db 3.7.0, IRanges 2.16.0, S4Vectors 0.20.0, SparseM 1.77, graph 1.60.0, miRNAtap 1.16.0, miRNAtap.db 0.99.10, org.Hs.eg.db 3.7.0, topGO 2.34.0
- Loaded via a namespace (and not attached): DBI 1.0.0, RSQLite 2.1.1, Rcpp 0.12.19, bit 1.1-14, bit64 0.9-7, blob 1.1.1, chron 2.3-53, compiler 3.5.1, digest 0.6.18, grid 3.5.1, gsubfn 0.7, lattice 0.20-35, magrittr 1.5, matrixStats 0.54.0, memoise 1.1.0, pkgconfig 2.0.2, plyr 1.8.4, proto 1.0.0, sqldf 0.4-11, stringi 1.2.4, stringr 1.3.1, tools 3.5.1

References

Alexa, A. and Rahnenfuhrer, J. (2010). *topGO: topGO: Enrichment analysis for Gene Ontology*. R package version 2.16.0.

Enright, A. J., John, B., Gaul, U., Tuschl, T., Sander, C., and Marks, D. S. (2003). MicroRNA targets in Drosophila. *Genome biology*, 5(1):R1.

Friedman, R. C., Farh, K. K.-H., Burge, C. B., and Bartel, D. P. (2009). Most mammalian mRNAs are conserved targets of microRNAs. *Genome research*, 19(1):92–105.

Griffiths-Jones, S., Saini, H. K., van Dongen, S., and Enright, A. J. (2008). miRBase: tools for microRNA genomics. *Nucleic acids research*, 36(Database issue):D154–8.

Kanehisa, M. and Goto, S. (2000). KEGG: kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1):27–30.

Lall, S., Grün, D., Krek, A., Chen, K., Wang, Y.-L., Dewey, C. N., Sood, P., Colombo, T., Bray, N., Macmenamin, P., Kao, H.-L., Gunsalus, K. C., Pachter, L., Piano, F., and Rajewsky, N. (2006). A genome-wide map of conserved microRNA targets in C. elegans. *Current biology : CB*, 16(5):460–71.

Maragkakis, M., Vergoulis, T., Alexiou, P., Reczko, M., Plomaritou, K., Gousis, M., Kourtis, K., Koziris, N., Dalamagas, T., and Hatzigeorgiou, A. G. (2011). DIANA-microT Web server upgrade supports Fly and Worm miRNA target prediction and bibliographic miRNA to disease association. *Nucleic acids research*, 39(Web Server issue):W145–8.

Wong, N. and Wang, X. (2015). miRDB: An online resource for microRNA target prediction and functional annotations. *Nucleic Acids Research*, 43(D1):D146–D152.

Zhao, S. (2012). *KEGGprofile: An annotation and visualization package for multi-types and multi-groups expression data in KEGG pathway*. R package version 1.6.1.