miRNAtap example use

Maciej Pajak, Ian Simpson

October 13, 2015

Contents

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 4 most commonly cited prediction algorithms: DIANA (Maragkakis et al., 2011), Miranda (Enright et al., 2003), PicTar (Lall et al., 2006) and TargetScan (Friedman et al., 2009).

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.

```
> source("http://bioconductor.org/biocLite.R")
> biocLite()
```

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

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("miRNAtap")
> biocLite("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:

```
> source("http://bioconductor.org/biocLite.R")
> biocLite("topGO")
> biocLite("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 rank_product rank_final
133923       NA     23.5       NA        1     2.423840          1
7707          1    102.5       NA       NA     5.062114          2
627         209     21.5        1       NA     5.500561          3
27253        61      2.0       NA       NA     5.522681          4
6095         29     10.0       16       NA     5.559701          5
64641        48     27.0        5       NA     6.214465          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 rank_product rank_final
627         209     21.5        1       NA            1        2.5
7707          1    102.5       NA       NA            1        2.5
8013        108      1.0       87       NA            1        2.5
133923       NA     23.5       NA        1            1        2.5
7022        182    376.0        2      192            2        6.0
10152         2    368.5       NA       70            2        6.0
```

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")

                        -- Classic Algorithm --

                the algorithm is scoring 875 nontrivial nodes
                parameters:
                        test statistic:  ks
                        score order:  increasing

> results.ks

Description:
Ontology: BP
'classic' algorithm with the 'ks' test
875 GO terms scored: 16 terms with p < 0.01
Annotation data:
    Annotated genes: 452
    Significant genes: 452
    Min. no. of genes annotated to a GO: 10
    Nontrivial nodes: 875
```

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:0006351                transcription, DNA-templated 0.0025
2  GO:0097659        nucleic acid-templated transcription 0.0025
3  GO:0006355 regulation of transcription, DNA-templat... 0.0034
4  GO:1903506 regulation of nucleic acid-templated tra... 0.0034
5  GO:2001141      regulation of RNA biosynthetic process 0.0034
6  GO:0016070                         RNA metabolic process 0.0037
7  GO:0032774                       RNA biosynthetic process 0.0037
8  GO:0006366 transcription from RNA polymerase II pro... 0.0043
9  GO:0043254      regulation of protein complex assembly 0.0055
10 GO:0090304               nucleic acid metabolic process 0.0056
11 GO:0018130             heterocycle biosynthetic process 0.0058
12 GO:0034654 nucleobase-containing compound biosynthe... 0.0058
13 GO:0006357 regulation of transcription from RNA pol... 0.0072
14 GO:0051252         regulation of RNA metabolic process 0.0074
15 GO:0006974    cellular response to DNA damage stimulus 0.0076
16 GO:0019438      aromatic compound biosynthetic process 0.0080
17 GO:0006139 nucleobase-containing compound metabolic... 0.0105
18 GO:1901362 organic cyclic compound biosynthetic pro... 0.0109
19 GO:0008406                            gonad development 0.0119
20 GO:0045137 development of primary sexual characteri... 0.0119
```

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 3.2.2 (2015-08-14), `x86_64-pc-linux-gnu`
- Locale: `LC_CTYPE=en_US.UTF-8`, `LC_NUMERIC=C`, `LC_TIME=en_US.UTF-8`, `LC_COLLATE=C`, `LC_MONETARY=en_US.UTF-8`, `LC_MESSAGES=en_US.UTF-8`, `LC_PAPER=en_US.UTF-8`, `LC_NAME=C`, `LC_ADDRESS=C`, `LC_TELEPHONE=C`, `LC_MEASUREMENT=en_US.UTF-8`, `LC_IDENTIFICATION=C`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: AnnotationDbi 1.32.0, Biobase 2.30.0, BiocGenerics 0.16.0, DBI 0.3.1, GO.db 3.2.2, IRanges 2.4.0, RSQLite 1.0.0, S4Vectors 0.8.0, SparseM 1.7, graph 1.48.0, miRNAtap 1.4.0, miRNAtap.db 0.99.7, org.Hs.eg.db 3.2.3, topGO 2.22.0
- Loaded via a namespace (and not attached): Rcpp 0.12.1, chron 2.3-47, grid 3.2.2, gsubfn 0.6-6, lattice 0.20-33, magrittr 1.5, plyr 1.8.3, proto 0.3-10, sqldf 0.4-10, stringi 0.5-5, stringr 1.0.0, tools 3.2.2

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.

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.