Processed human microRNA-overexpression data from GEO, and sequence information from TargetScan, and targetScore from TargetScore

Yue Li
yueli@cs.toronto.edu
October 30, 2021

1 MicroRNA perturbation datasets

We collected 84 Gene Expression Omnibus (GEO) series corresponding to 6 platforms, 77 human cells or tissues, and 112 distinct miRNAs. To our knowledge, this is by far the largest miRNA-overexpression data compendium. To automate the data download and processing, we developed a pipeline written in R, making use of the function getGEO from GEOquery R/Bioconductor package (Davis and Meltzer [2007]). For each dataset, the pipeline downloads the raw or processed data (if available) and calculates (when necessary) the log fold-change (logFC) in treatment (miRNA transfected) vs (mock) control, taking into account the unique properties of each data. Next, we combined all of the logFC data columns into a single \( N \times M \) matrix for all of the \( N = 19177 \) RefSeq mRNAs (NM_* obtained from UCSC) and \( M = 286 \) datasets. Missing data (logFC) for some genes across studies were imputed using impute.knn from impute R package (Troyanskaya et al. [2001]). For miRNA transfection data having multiple measurements (in different studies), we picked the one whose logFC correlate the most with the validated targets from mirTarBase Hsu et al. [2011] or average them if no validated target available.

```r
> library(TargetScoreData)
> transfection_data <- get_miRNA_transfection_data()$transfection_data
> datasummary <-
+ list(`MicroRNA` = table(names(transfection_data)),
+ `GEO Series` = table(sapply(transfection_data, function(df) df$Series[1]),
+ `Platform` = table(sapply(transfection_data, function(df) df$platform[1]),
+ `Cell/Tissue` = table(sapply(transfection_data, function(df) df$cell[1]))
> print(lapply(datasummary, length))

$MicroRNA
[1] 113

$`GEO Series`
```
2 TargetScan context score and PCT

TargetScan context score and PCT for all of the predicted sites (including conserved and nonconserved sites) downloaded from TargetScan website (http://www.targetscan.org/cgi-bin/targetscan/data_download.cgi?db=vert_61)

> targetScanCS <- get_TargetScanHuman_contextScore()
> targetScanPCT <- get_TargetScanHuman_PCT()
> head(targetScanCS)

<table>
<thead>
<tr>
<th>Gene Symbol</th>
<th>Transcript ID</th>
<th>miRNA 3'prime pairing local AU position</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1CF</td>
<td>NM_138932</td>
<td>hsa-miR-4711-3p -0.018 -0.095 -0.108</td>
</tr>
<tr>
<td>A1CF</td>
<td>NM_138933</td>
<td>hsa-miR-4711-3p -0.018 -0.095 -0.108</td>
</tr>
<tr>
<td>A1CF</td>
<td>NM_014576</td>
<td>hsa-miR-4711-3p -0.018 -0.095 -0.108</td>
</tr>
<tr>
<td>A1CF</td>
<td>NM_001198820</td>
<td>hsa-miR-4711-3p -0.018 -0.095 -0.108</td>
</tr>
<tr>
<td>A1CF</td>
<td>NM_001198819</td>
<td>hsa-miR-4711-3p -0.018 -0.095 -0.108</td>
</tr>
<tr>
<td>A1CF</td>
<td>NM_001198818</td>
<td>hsa-miR-4711-3p -0.018 -0.095 -0.108</td>
</tr>
</tbody>
</table>

TA SPS context+ score context+ score percentile

| 1 | 0.003 0.017 | -0.448 99 |
| 2 | 0.003 0.017 | -0.448 99 |
| 3 | 0.003 0.017 | -0.448 99 |
| 4 | 0.003 0.017 | -0.448 99 |
| 5 | 0.003 0.017 | -0.448 99 |
| 6 | 0.003 0.017 | -0.448 99 |

> dim(targetScanCS)

[1] 9569357 10

> head(targetScanPCT)

<table>
<thead>
<tr>
<th>miR Family</th>
<th>Gene Symbol</th>
<th>Transcript ID</th>
<th>PCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>miR-22/22-3p</td>
<td>A1BG</td>
<td>NM_130786</td>
<td>0.00</td>
</tr>
<tr>
<td>miR-23abc/23b-3p</td>
<td>A1BG</td>
<td>NM_130786</td>
<td>0.00</td>
</tr>
<tr>
<td>miR-26ab/1297/4465</td>
<td>A1BG</td>
<td>NM_130786</td>
<td>0.00</td>
</tr>
<tr>
<td>miR-101/101ab</td>
<td>A1BG</td>
<td>NM_130786</td>
<td>0.00</td>
</tr>
<tr>
<td>miR-103a/107/107ab</td>
<td>A1BG</td>
<td>NM_130786</td>
<td>0.00</td>
</tr>
<tr>
<td>miR-103a/107/107ab</td>
<td>A1BG</td>
<td>NM_130786</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Encouraged by the superior performance of TargetScore (manuscript in peer-review), we applied TargetScore to all of the transfection data above. For further exploring miRNA targetome and their associations, we enclose the targetScores results in this package.

```r
> targetScoreMatrix <- get_precomputed_targetScores()
> head(names(targetScoreMatrix))
[1] "hsa-miR-34b" "hsa-miR-34c" "hsa-miR-205" "hsa-miR-124" "hsa-miR-1"
[6] "hsa-miR-181a"
> head(targetScoreMatrix[[1]])

<table>
<thead>
<tr>
<th></th>
<th>logFC</th>
<th>targetScanCS</th>
<th>targetScanPCT</th>
<th>targetScore</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGIP1</td>
<td>0.077526011</td>
<td>0.00</td>
<td>0</td>
<td>0.03489650</td>
</tr>
<tr>
<td>AGBL4</td>
<td>0.020639084</td>
<td>0.00</td>
<td>0</td>
<td>0.03388637</td>
</tr>
<tr>
<td>NECAP2</td>
<td>0.078650400</td>
<td>0.00</td>
<td>0</td>
<td>0.03492518</td>
</tr>
<tr>
<td>CLIC4</td>
<td>0.016043400</td>
<td>-0.03</td>
<td>0</td>
<td>0.24335149</td>
</tr>
<tr>
<td>ADC</td>
<td>-0.002303429</td>
<td>0.00</td>
<td>0</td>
<td>0.03417828</td>
</tr>
<tr>
<td>SLC45A1</td>
<td>-0.018655797</td>
<td>0.00</td>
<td>0</td>
<td>0.03457975</td>
</tr>
</tbody>
</table>
```

We can reproduce targetScores using the above data as demonstrated in the following example (require TargetScore package). As a convenience function, we applied a wrapper function called getTargetScores that does the following: (1) given a miRNA ID, obtain fold-change(s) from logFC.imputed matrix or use the user-supplied fold-changes; (2) retrieves TargetScan context score (CS) and PCT (if found); (3) obtain validated targets from the local mirTarBase file; (4) compute targetScore. We apply getTargetScores function using miRNA hsa-miR-1, which we know has all three types of data, namely logFC, targetScan context score, and PCT.

```r
> library(TargetScore)
> library(gplots)
> myTargetScores <- getTargetScores("hsa-miR-1", tol=1e-3, maxiter=200)
> table((myTargetScores$targetScore > 0.1), myTargetScores$validated)  # a very lenient cutoff
# obtain all of targetScore for all of the 112 miRNA

> logFC.imputed <- get_precomputed_logFC()
> mirIDs <- unique(colnames(logFC.imputed))
#
# takes time
>
# targetScoreMatrix <- mclapply(mirIDs, getTargetScores)
>
# names(targetScoreMatrix) <- mirIDs

3
4 Session Info

> sessionInfo()

R version 4.1.1 (2021-08-10)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 20.04.3 LTS

Matrix products: default
BLAS: /home/biocbuild/bbs-3.14-bioc/R/lib/libRblas.so
LAPACK: /home/biocbuild/bbs-3.14-bioc/R/lib/libRlapack.so

locale:
[1] LC_CTYPE=en_US.UTF-8   LC_NUMERIC=C
[3] LC_TIME=en_GB          LC_COLLATE=C
[5] LC_MONETARY=en_US.UTF-8 LC_MESSAGES=en_US.UTF-8
[7] LC_PAPER=en_US.UTF-8   LC_NAME=C
[9] LC_ADDRESS=C           LC_TELEPHONE=C

attached base packages:
[1] stats     graphics   grDevices  utils     datasets   methods   base

other attached packages:
[1] TargetScoreData_1.30.0

loaded via a namespace (and not attached):
[1] compiler_4.1.1 tools_4.1.1

References


Sheng-Da Hsu, Feng-Mao Lin, Wei-Yun Wu, Chao Liang, Wei-Chih Huang, Wen-Ling Chan, Wen-Ting Tsai, Goun-Zhou Chen, Chia-Jung Lee, Chih-Min Chiu, Chia-Hung Chien, Ming-Chia Wu, Chi-Ying Huang, Ann-Ping Tsou, and Hsien-Da Huang. miRTarBase: a database curates experimentally validated microRNA-target interactions. Nucleic acids research, 39 (Database issue):D163–9, January 2011.