The multiMiR user's guide

Yuanbin Ru,Matt Mulvahill,Spencer Mahaffey, and Katerina Kechris

Package version: multiMiR

Contents

1 Introduction 1

2 Getting to know the multiMiR database 2

3 List miRNAs, genes, drugs and diseases in the multiMiR database 5

4 Use `get.multimir()` to query the multiMiR database 6

5 Examples of multiMiR queries 6

5.1 Example 1: Retrieve all validated target genes of a given miRNA 6
5.2 Example 2: Retrieve miRNA-target interactions associated with a given drug or disease 7
5.3 Example 3: Select miRNAs predicted to target a gene 8
5.4 Example 4: Select miRNA(s) predicted to target most, if not all, of the genes of interest 9
5.5 Example 5: Retrieve interactions between a set of miRNAs and a set of genes 10

6 Direct query to the database on the multiMiR web server 11

6.1 Direct query on the web server 11
6.2 Direct query in R 12

7 Session Info 12

1 Introduction

microRNAs (miRNAs) regulate expression by promoting degradation or repressing translation of target transcripts. miRNA target sites have been cataloged in databases based on experimental validation and computational prediction using a variety of algorithms. Several online resources provide collections of multiple databases but need to be imported into other software, such as R, for processing, tabulation, graphing and computation. Currently available miRNA target site packages in R are limited in the number of databases, types of databases and flexibility.

The R package *multiMiR*, with web server at http://multimir.ucdenver.edu, is a comprehensive collection of predicted and validated miRNA-target interactions and their associations with diseases and drugs. *multiMiR* includes several novel features not available in existing R packages:

1. Compilation of 14 different databases, more than any other collection
2. Expansion of databases to those based on disease annotation and drug response, in addition to many experimental and computational databases
3. User-defined cutoffs for predicted binding strength to provide the most confident selection.

The *multiMiR* package enables retrieval of miRNA-target interactions from 14 external databases in R without the need to visit all these databases. Advanced users can also submit SQL queries to the web server to retrieve results. See the publication on PubMed for additional detail on the database and its creation. The database is now versioned so it is possible to use previous versions of databases from the current R package, however the package defaults to the most recent version.

2 Getting to know the multiMiR database

The *multiMiR* web server http://multimir.ucdenver.edu hosts a database containing miRNA-target interactions from external databases. The package *multiMiR* provides functions to communicate with the *multiMiR* web server and its database. The *multiMiR* database is now versioned. By default *multiMiR* will use the most recent version each time *multiMiR* is loaded. However it is now possible to switch between database versions and get information about the *multiMiR* database versions. `multimir_dbInfoVersions()` returns a dataframe with the available versions.

```
library(multiMiR)
## Welcome to multiMiR.
##
## multiMiR database URL has been set to the
## default value: http://multimir.ucdenver.edu/
##
## Database Version: 2.1.0  Updated: 2016-12-22
db.ver = multimir_dbInfoVersions()
db.ver
##   VERSION    UPDATED                      RDA      DBNAME                 SCHEMA PUBLIC
## 1   2.1.0 2016-12-22 multimir_cutoffs_2.1.rda multimir2_1 multiMiR_DB_schema.sql      1
## 2   2.0.0 2015-05-01     multimir_cutoffs.rda    multimir multiMiR_DB_schema.sql      1
##                  TABLES
## 1 multiMiR_dbTables.txt
## 2 multiMiR_dbTables.txt
```

To switch between versions we can use `multimir_switchDBVersion()`.

```
multimir_switchDBVersion(db_version = "2.0.0")
## Now using database version: 2.0.0
multimir_switchDBVersion(db_version = "2.1.0")
## Now using database version: 2.1.0
```

The remaining functions will query the selected version until the package is reloaded or until we switch to another version.

Information from each external database is stored in a table in the *multiMiR* database. To see a list of the tables, we can use the `multimir_dbTables()` function.

```
db.tables = multimir_dbTables()
db.tables
##  [1] "diana_microt" "elmmo"        "map_counts"   "map_metadata" "microcosm"    "mir2disease"
##  [7] "miranda"      "mirdb"        "mirecords"    "mirna"        "mirtarbase"   "pharmaco_mir"
## [13] "phenomir"     "pictar"       "pita"         "tarbase"      "target"       "targetscan"
```

To display the database schema, we can use the `multimir_dbSchema()` function. Following is only a portion of the full output.

```
## --
## -- Table structure for table `mirna`
## --
##
## DROP TABLE IF EXISTS `mirna`;
## CREATE TABLE `mirna` (
##   mature_mirna_uid INTEGER UNSIGNED AUTO_INCREMENT,  -- mature miRNA unique ID
##   org VARCHAR(4) NOT NULL,                           -- organism abbreviation
##   mature_mirna_acc VARCHAR(20) default NULL,         -- mature miRNA accession
##   mature_mirna_id VARCHAR(20) default NULL,          -- mature miRNA ID/name
##   PRIMARY KEY (mature_mirna_uid),
##   KEY org (org),
##   KEY mature_mirna_acc (mature_mirna_acc),
##   KEY mature_mirna_id (mature_mirna_id)
## );
##
## --
## -- Table structure for table `target`
## --
##
## DROP TABLE IF EXISTS `target`;
## CREATE TABLE `target` (
##   target_uid INTEGER UNSIGNED AUTO_INCREMENT,   -- target gene unique ID
##   org VARCHAR(4) NOT NULL,                      -- organism abbreviation
##   target_symbol VARCHAR(80) default NULL,       -- target gene symbol
##   target_entrez VARCHAR(10) default NULL,       -- target gene Entrez gene ID
##   target_ensembl VARCHAR(20) default NULL,      -- target gene Ensembl gene ID
##   PRIMARY KEY (target_uid),
##   KEY org (org),
##   KEY target_symbol (target_symbol),
##   KEY target_entrez (target_entrez),
##   KEY target_ensembl (target_ensembl)
## );
##
## --
## -- Table structure for table `mirecords`
## --
##
## DROP TABLE IF EXISTS `mirecords`;
## CREATE TABLE `mirecords` (
##   mature_mirna_uid INTEGER UNSIGNED NOT NULL,   -- mature miRNA unique ID
##   target_uid INTEGER UNSIGNED NOT NULL,         -- target gene unique ID
##   target_site_number INT(10) default NULL,      -- target site number
##   target_site_position INT(10) default NULL,    -- target site position
##   experiment VARCHAR(160) default NULL,         -- supporting experiment
##   support_type VARCHAR(40) default NULL,        -- type of supporting experiment
##   pubmed_id VARCHAR(10) default NULL,           -- PubMed ID
##   FOREIGN KEY (mature_mirna_uid)
##     REFERENCES mirna(mature_mirna_uid)
##     ON UPDATE CASCADE ON DELETE RESTRICT,
##   FOREIGN KEY (target_uid)
##     REFERENCES target(target_uid)
##     ON UPDATE CASCADE ON DELETE RESTRICT
## );
##
## ......
##
## (Please note that only three of the 19 tables are shown here for demonstration
## purpose.)
```

The function `multimir_dbInfo()` will display information about the external miRNA and miRNA-target databases in *multiMiR*, including version, release date, link to download the data, and the corresponding table in *multiMiR*.

```
db.info = multimir_dbInfo()
db.info
##        map_name                  source_name source_version  source_date
## 1  diana_microt                 DIANA-microT              5   Sept, 2013
## 2         elmmo                        EIMMo              5    Jan, 2011
## 3     microcosm                    MicroCosm              5   Sept, 2009
## 4   mir2disease                  miR2Disease                Mar 14, 2011
## 5       miranda                      miRanda                   Aug, 2010
## 6         mirdb                        miRDB              5    Aug, 2014
## 7     mirecords                    miRecords              4 Apr 27, 2013
## 8    mirtarbase                   miRTarBase            6.1   Sept, 2015
## 9  pharmaco_mir Pharmaco-miR (Verified Sets)
## 10     phenomir                     PhenomiR              2 Feb 15, 2011
## 11       pictar                       PicTar              2 Dec 21, 2012
## 12         pita                         PITA              6 Aug 31, 2008
## 13      tarbase                      TarBase              6         2012
## 14   targetscan                   TargetScan            6.2    Jun, 2012
##                                                                         source_url
## 1  http://diana.imis.athena-innovation.gr/DianaTools/index.php?r=microT_CDS/index
## 2                         http://www.mirz.unibas.ch/miRNAtargetPredictionBulk.php
## 3      http://www.ebi.ac.uk/enright-srv/microcosm/cgi-bin/targets/v5/download.pl
## 4                                                     http://www.mir2disease.org
## 5                               http://www.microrna.org/microrna/getDownloads.do
## 6                                                               http://mirdb.org
## 7                                      http://mirecords.biolead.org/download.php
## 8                             http://mirtarbase.mbc.nctu.edu.tw/php/download.php
## 9                             http://www.pharmaco-mir.org/home/download_VERSE_db
## 10                                   http://mips.helmholtz-muenchen.de/phenomir/
## 11                                                   http://dorina.mdc-berlin.de
## 12                       http://genie.weizmann.ac.il/pubs/mir07/mir07_data.html
## 13   http://diana.imis.athena-innovation.gr/DianaTools/index.php?r=tarbase/index
## 14     http://www.targetscan.org/cgi-bin/targetscan/data_download.cgi?db=vert_61
```

Among the 14 external databases, eight contain predicted miRNA-target interactions (DIANA-microT-CDS, EIMMo, MicroCosm, miRanda, miRDB, PicTar, PITA, and TargetScan), three have experimentally validated miRNA-target interactions (miRecords, miRTarBase, and TarBase) and the remaining three contain miRNA-drug/disease associations (miR2Disease, Pharmaco-miR, and PhenomiR). To check these categories and databases from within R, we have a set of four helper functions:

```
predicted_tables()
## [1] "diana_microt" "elmmo"        "microcosm"    "miranda"      "mirdb"        "pictar"
## [7] "pita"         "targetscan"
validated_tables()
## [1] "mirecords"  "mirtarbase" "tarbase"
diseasedrug_tables()
## [1] "mir2disease"  "pharmaco_mir" "phenomir"
reverse_table_lookup("targetscan")
## [1] "predicted"
```

To see how many records are in these 14 external databases we refer to the multimir_dbCount function.

```
db.count = multimir_dbCount()
db.count
##        map_name human_count mouse_count rat_count total_count
## 1  diana_microt     7664602     3747171         0    11411773
## 2         elmmo     3959112     1449133    547191     5955436
## 3     microcosm      762987      534735    353378     1651100
## 4   mir2disease        2875           0         0        2875
## 5       miranda     5429955     2379881    247368     8057204
## 6         mirdb     1124831      654192    195136     1974159
## 7     mirecords        2425         449       171        3045
## 8    mirtarbase      440040       48731       470      489281
## 9  pharmaco_mir         308           5         0         313
## 10     phenomir       15138         491         0       15629
## 11       pictar      404066      302236         0      706302
## 12         pita     7710936     5163153         0    12874089
## 13      tarbase       23447        9024        67       32538
## 14   targetscan     6298122     2590814         0     8888936
apply(db.count[,-1], 2, sum)
## human_count mouse_count   rat_count total_count
##    33838844    16880015     1343781    52062680
```

The current version of *multiMiR* contains nearly 50 million records.

3 List miRNAs, genes, drugs and diseases in the multiMiR database

In addition to functions displaying database and table information, the *multiMiR* package also provides the `list.multimir()` function to list all the unique miRNAs, target genes, drugs, and diseases in the *multiMiR* database. An option for limiting the number of returned records has been added to help with testing and exploration.

```
miRNAs   = list.multimir("mirna", limit = 10)
genes    = list.multimir("gene", limit = 10)
drugs    = list.multimir("drug", limit = 10)
diseases = list.multimir("disease", limit = 10)
# executes 2 separate queries, giving 20 results
head(miRNAs)
##   mature_mirna_uid org mature_mirna_acc mature_mirna_id
## 1              936 hsa
## 2             1284 hsa                        hsa-let-7
## 3              202 hsa                     hsa-let-71f1
## 4              815 hsa                     hsa-let-7a-1
## 5              816 hsa                     hsa-let-7a-2
## 6              817 hsa                     hsa-let-7a-3
head(genes)
##   target_uid org target_symbol target_entrez  target_ensembl
## 1      60954 hsa                             ENSG00000011177
## 2      60956 hsa                             ENSG00000032514
## 3      60957 hsa                             ENSG00000051415
## 4      60958 hsa                             ENSG00000083622
## 5      60959 hsa                             ENSG00000099964
## 6      60960 hsa                             ENSG00000100101
head(drugs)
##                    drug
## 1 3,3'-diindolylmethane
## 2          5-fluoroucil
## 3               abt-737
## 4          alitretinoin
## 5       arabinocytosine
## 6      arsenic trioxide
head(diseases)
##                                                    disease
## 1 ACTH-INDEPENDENT MACRONODULAR ADRENAL HYPERPLASIA; AIMAH
## 2                      ACUTE LYMPHOBLASTIC LEUKEMIA (ALL)
## 3                       ACUTE MYELOGENEOUS LEUKEMIA (AML)
## 4                            ACUTE MYELOID LEUKEMIA (AML)
## 5                      ACUTE PROMYELOCYTIC LEUKEMIA (APL)
## 6                                                 ADENOMA
```

The current version of *multiMiR* has 5830 miRNAs and 97186 target genes from human, mouse, and rat, as well as 64 drugs and 223 disease terms. Depending on the speed of your Internet connection, it may take a few minutes to retrieve the large number of target genes.

4 Use `get.multimir()` to query the multiMiR database

`get.multimir()` is the main function in the package to retrieve predicted and validated miRNA-target interactions and their disease and drug associations from the *multiMiR* database.

To get familiar with the parameters in `get.multimir()`, you can type `?get.multimir` or `help(get.multimir)` in R. In the next section, many examples illustrate the use of the parameters.

5 Examples of multiMiR queries

In this section a variety of examples are described on how to query the multiMiR database.

5.1 Example 1: Retrieve all validated target genes of a given miRNA

In the first example, we ask what genes are validated targets of hsa-miR-18a-3p.

```
# The default is to search validated interactions in human
example1 <- get.multimir(mirna = 'hsa-miR-18a-3p', summary = TRUE)
## Searching mirecords ...
## Searching mirtarbase ...
## Searching tarbase ...
names(example1)
## [1] "validated"    "predicted"    "disease.drug" "queries"      "summary"
# Detailed information of the validated miRNA-target interaction
head(example1$validated)
##     database mature_mirna_acc mature_mirna_id target_symbol target_entrez  target_ensembl
## 1  mirecords     MIMAT0002891  hsa-miR-18a-3p          KRAS          3845 ENSG00000133703
## 2 mirtarbase     MIMAT0002891  hsa-miR-18a-3p          ACLY            47 ENSG00000131473
## 3 mirtarbase     MIMAT0002891  hsa-miR-18a-3p         AP1B1           162 ENSG00000100280
## 4 mirtarbase     MIMAT0002891  hsa-miR-18a-3p          ARF5           381 ENSG00000004059
## 5 mirtarbase     MIMAT0002891  hsa-miR-18a-3p        ATP2B4           493 ENSG00000058668
## 6 mirtarbase     MIMAT0002891  hsa-miR-18a-3p        ATP5A1           498 ENSG00000152234
##                                experiment          support_type pubmed_id
## 1 Western blot//Luciferase activity assay                        19372139
## 2                                   CLASH Functional MTI (Weak)  23622248
## 3                                   CLASH Functional MTI (Weak)  23622248
## 4                                   CLASH Functional MTI (Weak)  23622248
## 5                               HITS-CLIP Functional MTI (Weak)  19536157
## 6                                   CLASH Functional MTI (Weak)  23622248
```

```
# Which interactions are supported by Luciferase assay?
example1$validated[grep("Luciferase", example1$validated[, "experiment"]), ]
##      database mature_mirna_acc mature_mirna_id target_symbol target_entrez  target_ensembl
## 1   mirecords     MIMAT0002891  hsa-miR-18a-3p          KRAS          3845 ENSG00000133703
## 52 mirtarbase     MIMAT0002891  hsa-miR-18a-3p          KRAS          3845 ENSG00000133703
##                                           experiment   support_type pubmed_id
## 1           Western blot//Luciferase activity assay                 19372139
## 52 Luciferase reporter assay//qRT-PCR//Western blot Functional MTI  19372139
example1$summary[example1$summary[,"target_symbol"] == "KRAS",]
##   mature_mirna_acc mature_mirna_id target_symbol target_entrez  target_ensembl mirecords mirtarbase
## 1     MIMAT0002891  hsa-miR-18a-3p          KRAS          3845 ENSG00000133703         1          1
##   validated.sum all.sum
## 1             2       2
```

It turns out that *KRAS* is the only target validated by Luciferase assay. The interaction was recorded in miRecords and miRTarBase and supported by the same literature, whose PubMed ID is in column `pubmed_id`. The summary (by setting `summary = TRUE` when calling `get.multimir()`) shows the number of records in each of the external databases and the total number of databases supporting the interaction.

5.2 Example 2: Retrieve miRNA-target interactions associated with a given drug or disease

In this example we would like to know which miRNAs and their target genes are associated with Cisplatin, a chemotherapy drug used in several cancers.

```
example2 <- get.multimir(disease.drug = 'cisplatin', table = 'disease.drug')
## Searching mir2disease ...
## Searching pharmaco_mir ...
## Searching phenomir ...
names(example2)
## [1] "validated"    "predicted"    "disease.drug" "queries"      "summary"
nrow(example2$disease.drug)
## [1] 45
head(example2$disease.drug)
##       database mature_mirna_acc mature_mirna_id target_symbol target_entrez  target_ensembl
## 1 pharmaco_mir     MIMAT0000772  hsa-miR-345-5p         ABCC1          4363 ENSG00000103222
## 2 pharmaco_mir     MIMAT0000720 hsa-miR-376c-3p          ALK7
## 3 pharmaco_mir     MIMAT0000423 hsa-miR-125b-5p          BAK1           578 ENSG00000030110
## 4 pharmaco_mir                       hsa-miR-34          BCL2           596 ENSG00000171791
## 5 pharmaco_mir     MIMAT0000318 hsa-miR-200b-3p          BCL2           596 ENSG00000171791
## 6 pharmaco_mir     MIMAT0000617 hsa-miR-200c-3p          BCL2           596 ENSG00000171791
##   disease_drug paper_pubmedID
## 1    cisplatin       20099276
## 2    cisplatin       21224400
## 3    cisplatin       21823019
## 4    cisplatin       18803879
## 5    cisplatin       21993663
## 6    cisplatin       21993663
```

`get.multimir()` returns 53 miRNA-target pairs. For more information, we can always refer to the published papers with PubMed IDs in column `paper_pubmedID`.

5.3 Example 3: Select miRNAs predicted to target a gene

`get.multimir()` also takes target gene(s) as input. In this example we retrieve miRNAs predicted to target *Gnb1* in mouse. For predicted interactions, the default is to query the top 20% predictions within each external database, which is equivalent to setting parameters `predicted.cutoff = 20` and `predicted.cutoff.type = 'p'` (for percentage cutoff). Here we search the top 35% among all conserved and nonconserved target sites.

```
example3 <- get.multimir(org     = "mmu",
                         target  = "Gnb1",
                         table   = "predicted",
                         summary = TRUE,
                         predicted.cutoff      = 35,
                         predicted.cutoff.type = "p",
                         predicted.site        = "all")
## Searching diana_microt ...
## Searching elmmo ...
## Searching microcosm ...
## Searching miranda ...
## Searching mirdb ...
## Searching pictar ...
## Searching pita ...
## Searching targetscan ...
names(example3)
## [1] "validated"    "predicted"    "disease.drug" "queries"      "summary"
head(example3$predicted)
##       database mature_mirna_acc   mature_mirna_id target_symbol target_entrez     target_ensembl
## 1 diana_microt     MIMAT0000663    mmu-miR-218-5p          Gnb1         14688 ENSMUSG00000029064
## 2 diana_microt     MIMAT0017276    mmu-miR-493-5p          Gnb1         14688 ENSMUSG00000029064
## 3 diana_microt     MIMAT0000656    mmu-miR-139-5p          Gnb1         14688 ENSMUSG00000029064
## 4 diana_microt     MIMAT0014946 mmu-miR-3074-2-3p          Gnb1         14688 ENSMUSG00000029064
## 5 diana_microt     MIMAT0000144    mmu-miR-132-3p          Gnb1         14688 ENSMUSG00000029064
## 6 diana_microt     MIMAT0020608      mmu-miR-5101          Gnb1         14688 ENSMUSG00000029064
##   score
## 1 0.975
## 2 0.964
## 3  0.96
## 4 0.921
## 5  0.92
## 6 0.918
head(example3$summary)
##   mature_mirna_acc mature_mirna_id target_symbol target_entrez     target_ensembl diana_microt
## 1     MIMAT0000133 mmu-miR-101a-3p          Gnb1         14688 ENSMUSG00000029064            1
## 2     MIMAT0000616 mmu-miR-101b-3p          Gnb1         14688 ENSMUSG00000029064            1
## 3     MIMAT0003476 mmu-miR-669b-5p          Gnb1         14688 ENSMUSG00000029064            1
## 4     MIMAT0000663  mmu-miR-218-5p          Gnb1         14688 ENSMUSG00000029064            1
## 5     MIMAT0002106 mmu-miR-465a-5p          Gnb1         14688 ENSMUSG00000029064            1
## 6     MIMAT0003739  mmu-miR-673-5p          Gnb1         14688 ENSMUSG00000029064            1
##   elmmo microcosm miranda mirdb pictar pita targetscan predicted.sum all.sum
## 1     1         1       1     0      2    0          1             6       6
## 2     1         1       1     0      2    0          1             6       6
## 3     0         0       1     2      0    1          1             5       5
## 4     2         0       0     0      0    2          2             4       4
## 5     1         0       1     0      0    0          2             4       4
## 6     0         0       0     0      2    1          1             4       4
```

The records in `example3$predicted` are ordered by scores from best to worst within each external database. Once again, the summary option allows us to examine the number of target sites predicted by each external database and the total number of databases predicting the interaction.

Finally we examine how many predictions each of the databases has.

```
apply(example3$summary[, 6:13], 2, function(x) sum(x > 0))
## diana_microt        elmmo     microcosm       miranda         mirdb        pictar          pita
##          105           51             5            44            16             9           132
##   targetscan
##           58
```

5.4 Example 4: Select miRNA(s) predicted to target most, if not all, of the genes of interest

You may have a list of genes involved in a common biological process. It is interesting to check whether some, or all, of these genes are targeted by the same miRNA(s). Here we have four genes involved in chronic obstructive pulmonary disease (COPD) in human and want to know what miRNAs target these genes by searching the top 500,000 predictions in each external database.

```
example4 <- get.multimir(org     = 'hsa',
                         target  = c('AKT2', 'CERS6', 'S1PR3', 'SULF2'),
                         table   = 'predicted',
                         summary = TRUE,
                         predicted.cutoff.type = 'n',
                         predicted.cutoff      = 500000)
## Number predicted cutoff (predicted.cutoff) 500000 is larger than the total number of records in table p
## Searching diana_microt ...
## Searching elmmo ...
## Searching microcosm ...
## Searching miranda ...
## Searching mirdb ...
## Searching pictar ...
## Searching pita ...
## Searching targetscan ...
```

Then we count the number of target genes for each miRNA.

```
example4.counts <- addmargins(table(example4$summary[, 2:3]))
example4.counts <- example4.counts[-nrow(example4.counts), ]
example4.counts <- example4.counts[order(example4.counts[, 5], decreasing = TRUE), ]
head(example4.counts)
##                  target_symbol
## mature_mirna_id  AKT2 CERS6 S1PR3 SULF2 Sum
##   hsa-miR-429       1     1     0     1   3
##   hsa-miR-524-5p    1     1     0     1   3
##   hsa-miR-548p      0     1     1     1   3
##   hsa-miR-876-5p    0     1     1     1   3
##   hsa-let-7a-5p     1     0     0     1   2
##   hsa-let-7b-5p     1     0     0     1   2
```

5.5 Example 5: Retrieve interactions between a set of miRNAs and a set of genes

In this example, we profiled miRNA and mRNA expression in poorly metastatic bladder cancer cell lines T24 and Luc, and their metastatic derivatives FL4 and Lul2, respectively. We identified differentially expressed miRNAs and genes between the metastatic and poorly metastatic cells. Let's load the data.

```
load(url("http://multimir.ucdenver.edu/bladder.rda"))
```

Variable `DE.miRNA.up` contains 9 up-regulated miRNAs and variable `DE.entrez.dn` has 47 down-regulated genes in the two metastatic cell lines. The hypothesis is that interactions between these miRNAs and genes whose expression changed at opposite directions may play a role in cancer metastasis. So we use `multiMiR` to check whether any of the nine miRNAs could target any of the 47 genes.

```
## Searching diana_microt ...
## Searching elmmo ...
## Searching microcosm ...
## Searching miranda ...
## Searching mirdb ...
## Searching pictar ...
## Searching pita ...
## Searching targetscan ...
## Searching mirecords ...
## Searching mirtarbase ...
## Searching tarbase ...
## Searching mir2disease ...
## Searching pharmaco_mir ...
## Searching phenomir ...
```

In the result, there are 3 unique miRNA-gene pairs that have been validated.

```
##     database mature_mirna_acc mature_mirna_id target_symbol target_entrez
## 1 mirtarbase     MIMAT0000087  hsa-miR-30a-5p          FDX1          2230
## 2 mirtarbase     MIMAT0000087  hsa-miR-30a-5p        LIMCH1         22998
## 3    tarbase     MIMAT0000087  hsa-miR-30a-5p          FDX1          2230
## 4    tarbase     MIMAT0000424     hsa-miR-128          NEK2          4751
## 5    tarbase     MIMAT0000087  hsa-miR-30a-5p        LIMCH1         22998
##    target_ensembl               experiment          support_type pubmed_id
## 1 ENSG00000137714               Proteomics Functional MTI (Weak)  18668040
## 2 ENSG00000064042 pSILAC//Proteomics;Other Functional MTI (Weak)  18668040
## 3 ENSG00000137714               Proteomics              positive
## 4 ENSG00000117650               Microarray              positive
## 5 ENSG00000064042               Proteomics              positive
```

Two miRNAs are associated with bladder cancer in miR2Disease and PhenomiR.

```
##        database mature_mirna_acc mature_mirna_id target_symbol target_entrez
## 18  mir2disease     MIMAT0000418  hsa-miR-23b-3p            NA            NA
## 711    phenomir     MIMAT0000418  hsa-miR-23b-3p            NA            NA
## 311    phenomir     MIMAT0000449 hsa-miR-146a-5p            NA            NA
##     target_ensembl   disease_drug
## 18              NA bladder cancer
## 711             NA Bladder cancer
## 311             NA Bladder cancer
##                                                paper_pubmedID
## 18  2007. Micro-RNA profiling in kidney and bladder cancers.
## 711                                                  17826655
## 311                                                  19127597
```

The predicted databases predict 65 miRNA-gene pairs between the 9 miRNAs and 28 of the 47 genes.

```
## [1] 9
```

```
## [1] 28
```

```
unique.pairs <-
    unique(data.frame(miRNA.ID = as.character(example5$predicted$mature_mirna_id),
                      target.Entrez = as.character(example5$predicted$target_entrez)))
nrow(unique.pairs)
```

```
## [1] 65
```

```
##         miRNA.ID target.Entrez
## 1 hsa-miR-182-5p          2017
## 2 hsa-miR-182-5p          1112
## 3 hsa-miR-30d-5p         22998
## 4 hsa-miR-30a-5p         22998
## 5 hsa-miR-30b-5p         22998
## 6 hsa-miR-182-5p          5962
```

Results from each of the predicted databases are already ordered by their scores from best to worst.

6 Direct query to the database on the multiMiR web server

As shown previously, *get.multimir* is the main function to retrieve information from the *multiMiR* database, which is hosted at http://multimir.ucdenver.edu. The function builds one SQL query for every external database that the user is going to search, submits the query to the web server, and parses, combines, and summarizes results from the web server. For advanced users, there are a couple ways to query the *multiMiR* database without using the *multiMiR* package; but they have to be familiar with SQL queries. In general, users are still advised to use the `get.multimir()` function when querying multiple external databases in *multiMiR*.

6.1 Direct query on the web server

The *multiMiR* package communicates with the *multiMiR* database via the script http://multimir.ucdenver.edu/cgi-bin/multimir_univ.pl on the web server. Once again, data from each of the external databases is stored in a table in *multiMiR*. There are also tables for miRNAs (table *mirna*) and target genes (table *target*).

NOTE: While it is possible to complete short queries from a browser, the limits of submitting a query through typing in the address bar of a browser are quickly reached (8192 characters total). If you are a developer you should use your preferred method to submit a HTTP POST which will allow for longer queries. The fields to include are `query` and `dbName`. `query` is the SQL query to submit. `dbName` is the DBNAME column from a call to `multimir_dbInfoVersions()`, however if this is excluded the current version is the default.

To learn about the structure of a table (e.g. DIANA-microT data in table *diana_microt*), users can use URL

http://multimir.ucdenver.edu/cgi-bin/multimir_univ.pl?query=describe diana_microt

Similar with Example 1, the following URL searches for validated target genes of hsa-miR-18a-3p in miRecords.

http://multimir.ucdenver.edu/cgi-bin/multimir.pl?query=SELECT m.mature_mirna_acc, m.mature_mirna_id, t.target_symbol, t.target_entrez, t.target_ensembl, i.experiment, i.support_type, i.pubmed_id FROM mirna AS m INNER JOIN mirecords AS i INNER JOIN target AS t ON (m.mature_mirna_uid=i.mature_mirna_uid and i.target_uid=t.target_uid) WHERE m.mature_mirna_id='hsa-miR-18a-3p'

As you can see, the query is long and searches just one of the three validated tables in *multiMiR*. While in Example 1, one line of R command using the `get.multimir()` function searches, combines and summarizes results from all three validated external databases (miRecords, miRTarBase and TarBase).

6.2 Direct query in R

The same direct queries we did above on the web server can be done in R as well. This is the preferred method if you are unfamiliar with HTTP POST. Be sure to set the correct database version, if you wish to change versions, before calling `search.multimir()` it uses the currently set version.

To show the structure of table *diana_microt*:

```
direct2 <- search.multimir(query = "describe diana_microt")
direct2
##              Field             Type Null Key Default Extra
## 1 mature_mirna_uid int(10) unsigned   NO MUL
## 2       target_uid int(10) unsigned   NO MUL
## 3       miTG_score           double   NO MUL
## 4         UTR3_hit int(10) unsigned   NO
## 5          CDS_hit int(10) unsigned   NO
```

To search for validated target genes of hsa-miR-18a-3p in miRecords:

```
qry <- "SELECT m.mature_mirna_acc, m.mature_mirna_id, t.target_symbol,
               t.target_entrez, t.target_ensembl, i.experiment, i.support_type,
               i.pubmed_id
        FROM mirna AS m INNER JOIN mirecords AS i INNER JOIN target AS t
        ON (m.mature_mirna_uid=i.mature_mirna_uid and
            i.target_uid=t.target_uid)
        WHERE m.mature_mirna_id='hsa-miR-18a-3p'"
direct3 <- search.multimir(query = qry)
direct3
##   mature_mirna_acc mature_mirna_id target_symbol target_entrez  target_ensembl
## 1     MIMAT0002891  hsa-miR-18a-3p          KRAS          3845 ENSG00000133703
##                                 experiment support_type pubmed_id
## 1 Western blot//Luciferase activity assay               19372139
```

7 Session Info

```
sessionInfo()
## R version 3.3.2 (2016-10-31)
## Platform: x86_64-apple-darwin13.4.0 (64-bit)
## Running under: macOS Sierra 10.12.4
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] multiMiR_0.99.3 knitr_1.15.1    BiocStyle_2.2.1
```

```
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.10    XML_3.98-1.6    digest_0.6.12   rprojroot_1.2   bitops_1.0-6    backports_1.0.5
##  [7] magrittr_1.5    evaluate_0.10   stringi_1.1.5   rmarkdown_1.4   tools_3.3.2     stringr_1.2.0
## [13] purrr_0.2.2     RCurl_1.95-4.8  yaml_2.1.14     htmltools_0.3.5
warnings()
## NULL
```