The mirbaseID package

David Montaner

October 31, 2013

1 Summary

This document shows the basic usage of the package *mirbaseID*. The library contains pre-collected data form different versions of the miRBbase database. The purpose of the package is to provide utilities for the quick conversion between identifiers of different miRBbase versions.

2 Usage

The main dataset in the library is the `mirIDmat` matrix which you can access simply loading the package:

```
> library (mirbaseID)
> mirIDmat[1:3,]
```

```
             mirBase11   mirBase12   mirBase13   mirBase14   mirBase15
MIMAT0000001 "cel-let-7" "cel-let-7" "cel-let-7" "cel-let-7" "cel-let-7"
MIMAT0000002 "cel-lin-4" "cel-lin-4" "cel-lin-4" "cel-lin-4" "cel-lin-4"
MIMAT0000003 "cel-miR-1" "cel-miR-1" "cel-miR-1" "cel-miR-1" "cel-miR-1"
             mirBase16   mirBase17   mirBase18      mirBase19
MIMAT0000001 "cel-let-7" "cel-let-7" "cel-let-7-5p" "cel-let-7-5p"
MIMAT0000002 "cel-lin-4" "cel-lin-4" "cel-lin-4-5p" "cel-lin-4-5p"
MIMAT0000003 "cel-miR-1" "cel-miR-1" "cel-miR-1-3p" "cel-miR-1-3p"
             mirBase20
MIMAT0000001 "cel-let-7-5p"
MIMAT0000002 "cel-lin-4-5p"
MIMAT0000003 "cel-miR-1-3p"
```

```
> dim (mirIDmat)
```

```
[1] 30849    10
```

The `mirIDmat` matrix relates miRBbase IDs form different versions through the miRBbase accessions.

The function `buildVersion` uses the information in matrix `mirIDmat` to create a conversion vector to the desired miRBbase version. If we want for instance to get a conversion vector from any miRBbase version to the 18^{th} we can do:

```
> to18version <- buildVersion (18, verbose = FALSE)

  37 IDs do not have a unique conversion to mirBase version 18;
  they will be removed.
```

The returned vector *to18version* has, as names, all miRBbase IDs form all versions (including the 18th). The values of the vector are the IDs from miRBbase version 18th.

```
> to18version[1:4]

  aae-bantam-3p   aae-bantam-5p       aae-let-7       aae-miR-1
"aae-bantam-3p" "aae-bantam-5p"     "aae-let-7"     "aae-miR-1"
```

And can be used to convert any ID vector to the 18th version as follows:

```
> v <- c ("hsa-let-7a*", "hsa-miR-105*", "nada")
> to18version[v]

     hsa-let-7a*     hsa-miR-105*             <NA>
 "hsa-let-7a-3p" "hsa-miR-105-3p"               NA
```

For the 19th and 20th versions of miRBbase there are already pre-calculated vectors `id2mir19` and `id2mir20`:

```
> id2mir19[1:3]

  aae-bantam-3p   aae-bantam-5p       aae-let-7
"aae-bantam-3p" "aae-bantam-5p"     "aae-let-7"
```

That can be used as:

```
> v <- c ("hsa-let-7a*", "hsa-miR-105*", "nada")
> id2mir19[v]

     hsa-let-7a*     hsa-miR-105*             <NA>
 "hsa-let-7a-3p" "hsa-miR-105-3p"               NA
```

there is also a vector for the latest version of miRBbase called `id2mirLast`.

```
> v <- c ("hsa-let-7a*", "hsa-miR-105*", "nada")
> id2mirLast[v]

     hsa-let-7a*     hsa-miR-105*             <NA>
 "hsa-let-7a-3p" "hsa-miR-105-3p"               NA
```

There is also the a utility function to make the conversions straight froward

```
> conv <- c("cel-lin-4", "cel-miR-1", "mas", NA)
> idConvert (conv, mirversion = 18, verbose = FALSE)

  37 IDs do not have a unique conversion to mirBase version 18;
  they will be removed.
     cel-lin-4      cel-miR-1            mas           <NA>
"cel-lin-4-5p" "cel-miR-1-3p"             NA             NA
```

In current version of the *mirbaseID* package you can acces these miRBbase version.

```
> mirbVersions ()

 [1] "11" "12" "13" "14" "15" "16" "17" "18" "19" "20"
```

3 Download Date

The data in this version of the package *mirbaseID* was downloaded from miRBbase in:

```
> miRBaseDownloadDate

[1] "Thu Oct 31 17:57:50 2013"
```