Using inSilicoDb 2.0

Quentin De Clerck*

August 19, 2015

1 Functions from inSilicoDb 2.0

This new version of the package provides all the functionality of previous inSilicoDb packages developed by *Jonatan Taminau*. The functions did not change, therefore we refer to the description of previous versions.

1.1 Access to InSilico MySafe

One of the new features of this package is the possibility for users to access their private data stored on InSilico MySafe. This feature is implemented by the functions:

InSilicoLogin(login, password) Logs the user in for the given login and password. There is no secure way to login to a webservice in R. Therefore the password given to the function has to be the md5 hash of the real password. After login in, the user can use the normal functions as described earlier with their private data.

InSilicoLogout() Logs the currently user out of the InSilico DB webservice.

getInSilicoUserDetails() Returns informations (id, name, email) about the user that is currently logged in.

1.2 Check the accessibility of data

Some helper functions were added to provide information about the availability of data: getDatasetInfo and getPlatformList.

The purpose of getDatasetInfo is two-fold. First, it will return the default values of all optional parameters and the title of the study. Second, it will return the availability of the requested dataset for specified parameters. It returns an error if the data is not available for download. You need to login to access datasets and datasetinfo. Use your InSilicoDB login and an md5 hash of your password. For this example we're using a restricted test account.

---

*qdeclerc@vub.ac.be

```
> library("inSilicoDb");
> InSilicoLogin("rpackage_tester@insilicodb.com", "5c4d0b231e5cba4a0bc54783b385cc9a");

[1] 5296

> eset = getDatasetInfo("GSE781", "GPL96");
> print(eset);

$dataset
[1] "GSE781"

$platform
[1] "GPL96"

$format
[1] "CURESET"

$features
[1] "PROBE"

$norm
[1] "ORIGINAL"

$curation
[1] 4864

$type
[1] "RNA"

$title
[1] "Normal and Renal Cell Carcinoma Kidney Tissue, Human"

$state
[1] "finished"

$path
[1] "GSE781GPL96_RNA_ORIGINALPROBE_4864.RData"

> ## We check the availability of following normalizations for series GSE781 on platform GPl
> norms = c("FRMA", "ORIGINAL")
> output = sapply(norms, function(n) {
+        tryCatch({
+                        eset <- getDatasetInfo("GSE781", "GPL97", norm = n);
+                        eset$norm;
+                }, error = function(e) {
+                        "Unavailable"
+                });
```

```
+           });
> print(output);

         FRMA      ORIGINAL
"Unavailable"    "ORIGINAL"

>
> # We can thus conclude that the series GSE781 on platform GPL97 does not support FRMA
```

The getPlatformList function returns all the platforms supported by the package. It is also possible to query the availability of the normalizations using this function.

```
> platforms = getPlatformList();
> print(platforms);

 [1] "GPL96"    "GPL570"   "GPL571"   "GPL3921"  "GPL4685"  "GPL97"
 [7] "GPL1355"  "GPL1261"  "GPL8321"  "GPL91"    "GPL8300"  "GPL85"
[13] "GPL6947"  "GPL10558" "GPL13667" "GPL13916" "GPL84"

> FRMAplatforms = getPlatformList(norm = "FRMA");
> print(FRMAplatforms);

[1] "GPL96"    "GPL570"   "GPL571"   "GPL3921"  "GPL1261"  "GPL8321"  "GPL13916"

>
```

1.3   Curated Data

What changed in the functionality with respect to previous package is the better integration of the curated data. The user can now decide if he wants curated data or not by specifying the format.There are two possible formats: ESET and CURESET. By default the format is CURESET, this means with curated data. It is also possible to specify the curation wanted for the requested expression set.

```
> # without curated data
> eset = getDataset("GSE4635", "GPL96", format = "ESET");
> # with curated data
> cureset = getDataset("GSE4635", "GPL96", format = "CURESET", curation = 9016);
> print(phenoData(eset));

An object of class 'AnnotatedDataFrame': none

> print(phenoData(cureset));

An object of class 'AnnotatedDataFrame'
  Measurements: GSM15729 GSM104072 ... GSM104082 (8 total)
  varLabels: characteristics source_name title description
  varMetadata: labelDescription
```

3

It is possible to get information about the default curation (id, time, curator) of a particular dataset by calling the getDefaultCuration function. Using the getCurationInfo function returns a complete overview of all the possible curations in more details.

```
> default = getDefaultCuration("GSE4635");
> print(default);

$version
[1] 12

$date
[1] "2015-07-12 09:21:45"

$versiontoshow
[1] TRUE

$updateversion
[1] 0

$private
[1] FALSE

$id
[1] 36314

> getCurationInfo("GSE4635");
>
```

2 SCAN normalizations

Two normalizations were added in to the package. It is now possible to request data in Single Channel Array Normalization (SCAN) [1] SCAN is a possible alternative to fRMA outputs the probabilities of the probes/genes being expressed.

```
> scan = getDataset("GSE7670", "GPL96", norm = "SCAN", features = "gene");
> # example of values in SCAN normalization
> print(exprs(scan)[1:10, 1:5]);

          GSM185811    GSM185812    GSM185813    GSM185814    GSM185815
A1CF   -0.07358740 -0.18544195 -0.14047652 -0.047070543 -0.088030791
A2M     3.97302974   3.61329254   3.64422315   3.175149736   3.706938979
A4GALT -0.10135320 -0.21700383   0.01020519   0.004723602 -0.016453412
A4GNT  -0.04818053 -0.15516792 -0.09574640 -0.269577022 -0.163187218
AAAS    0.09985320   0.30364570   0.35217938   0.412674982   0.328749496
```

4

```
AACS    0.63019153  0.58221175  0.68837054  0.741976053  0.567470020
AADAC   0.54634280  0.11523122  0.23631537 -0.004144228  0.538300953
AAGAB   0.25438358  0.21286124  0.32846200  0.208153623  0.257049573
AAK1    0.02925236 -0.07714403  0.03666870  0.015383268  0.007400342
AAMDC   0.60971253  0.29819718  0.49897349  0.184285389  0.444644551
```

3   Session Info

```
> sessionInfo()

R version 3.2.2 (2015-08-14)
Platform: x86_64-pc-linux-gnu (64-bit)
Running under: Ubuntu 14.04.3 LTS

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_US.UTF-8        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
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

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

other attached packages:
[1] limma_3.24.15      inSilicoDb_2.4.1   RCurl_1.95-4.7
[4] bitops_1.0-6       Biobase_2.28.0     BiocGenerics_0.14.0
[7] rjson_0.2.15

loaded via a namespace (and not attached):
[1] tools_3.2.2
```

References

[1] Stephen R Piccolo, Ying Sun, Joshua D Campbell, Marc E Lenburg, Andrea H Bild, and W Evan Johnson. A single-sample microarray normalization method to facilitate personalized-medicine workflows. *Genomics*, 100(6):337–344, 2012.