MGFR: Marker Gene Finder in RNA-seq data

Khadija El Amrani *†

May 19, 2021

Contents

1 Introduction

Identification of marker genes associated with a specific tissue/cell type is a fundamental challenge in genetic and genomic research. In addition to other genes, marker genes are of great importance for understanding the gene function, the molecular mechanisms underlying complex diseases, and may lead to the development of new drugs. We updated our marker tool MGFM [1] to work with RNA-seq data.
`MGFR` is a package enabling the detection of marker genes from RNA-seq data.

2 Requirements

The tool expects replicates for each sample type. Using replicates has the advantage of increased precision of gene expression measurements and allows smaller changes to be detected. It is not

*Charité-Universitätsmedizin Berlin, Berlin Brandenburg Center for Regenerative Therapies (BCRT), 13353 Berlin, Germany
†Package maintainer, Email: `a.khadija@gmx.de`

necessary to use the same number of replicates for all sample types. Normalization is necessary before any analysis to ensure that differences in intensities are indeed due to differential expression, and not to some experimental factors that add systematic biases to the measurements.

3 Contents of the package

The `MGFR` package contains the following objects:

```
> library("MGFR")
> ls("package:MGFR")

[1] "getMarkerGenes.rnaseq"      "getMarkerGenes.rnaseq.html"
[3] "ref.mat"
```

The functions `getMarkerGenes.rnaseq()` and `getMarkerGenes.rnaseq.html()` are the main functions, and `ref.mat` is an example RNA-seq data set, which is used for demonstration.

3.1 `getMarkerGenes.rnaseq`

`getMarkerGenes.rnaseq()` performs marker gene detection using a given RNA-seq expression matrix and returns a list of marker genes associated with each given sample type.

3.1.1 Parameter Settings

1. *data.mat*: RNA-Seq gene expression matrix with genes corresponding to rows and samples corresponding to columns.
2. *class.vec*: A character vector containing the classes of samples (columns) of data.mat in the same order as provided in the matrix.
3. *samples2compare* (optional): A character vector with the sample names to be compared (e.g. c("liver", "lung", "brain")). By default all samples are used.
4. *annotate* (optional): A boolean value. If TRUE the gene symbol and the entrez gene id are shown. Default is FALSE. For mapping between gene ids and gene symbols, the Bioconductor R package *biomaRt* is used.
5. *gene.ids.type*: Type of the used gene identifiers, the following gene identifiers are supported: ensembl, refseq and ucsc gene ids. Default is ensembl.
6. *score.cutoff* (optional): It can take values in the interval [0,1]. This value is used to filter the markers according to their specificity score. The default value is 1 (no filtering).

3.1.2 Output

The function `getMarkerGenes.rnaseq()` returns a list as output. The entries of the result list contain the markers that are associated with each given sample type. For each marker the gene id, the gene symbol, the entrez gene id and the corresponding specificity score are shown in this order.

3.2 `getMarkerGenes.rnaseq.html`

`getMarkerGenes.rnaseq.html()` is a function to detect marker genes using normalized RNA-seq data and show the marker genes in HTML tables with links to various online annotation sources (Ensembl, GenBank and EntrezGene repositories).

3.2.1 Parameter Settings

1. *data.mat*: RNA-Seq gene expression matrix with genes corresponding to rows and samples corresponding to columns.
2. *class.vec*: A character vector containing the classes of samples (columns) of data.mat in the same order as provided in the matrix.
3. *samples2compare* (optional): A character vector with the sample names to be compared (e.g. c("liver", "lung", "brain")). By default all samples are used.
4. *gene.ids.type*: Type of the used gene identifiers, the following gene identifiers are supported: ensembl, refseq and ucsc gene ids. Default is ensembl.
5. *score.cutoff* (optional): It can take values in the interval [0,1]. This value is used to filter the markers according to their specificity score. The default value is 1 (no filtering).
6. *directory*: Path to the directory where to save the html pages, default is the current working directory.

3.2.2 Output

The function `getMarkerGenes.rnaseq.html()` is used only for the side effect of creating HTML tables.

3.3 Example data

`ref.mat`: is an RNA-seq gene expression data set derived from 5 tissue types (lung, liver, heart, kidney, and brain) from the ArrayExpress (www.ebi.ac.uk/arrayexpress) database (E-MTAB-1733 [2]). Each tissue type is represented by 3 replicates.

```
> data("ref.mat")
> dim(ref.mat)

[1] 35287    15

> colnames(ref.mat)

 [1] "brain"  "brain"  "brain"  "heart"  "heart"  "heart"
 [7] "kidney" "kidney" "kidney" "liver"  "liver"  "liver"
[13] "lung"   "lung"   "lung"
```

4 Processing of RNA-seq data

The reads from the study E-MTAB-1733 were mapped to the GRCh37 version of the human genome with Tophat v2.1.0. FPKM (fragments per kilobase of exon model per million mapped reads) values were calculated using Cufflinks v2.2.1. The used example data was extracted after processing of all samples and averaging across technical replicates.

5 Marker search

To use the package, we should load it first.

```
> library(MGFR)
> data("ref.mat")
> markers.list <- getMarkerGenes.rnaseq(ref.mat, class.vec = colnames(ref.mat),samples2compare="a
```

```
Detecting marker genes...
Mapping gene IDs to gene symbols...
Done!
```

```
> names(markers.list)
```

```
[1] "brain_markers"  "heart_markers"  "kidney_markers"
[4] "liver_markers"  "lung_markers"
```

```
> # show the first 20 markers of liver
> markers.list[["liver_markers"]][1:20]
```

```
 [1] "ENSG00000080910 : CFHR2 : 3080 : 0"
 [2] "ENSG00000101981 : F9 : 2158 : 0"
 [3] "ENSG00000105398 : SULT2A1 : 6822 : 0"
 [4] "ENSG00000105550 : FGF21 : 26291 : 0"
 [5] "ENSG00000109181 : UGT2B10 : 7365 : 0"
 [6] "ENSG00000111700 : SLCO1B3 : 28234 : 0"
 [7] "ENSG00000122787 : AKR1D1 : 6718 : 0"
 [8] "ENSG00000132703 : APCS : 325 : 0"
 [9] "ENSG00000134538 : SLCO1B1 : 10599 : 0"
[10] "ENSG00000143278 : F13B : 2165 : 0"
[11] "ENSG00000145192 : AHSG : 197 : 0"
[12] "ENSG00000148965 : SAA4 : 6291 : 0"
[13] "ENSG00000151631 : AKR1C6P : NA : 0"
[14] "ENSG00000157131 : C8A : 731 : 0"
[15] "ENSG00000158731 : OR10J6P : NA : 0"
[16] "ENSG00000158874 : APOA2 : 336 : 0"
[17] "ENSG00000165471 : MBL2 : 4153 : 0"
[18] "ENSG00000180210 : F2 : 2147 : 0"
[19] "ENSG00000197851 : NA : NA : 0"
[20] "ENSG00000198099 : ADH4 : 127 : 0"
```

6 MGFR algorithm details

Marker genes are identified as follows:

- **Sort of expression values for each gene:** In this step the expression values are sorted in decreasing order.
- **Marker selection**: To analyze the sorted distribution of expression values of a gene to define if it is a potential candidate marker we define cut-points as those that segregate samples of different types. A sorted distribution can have multiple cut-points; a cut-point can segregate one sample type from the others, or it can segregate multiple sample types from multiple sample types. Each cut-point is defined by the ratio of the expression averages of the groups of samples adjacent to it. That is, given a distribution with n cut-points and n+1 segregated groups, cut-point i receives a score that is the ratio of the average expression of samples in the group i+1 (following the cut-point) divided by that of group i (preceding the cut-point). This value is < 1 because the values are sorted in decreasing order. The closer the values, the closer the score to 1 and therefore the smaller is the gap between expression values at the cut-point. The specificity score of a marker gene is defined as the score of the first cut-point. For simplicity, we take only genes as markers if they have a cut-point that segregates one tissue at high expression from the rest. We disregard negative markers (segregating samples from one tissue at low expression) or multiple tissue markers (segregating samples from more than one tissue from other multiple tissues).

7 Conclusion

The development of this tool was motivated by the desire to provide a software package that enables the user to get marker genes associated with a set of samples of interest. A further objective of this tool was to enable the user to modify the set of samples of interest by adding or removing samples in a simple way.

8 R sessionInfo

The results in this file were generated using the following packages:

```
> sessionInfo()

R version 4.1.0 RC (2021-05-10 r80283)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows Server x64 (build 17763)

Matrix products: default

locale:
[1] LC_COLLATE=C
[2] LC_CTYPE=English_United States.1252
[3] LC_MONETARY=English_United States.1252
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.1252

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

other attached packages:
[1] MGFR_1.18.0

loaded via a namespace (and not attached):
 [1] KEGGREST_1.32.0        progress_1.2.2
 [3] tidyselect_1.1.1       purrr_0.3.4
 [5] vctrs_0.3.8            generics_0.1.0
 [7] stats4_4.1.0           BiocFileCache_2.0.0
 [9] utf8_1.2.1             blob_1.2.1
[11] XML_3.99-0.6           rlang_0.4.11
[13] pillar_1.6.1           withr_2.4.2
[15] glue_1.4.2             DBI_1.1.1
[17] rappdirs_0.3.3         BiocGenerics_0.38.0
[19] bit64_4.0.5            dbplyr_2.1.1
[21] GenomeInfoDbData_1.2.6 lifecycle_1.0.0
[23] stringr_1.4.0          zlibbioc_1.38.0
[25] Biostrings_2.60.0      memoise_2.0.0
[27] Biobase_2.52.0         IRanges_2.26.0
[29] fastmap_1.1.0          biomaRt_2.48.0
[31] GenomeInfoDb_1.28.0    parallel_4.1.0
[33] curl_4.3.1             AnnotationDbi_1.54.0
[35] fansi_0.4.2            Rcpp_1.0.6
[37] xtable_1.8-4           filelock_1.0.2
```

```
[39] cachem_1.0.5           S4Vectors_0.30.0
[41] annotate_1.70.0        XVector_0.32.0
[43] bit_4.0.4              hms_1.1.0
[45] png_0.1-7              digest_0.6.27
[47] stringi_1.6.2          dplyr_1.0.6
[49] tools_4.1.0            bitops_1.0-7
[51] magrittr_2.0.1         RCurl_1.98-1.3
[53] RSQLite_2.2.7          tibble_3.1.2
[55] crayon_1.4.1           pkgconfig_2.0.3
[57] ellipsis_0.3.2         xml2_1.3.2
[59] prettyunits_1.1.1      assertthat_0.2.1
[61] httr_1.4.2             rstudioapi_0.13
[63] R6_2.5.0               compiler_4.1.0
```

References

[1] Khadija El Amrani, Harald Stachelscheid, Fritz Lekschas, Andreas Kurtz, and Miguel A Andrade-Navarro. MGFM: a novel tool for detection of tissue and cell specific marker genes from microarray gene expression data. *BMC genomics*, 16(1):645, jan 2015.

[2] Linn Fagerberg, Björn M Hallström, Per Oksvold, Caroline Kampf, Dijana Djureinovic, Jacob Odeberg, Masato Habuka, Simin Tahmasebpoor, Angelika Danielsson, Karolina Edlund, Anna Asplund, Evelina Sjöstedt, Emma Lundberg, Cristina Al-Khalili Szigyarto, Marie Skogs, Jenny Ottosson Takanen, Holger Berling, Hanna Tegel, Jan Mulder, Peter Nilsson, Jochen M Schwenk, Cecilia Lindskog, Frida Danielsson, Adil Mardinoglu, Asa Sivertsson, Kalle von Feilitzen, Mattias Forsberg, Martin Zwahlen, IngMarie Olsson, Sanjay Navani, Mikael Huss, Jens Nielsen, Fredrik Ponten, and Mathias Uhlén. Analysis of the human tissue-specific expression by genome-wide integration of transcriptomics and antibody-based proteomics. *Molecular & cellular proteomics : MCP*, 13(2):397–406, feb 2014.

[3] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2007. ISBN 3-900051-07-0.