Chapter 9

Microarray-Based MicroRNA Expression Data Analysis with Bioconductor

Emilio Mastriani, Rihong Zhai, and Songling Zhu

Abstract

MicroRNAs (miRNAs) are small, noncoding RNAs that are able to regulate the expression of targeted mRNAs. Thousands of miRNAs have been identified; however, only a few of them have been functionally annotated. Microarray-based expression analysis represents a cost-effective way to identify candidate miRNAs that correlate with specific biological pathways, and to detect disease-associated molecular signatures. Generally, microarray-based miRNA data analysis contains four major steps: (1) quality control and normalization, (2) differential expression analysis, (3) target gene prediction, and (4) functional annotation. For each step, a large couple of software tools or packages have been developed. In this chapter, we present a standard analysis pipeline for miRNA microarray data, assembled by packages mainly developed with *R* and hosted in Bioconductor project.

Key words MicroRNA (miRNA), Bioconcductor, R Package, Gene expression analysis, Microarray data analysis

1 Introduction

MicroRNAs (miRNAs) are small, noncoding and conserved RNA molecules that can inhibit protein expression by post-transcriptional regulation or translational repression. More than 20,000 different miRNAs have been disclosed among hundreds of species [1]. Although miRNAs play important roles in various biological processes, the function has only been well clarified for a small subset.

The expression profile of miRNAs often shows developmental stage or tissue specific patterns, suggesting that they may participate in the specific regulatory processes [2, 3]. Microarray is attractive to profile the miRNA expression under different conditions because it can detect thousands of miRNAs simultaneously [4]. Compared with other high-throughput technique, such as RNA-Seq, the cost of microarray-based studies appears much

Yejun Wang and Ming-an Sun (eds.), *Transcriptome Data Analysis: Methods and Protocols*, Methods in Molecular Biology, vol. 1751, https://doi.org/10.1007/978-1-4939-7710-9_9,

lower and hundreds or thousands of biological samples can be studied in one experiment with a cost-effective way.

There is some difference between the analytic pipelines of miRNA and other microarray-based expression data. Besides the routine preprocessing, expression comparison and functional annotation, miRNA data also involve additional target prediction and target gene annotation steps. For each step, a large number of bioinformatic tools have been developed. Experimental researchers will struggle to find, assemble and test the tools for the task of each step. In this chapter, we are going to present a pipeline specific for microarray-based miRNA expression data analysis. The pipeline is assembled by packages mostly hosted in Bioconductor project, and therefore all the analysis can be completed in *R* environment conveniently (R: http://www.r-project.org; Bioconductor: http://www.bioconductor.org).

2 Materials

2.1 Software Tools

2.1.1 R/Bioconductor

The most recent version of R was downloaded and installed. For this chapter, Linux platform is used. For R installation and administration, the FAQs and documents can be referred: https://www.r-project.org/. Bioconductor can be installed by entering the following commands after starting R:

```
> source("https://bioconductor.org/biocLite.R")
> biocLite()
```

2.1.2 Installation of R/Bioconductor Packages

Install the R/Bioconductor packages for miRNA microarray data analysis with `biocLite()`. The packages are summarized in Table 1 [5–16].

```
> biocLite(c("Biobase", "GEOquery", "limma", "mclust",
"devtools",
+ "GOstats","gplots","networkD3","miRNAtap","miRNAtap.db",
+ "visNetwork","SpidermiR"))
```

2.2 Datasets

A public available dataset, GSE54578, is used as an example for demonstration (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE54578). The study profiles genome-wide miRNA expression in blood from 15 early-onset schizophrenia cases and 15 healthy controls, detecting a total of 1070 miRNAs by the microarrays [17]. A GPL16016 platform (Exiqon miRCURY LNA microRNA array) was used [17]. The dataset can be downloaded through the link directly; alternatively, it can be accessed with "`getGEO`" function of the "`GEOquery`" package.

```
> library("GEOquery")
> gset <- getGEO("GSE54578",GSEMatrix=TRUE,AnnotGPL=FALSE)
```

Table 1
R packages used in the chapter for miRNA data analysis

Package name	Short description
Biobase [5]	Functions that are needed by many other packages or which replace R functions
devtools [6]	Collection of package development tools
GOstats [7]	Tools for manipulating GO and microarrays
GEOquery [8]	GEOquery is the bridge between GEO and BioConductor
gplots [9]	Various R programming tools for plotting data
limma [10]	Data analysis, linear models and differential expression for microarray data
mclust [11]	Gaussian finite mixture models fitted via EM algorithm for model-based clustering, classification, and density estimation
miRNAtap [12]	microRNA targets aggregated predictions
miRNAtap.db [13]	Holding the database for miRNAtap
networkD3 [14]	Creates 'D3' 'JavaScript' network, tree, dendrogram, and Sankey graphs from 'R'
SpidermiR [15]	The package provides multiple methods for query, prepare and download network data, and the integration with validated and predicted miRNA data and the use of standard analysis and visualization methods
visNetwork [16]	Provides an R interface to the 'vis.js' JavaScript charting library

```
> if(length(gset)>1) idx <- grep("GPL16016",attr(gset,"-
names")) else idx <- 1
> gset <- gset[[idx]]
```

The GSE54578 dataset is now stored in `gset`, which will be used for further processing and analysis.

3 Methods

3.1 Preprocessing and Normalization

3.1.1 Preprocessing

The original miRNA expression data could contain some "NA" values and the columns are named with GSM accessions in default. The data structure and content can be shown with "`head(exprs(gset))`" command (Fig. 1a). In the preprocessing step, we may wish to remove all the "NA" records and rename the columns with user-readable format (Fig. 1b).

```
> head(exprs(gset))
> rmv <- which(apply(exprs(gset),1,function(x) any (is.na
(x))))
```

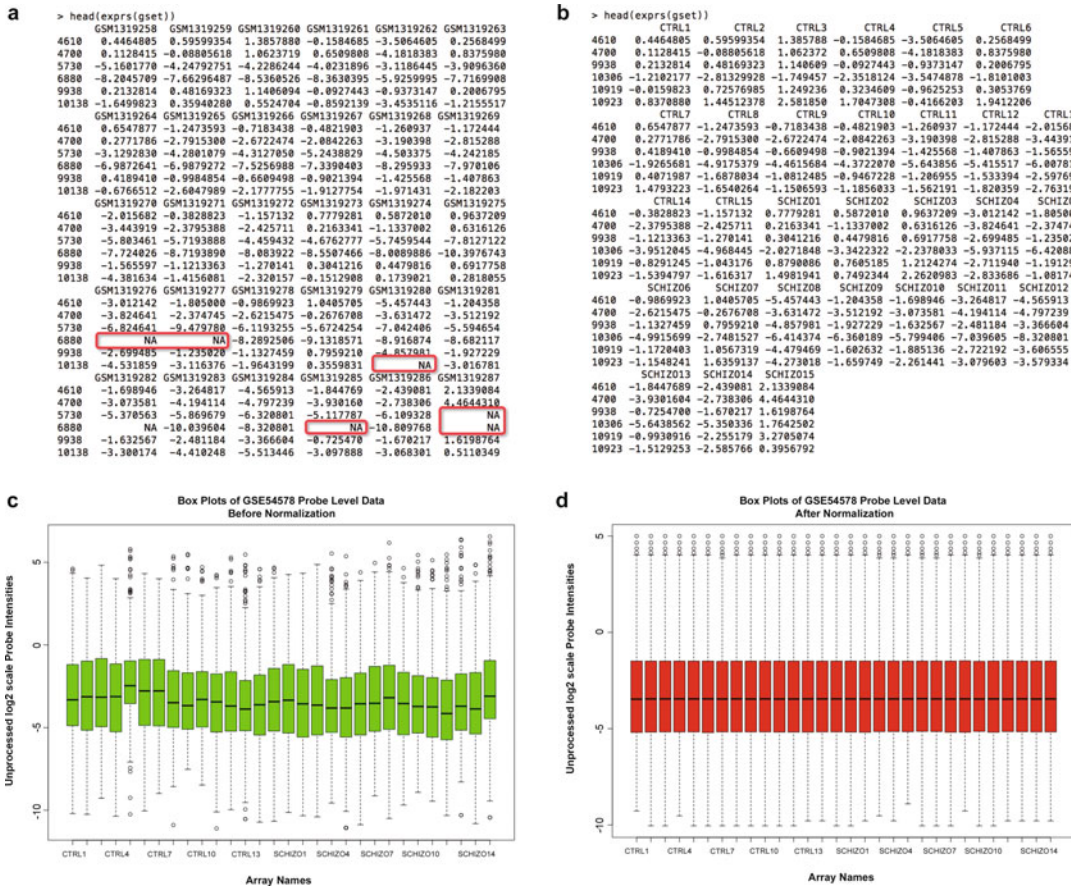

Fig. 1 Preprocessing of miRNA microarray data. (**a**) Raw expression data containing "NA" values. (**b**) "NA" filtered expression data. (**c**) Variance among samples before normalization. (**d**) Variance among samples after normalization

```
> gset <- gset[-rmv,]
> sampleNames(gset) <- c("CTRL1", ...,"CTRL15","SCHIZO1",...,"SCHIZO15")
> gsms <-"00000000000000011111111111111" #Grouping names
> sml <- c()
> for(i in 1:nchar(gsms)) {sml[i] <- substr(gsms,i,i)}
> head(exprs(gset))
```

Note that the "CTRL2"~"CTRL14" and "SCHIZO1"~"SCHIZO15" were omitted in the demonstrated command line.

Before normalization, the probe intensities should be checked to find out the apparent outliers caused by nonsystem errors. These outliers must be excluded for further analysis. Typically, a "`boxplot`" can be generated and show the uniformity of the signal intensity.

```
> ex <- exprs(gset)
> boxplot(ex, which='pm', ylab="Intensities", xlab="Array names")
```

After recalling and filtering the arrays with apparent experimental biases, the general signal intensity distribution should follow the distribution patterns as in Fig. 1c, with small variance among arrays.

3.1.2 Normalization

After preprocessing, the microarray data must be normalized to get rid of variations with nonbiological sources. A large number of methods have been proposed to normalize microarray-based transcriptome data. The methods are suited for different platforms and integrated in packages for corresponding data analysis, e.g., "`NormiR`" function in the "`ExiMiR`" package for two-color microarray experiments using a common reference or similar methods in the "`affy`" package for single-channel Affymetrix arrays, "`normalizeBetweenArrays`" function in the "`limma`" package, etc. In the example, "`normalizeBetweenArrays`" is applied, with a quantile normalization procedure.

```
> library("limma")
> ex_norm <- normalizeBetweenArrays(ex)
> qu <- as.numeric(quantile(ex,c(0.,0.25,0.5,0.75,0.99,1.0),
na.rm=T))
> filt <- ( qu[5]>100 || (qu[6]-qu[1]>50 && qu[2]>0) || (qu[2]>
0 && qu[2]<1 && qu[4]>1
&& qu[4]<2))
> if(filt){ex_norm[which(ex<=0)] <- NaN; exprs(gset) <- log2
(ex_norm)}
```

A log2 transformation is done to the normalized expression values to make the data follow Gaussian distribution more approximately. A boxplot generated with the normalized data shows more even distribution of the expression levels among different arrays (Fig. 1d).

3.2 Expression Difference and Clustering Analysis

The normalized expression data can be compared directly between groups. *T* Test is the most straightforward statistic comparison method between two groups, which will measure the significance of difference with probability of no difference (*p* values: the lower, the more significant). For microarray data, tens of thousands of genes are compared between groups simultaneously and it is a massive multiple testing problem. It is more complicated that the measured expression levels do not always follow normal distributions and have nonidentical and dependent distributions between genes. To solve this problem and identify the differentially expressed genes more precisely, Smyth proposed an empirical Bayes moderated t test, which has been incorporated into the "`limma`" package [10]. An example is shown as following, and more details about the usage of "`eBayes`" can refer to the document: http://web.mit.edu/~r/current/arch/i386_linux26/lib/R/library/limma/html/ebayes.html.

```
> sml <- paste("G",sml,sep="")
> fl <- as.factor(sml)
> gset$description <- fl
&gt; design &lt;- model.matrix(~ description + 0, gset)
> colnames(design) <- levels(fl)
> fit <- lmFit(gset,design)
> cont.matrix <- makeContrasts(G1-G0,levels=design)
> fit2 <- contrasts.fit(fit,cont.matrix)
> fit2 <- eBayes(fit2,0.01)
> tT <- topTable(fit2,adjust="fdr",sort.by="B",number=1000)
```

The comparison results are stored in objects *fit2* and *tT*, which will be used for further analysis.

Besides the significance measured by the statistic *p* values, the fold change amplitude of miRNA gene expression levels also appears important to biologists. A volcano plot can show the statistic significance and change amplitude in a two-dimensional plane simultaneously, which plots the fold change and *p* values (log-transformed results) on *x*- and *y*-axis respectively (Fig. 2a). The "`volcanoplot`" function in the "`limma`" package can be applied conveniently. Note that the 'highlight' argument indicates the top probe sets are highlighted. Other packages such as "`ggplot2`" also have functions to draw volcano plots.

```
> volcanoplot(fit2,coef=1,highlight=10)
```

Alternatively, basic *R* plot function can also generate the volcano plot.

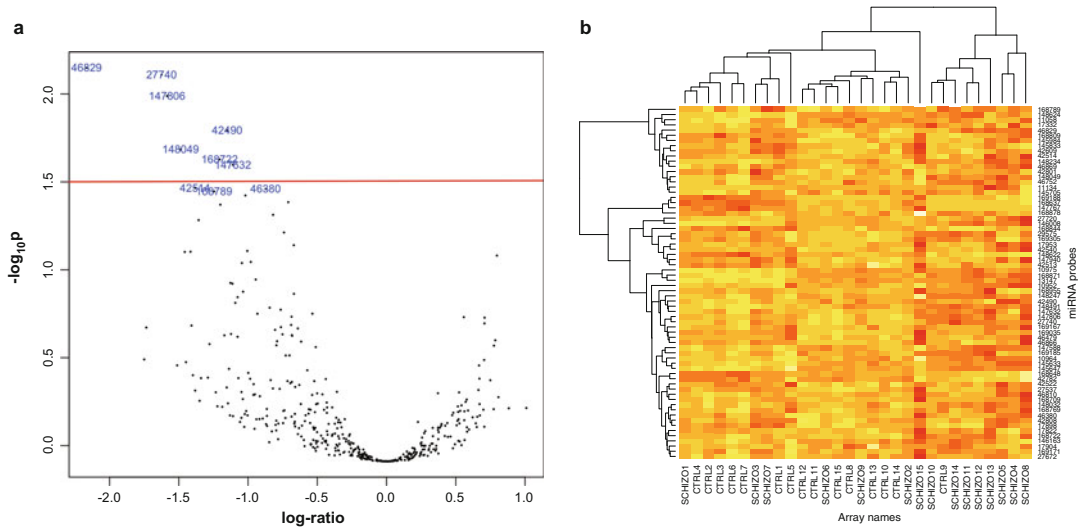

Fig. 2 Volcano plot and heat map of miRNA expression data. (**a**) Volcano plot showing the differentially expressed miRNAs between disease and control samples. (**b**) Clustering the samples and genes with expression patterns of significantly differential miRNAs

```
> lod <- -log10(tT$adj.P.Val)
> plot(tT$logFC,lod,xlab="log-ratio",ylab=expression(-log[10]~p))
> abline(h=1.5,col="red")
```

As in other transcriptome data analysis, besides gene expression difference analysis, clustering analysis can also be performed for miRNA microarray data. For example, a simple heatmap plot can be generated for a subset of the miRNAs with significant expression difference between disease and control (Fig. 2b; FDR adjusted p-value < 0.05).

```
> selected <- which(p.adjust(fit2$p.value[,1]<0.05) == 1)
> esetSel <- ex_norm[selected,]
> heatmap(esetSel)
```

For more in-depth clustering analysis, readers can refer to Chapter 2 of the book, since the procedure and tools are general rather than specific for miRNA datasets.

3.3 miRNA Target Analysis

3.3.1 Target Identification

The difference between miRNA and general transcriptome data analysis is mainly represented by the specific target gene analysis of the former. The major activity of miRNAs is to regulate the expression of target genes posttranscriptionally or translationally, and therefore annotation of the target genes of interesting miRNAs appears important.

There are multiple options to identify target genes of miRNAs. For example, Brock et al proposed a pipeline for miRNA target analysis with R packages "`targetscan.Mm.eg.db`", "`microRNA`" and "`org.Mm.eg.db`". In the example shown below, an integrated package "`SpidermiR`" is adopted, which provides both validated and predicted target genes from multiple databases or software tools including mirWalk [18], miR2Disease [19], miRTar [20], miRTarBase [21], miRandola [22], Pharmaco-miR [23], DIANA [24], Miranda [25], PicTar [26], and TargetScan [27]. It can also retrieve and visualize the gene networks. The following commands give an example of target gene determination for some interesting miRNAs, e.g., the top significant five miRNAs with expression difference between groups (*see* **Note 1**). The potential targets of these miRNAs will be predicted with `SpidermiRdownload_miRNAprediction` and exported to `mirnaTar`.

```
> tT[selected,]$Name[1:5]
> mirna <-
c('hsa-miR-4429','hsa-miR-1827','hsa-miR-5002-5p','hsa-miR-
5187-3p','hsa-miR-4455')
> mirnaTar <- SpidermiRdownload_miRNAprediction(mirna_list=-
mirna)
```

The data frame of `mirnaTar` can be checked with `head(mirnaTar)`, and there are two columns, `V1` showing miRNA names and `V2` listing the target genes.

Note that `SpidermiRdownload_miRNAprediction` gave the prediction targets of four tools: DIANA, Miranda, PicTar, and TargetScan. The validated targets could be downloaded from miRTAR and miRwalk with `SpidermiRdownload_miRNAvalidate` function.

3.3.2 Network and Gene Set Enrichment Analysis

Network analysis and visualization can show not only the shared targets of multiple miRNAs, but also the interactions and pathways among the target genes. There are many tools developed for network building and visualization, e.g., user-friendly interfaced tool Cytoscape [28], R package SpidermiR [15]. Here, we use Cytoscape to construct the regulatory network between the miRNAs (top significant 5) and their predicted targets (50 for each miRNA), since Cytoscape is quite straightforward and particularly useful for network construction with user-customized interactions (Fig. 3a) (*see* **Note 2**). GeneMANIA curates validated and predicted networks between genes from a variety of species [29]. The network types include coexpression, colocalization, genetic interactions, pathway, physical interactions, shared protein domains, and predicted interactions. GeneMANIA also provides a webserver to implement the network construction. SpidermiR can download the interaction data from GeneMANIA and visualize the networks among the user-customized genes, and the functions are still being debugged and updated. Here, we directly use the GeneMANIA prediction server (http://genemania.org/) to construct the pathway network of miRNA target genes (Fig. 3b) (*see* **Note 3**).

Besides the network analysis, statistics-based gene set enrichment analysis (GSEA) should be done for the miRNAs and miRNA targets, so as to find biological meanings and help increase the statistical power through aggregating the signal across groups of related genes. `GOstats` and a number of other R/Bioconductor packages (e.g., `GeneAnswers` [30]) can make the enrichment analysis with hypergeomtric tests (`hyperGTest` function for `GOstats`). As an example, we use `GOstats` to make GO enrichment analysis (Biological Process) to the predicted target genes of the top 5 miRNAs (*see* **Note 4**).

```
> library("org.Hs.eg.db")
> library("GSEABase")
> library("GOstats")
> mirTarget <- mirnaTar$V2
> goAnn <- get("org.Hs.egGO")
> universe <- Lkeys(goAnn)
> entrezIDs <- mget(mirTarget, org.Hs.egSYMBOL2EG, ifnotfound=NA)
> entrezIDs <- as.character(entrezIDs)
```

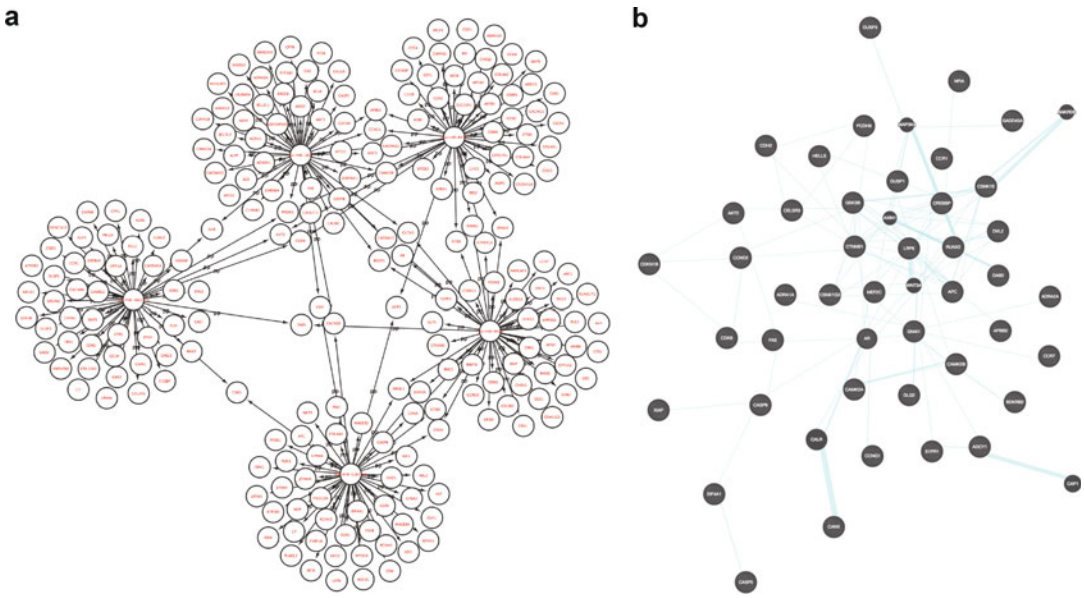

Fig. 3 Interaction networks among miRNAs and their targets. (**a**) Regulatory network between miRNAs and target genes. (**b**) Pathway sub-network among the miRNA target genes

```
> params <- new("GOHyperGParams",
+ geneIds=entrezIDs,
+ universeGeneIds=universe,
+ annotation="org.Hs.eg.db",
+ ontology="BP",
+ pvalueCutoff=0.01,
+ conditional=FALSE,
+ testDirection="over")
> goET <- hyperGTest(params)
> library(Category)
> genelist <- geneIdsByCategory(goET)
> genelist <- sapply(genelist, function(.ids) {
+ .sym &lt;- mget(.ids, envir=org.Hs.egSYMBOL, ifnotfound=NA)
+ .sym[is.na(.sym)] &lt;- .ids[is.na(.sym)]
+ paste(.sym, collapse=";")
+ })
> GObp <- summary(goET)
> GObp$Symbols <- genelist[as.character(GObp$GOBPID)]
> head(GObp)
```

KEGG enrichment can also be performed:

```
> keggAnn <- get("org.Hs.egPATH")
> universe <- Lkeys(keggAnn)
> params <- new("KEGGHyperGParams",
+ geneIds=entrezIDs,
```

```
+ universeGeneIds=universe,
+ annotation="org.Hs.eg.db",
+ categoryName="KEGG",
+ pvalueCutoff=0.01,
+ testDirection="over")
> keggET <- hyperGTest(params)
> kegg <- summary(keggET)
> library(Category)
> genelist <- geneIdsByCategory(keggET)
> genelist <- sapply(genelist, function(.ids) {
+ .sym &lt;- mget(.ids, envir=org.Hs.egSYMBOL, ifnotfound=NA)
+ .sym[is.na(.sym)] &lt;- .ids[is.na(.sym)]
+ paste(.sym, collapse=";")
+ })
> kegg$Symbols <- genelist[as.character(kegg$KEGGID)]
> head(kegg)
```

4 Notes

1. For illustration convenience, the top five miRNAs are selected for target analysis. In practice, all the meaningful miRNAs should be analyzed for targets. For target prediction, multiple prediction tools should be combined and the intersected set will be selected for further analysis if the number of prediction results is large.
2. Cytoscape can be downloaded from http://www.cytoscape.org. There is a detailed manual demonstrating how to install and use the tool. To visualize the interaction network of miRNAs and their target genes, a two-column table is prepared in which the first column records miRNAs and the second records the corresponding targets. Directly import the interaction table to Cytoscape, indicate the interaction sources and targets, and then draw the network with directions.
3. GeneMANIA curates several categories of gene interaction databases, and the database(s) can be selected in the server for network prediction. In the GeneMANIA prediction webserver (http://genemania.org), simply copy the gene symbols (one per line) into the input area, select the desired database(s) and run prediction.
4. Besides `GOstats`, there are also other R packages making Gene Set Enrichment Analysis (GSEA). Chapter 3 in this book can be referred to, which gives a comprehensive introduction on the methods and related packages. The website of Gene Ontology Consortium (http://geneontology.org) also presents an online GO enrichment analysis tool, and it would be an easy choice.

References

1. Kozomara A, Griffiths-Jones S (2014) miRBase: annotating high confidence microRNAs using deep sequencing data. Nucleic Acids Res 42(Database issue):D68–D73. https://doi.org/10.1093/nar/gkt1181
2. McCall MN, Kim MS, Adil M, Patil AH, Lu Y, Mitchell CJ, Leal-Rojas P, Xu J, Kumar M, Dawson VL, Dawson TM, Baras AS, Rosenberg AZ, Arking DE, Burns KH, Pandey A, Halushka MK (2017) Toward the human cellular microRNAome. Genome Res. https://doi.org/10.1101/gr.222067.117
3. Otto T, Candido SV, Pilarz MS, Sicinska E, Bronson RT, Bowden M, Lachowicz IA, Mulry K, Fassl A, Han RC, Jecrois ES, Sicinski P (2017) Cell cycle-targeting microRNAs promote differentiation by enforcing cell-cycle exit. Proc Natl Acad Sci U S A 114 (40):10660–10665. pii 201702914. https://doi.org/10.1073/pnas.1702914114
4. Gao L, Jiang F (2016) MicroRNA (miRNA) profiling. Methods Mol Biol 1381:151–161
5. Huber W, Carey VJ, Gentleman R, Anders S, Carlson M, Carvalho BS, Bravo HC, Davis S, Gatto L, Girke T, Gottardo R, Hahne F, Hansen KD, Irizarry RA, Lawrence M, Love MI, MacDonald J, Obenchain V, Oleś AK, Pagès H, Reyes A, Shannon P, Smyth GK, Tenenbaum D, Waldron L, Morgan M (2015) Orchestrating high-throughput genomic analysis with Bioconductor. Nat Methods 12(2):115–121. https://doi.org/10.1038/nmeth.3252
6. Wickham H, Chang W (2017) devtools: tools to make developing R packages easier. R package version 1.13.3. https://CRAN.R-project.org/package=devtools.
7. Falcon S, Gentleman R (2007) Using GOstats to test gene lists for GO term association. Bioinformatics 23(2):257–258
8. Davis S, Meltzer PS (2017) GEOquery: a bridge between the Gene Expression Omnibus (GEO) and BioConductor. Bioinformatics 23 (14):1846–1847
9. Warnes GR, Bolker B, Bonebakker L, et al. (2016) gplots: various R programming tools for plotting data. R package version 3.0.1. https://CRAN.R-project.org/package=gplots.
10. Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK (2015) limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Res 43 (7):e47
11. Scrucca L, Fop M, Murphy TB, Raftery AE (2016) mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. R J 8(1):289–317
12. Pajak M, Simpson TI (2016) miRNAtap: miRNAtap: microRNA targets – aggregated predictions. R package version 1.8.0.
13. Pajak M, Simpson TI (2016) miRNAtap.db: data for miRNAtap. R package version 0.99.10.
14. Allaire JJ, Gandrud C, Russell K, Yetman CJ (2017) networkD3: D3 JavaScript network graphs from R. R package version 0.4. https://CRAN.R-project.org/package=networkD3.
15. Cava C, Colaprico A, Bertoli G, Graudenzi A, Silva TC, Olsen C, Noushmehr H, Bontempi G, Mauri G, Castiglioni I (2017) SpidermiR: an R/bioconductor package for integrative analysis with miRNA data. Int J Mol Sci 18(2.): pii: E274). https://doi.org/10.3390/ijms18020274
16. Almende BV, Thieurmel B, Robert T (2017) visNetwork: network visualization using 'vis.js' library. R package version 2.0.1. https://CRAN.R-project.org/package=visNetwork.
17. Zhang F, Xu Y, Shugart YY, Yue W et al (2015) Converging evidence implicates the abnormal microRNA system in schizophrenia. Schizophr Bull 41(3):728–735
18. Dweep H, Sticht C, Pandey P, Gretz N (2011) miRWalk--database: prediction of possible miRNA binding sites by "walking" the genes of three genomes. J Biomed Inform 44 (5):839–847
19. Jiang Q, Wang Y, Hao Y, Juan L, Teng M, Zhang X, Li M, Wang G, Liu Y (2009) miR2Disease: a manually curated database for microRNA deregulation in human disease. Nucleic Acids Res 37(Database issue):D98–104
20. Hsu JB, Chiu CM, Hsu SD, Huang WY, Chien CH, Lee TY, Huang HD (2011) miRTar: an integrated system for identifying miRNA-target interactions in human. BMC Bioinformatics 12:300. https://doi.org/10.1186/1471-2105-12-300
21. Hsu SD, Lin FM, Wu WY, Liang C, Huang WC, Chan WL, Tsai WT, Chen GZ, Lee CJ, Chiu CM, Chien CH, Wu MC, Huang CY, Tsou AP, Huang HD (2011) miRTarBase: a database curates experimentally validated microRNA-target interactions. Nucleic Acids Res 39(Database issue):D163–D169. https://doi.org/10.1093/nar/gkq1107

22. Russo F, Di Bella S, Nigita G, Macca V, Laganà A, Giugno R, Pulvirenti A, Ferro A (2012) miRandola: extracellular circulating microRNAs database. PLoS One 7(10): e47786. https://doi.org/10.1371/journal.pone.0047786
23. Rukov JL, Wilentzik R, Jaffe I, Vinther J, Shomron N (2014) Pharmaco-miR: linking microRNAs and drug effects. Brief Bioinform 15(4):648–659. https://doi.org/10.1093/bib/bbs082
24. Maragkakis M, Reczko M, Simossis VA, Alexiou P, Papadopoulos GL, Dalamagas T, Giannopoulos G, Goumas G, Koukis E, Kourtis K, Vergoulis T, Koziris N, Sellis T, Tsanakas P, Hatzigeorgiou AG (2009) DIANA-microT web server: elucidating microRNA functions through target prediction. Nucleic Acids Res 37(Web Server issue): W273–W276
25. John B, Enright AJ, Aravin A, Tuschl T, Sander C, Marks DS (2004) Human MicroRNA targets. PLoS Biol 2(11):e363
26. Krek A, Grün D, Poy MN, Wolf R, Rosenberg L, Epstein EJ, MacMenamin P, da Piedade I, Gunsalus KC, Stoffel M, Rajewsky N (2005) Combinatorial microRNA target predictions. Nat Genet 37(5):495–500
27. Agarwal V, Bell GW, Nam J, Bartel DP (2015) Predicting effective microRNA target sites in mammalian mRNAs. eLife 4:e05005
28. Saito R, Smoot ME, Ono K, Ruscheinski J, Wang PL, Lotia S, Pico AR, Bader GD, Ideker T (2012) A travel guide to cytoscape plugins. Nat Methods 9(11):1069–1076. https://doi.org/10.1038/nmeth.2212
29. Montojo J, Zuberi K, Rodriguez H, Bader GD, Morris Q (2014) GeneMANIA: fast gene network construction and function prediction for cytoscape. F1000Res 3(153). https://doi.org/10.12688/f1000research.4572.1. eCollection 2014
30. Feng G, Shaw P, Rosen ST, Lin SM, Kibbe WA (2012) Using the bioconductor GeneAnswers package to interpret gene lists. Methods Mol Biol 802:101–112. https://doi.org/10.1007/978-1-61779-400-1_7