REVIEW ARTICLE

Review on Applications of R programming in Biological data analysis

Selva Babu S1 , Kiruthika A1 , Ragapriya V1 , Monika M I1 , Anandhi V2 and Saranya N1*

[1]Department of Plant Molecular Biology and Bioinformatics Centre for Plant Molecular Biology and Biotechnology Tamil Nadu Agricultural University, Coimbatore -3

[2]Department of Physical Sciences & Information Technology, AEC&RI, TNAU, Coimbatore-641003

ABSTRACT

Received: 04 April 2022

Revised: 11 April 2022

Revised: 05 May 2022

Accepted: 24 May 2022

The key programming language used in the field of data science is R programming, because of the free availability and ability to handle huge data. Packages developed in the R are used widely by scientists to obtain solutions for various real-world problems in several disciplines like healthcare, agriculture, and information technology. Next-generation sequencing data requires various bioinformatics analyses like quality checking, differential gene expression studies, and annotation which is facilitated by R packages and software. Bioconductorprojectremains a hub of R packages that are open access and move the researchers to the comfort zone for various kinds of analysis. In this review, we briefly discussed various packages useful for NGS data analysis and also explained how to use the packages for basic level analysis that benefits researchers having less exposure to R programming.

Keywords: *NGS analysis; Bioconductor; R; Biological data; NGS Reports*

INTRODUCTION

R was designed by Ross Ihaka and Robert Gentleman in 1993 (https://www.r-project.org.html) at the University of Auckland, New Zealand and is currently developed by the R development core team. It was developed at Bell Laboratories and is an open-source programming language and also case-sensitive. It is extensively used by software programmers, statisticians, data scientists, and data miners. It has numerous applications in domains like healthcare, academics, consulting, finance, media, and many more. Its vast applicability in statistics, data visualization, and machine learning have contributed to the analysis and interpretation of biological data from various experimental techniques.

Modern statistical techniques have been implemented. A few of these are built into the base R environment, but many are supplied as packages. There are about 25 packages supplied with R (called “standard” and “recommended” packages) and many more are available through the CRAN (Comprehensive R Archive Network) family of Internet sites (https://CRAN.R-project.org).

Importance of R programming

R programming language is not only a statistical package but also allows us to integrate with other languages (C, C++). Objects, functions, and packages can easily be created by R. Since R is much similar to other widely used languages syntactically, it is easier to code and learn in R. Programs can be written in R in any of the widely used IDE like R Studio, Rattle, Tinn-R, etc. R programming language is suitable for GNU/Linux and Windows operating systems.

R is becoming the most widely used software in bioinformatics. In life sciences especially in bioinformatics, R has been frequently used for statistical analysis of biological data from various experiments like microarray, RNA-Seq, ChIP-Seq, whole genome sequencing, small RNA-seq, single-cell RNA sequencing, etc, and also for data visualizations to create high quality multi-dimensional interactive graphs and plots. For example, R can be used for building co-expression networks between genes using their expression values which can reveal many interaction pathways that give insight into the function of genes altogether. In such cases, correlation networks or weighted correlation networks are very helpful. In this chapter, we will discuss some of the R packages that are developed for handling biological data and its applications.

*Corresponding author’s e-mail: saranya.n@tnau.ac.in

R Packages for biological data analysis

A glimpse of some of the packages chosen randomly among plenty of available R packages are explained here, which will provide an insight into the applicability of R towards biological data. Many R packages developed for handling biological data are available through Bioconductor(https://bioconductor.org/) and GitHub(https://github.com) repositories.

Bioconductor Package

The Bioconductor (https://bioconductor.org/) remains a hub of thousands of R packages for performing statistical analysis emerging through high-throughput biological assays. All the packages within Bioconductor are committed to open source, collaborative, distributed software development and literate, reproducible research. Algorithms remain fully accessible to the scientific user community and can be edited, and manipulated according to their needs. The Bioconductor project started in 2001 and is overseen by a core team, based primarily at Roswell Park Comprehensive Cancer Center, and by other members coming from US and international institutions. Currently, there are 2083 software packages available in Bioconductor release 3.14.

Bioconductor packages are organized into workflows for the analysis of oligonucleotide arrays, sequence analysis, flow cytometry, and other high-throughput genomic data. These workflows remain the compilation of various R packages that are integrated sequentially in order to carry out the statistical and graphical analysis on the high-throughput biological data. Under each workflow, the packages are ranked based on their applicability and feasibility so that the user can choose their appropriate packages for analysis.

For example, RNA-seq 1-2-3 workflow (Law *et al*., 2016) benchmarks three packages such as(https://bioconductor.org/packages/release/workflows/html/RNAseq123.html) edgeR package (Robinson *et al*.,2010) (to import, organize, filter and normalize the RNA-seq data), followed by the limma package (Ritchie *et al*., 2015) (to assess the differential expression and perform gene set testing) and then Glimma package (Su *et al*.,2017) (for interactive exploration of the enriched genes).

Bioconductor packages also integrate and associate with various biological databases such as GenBank (Benson *et al*., 2005), Entrez genes (Maglott *et al*., 2010) and PubMed (Wheeler *et al*., 2004) etc for easy retrieval, analysis, and functional annotation of the biological data.

ngsReports

The ngsReports package (Ward *et al*., 2020)is available at Bioconductor (https://bioconductor.org/packages/release/bioc/html/ngsReports.html) and the GUI shiny app (https://github.com/UofABioinformaticsHub/shinyNgsreports). This package performs quality checking of raw and processed data obtained from high throughput sequencing pipelines. ngsReports can be used for quality checking and the graphical visualization of the generated reports which is the first step in high throughput sequencing data analysis. Visualization can be carried out across many samples using default, highly customizable plots with options to perform hierarchical clustering to quickly identify outlier libraries. ngsReports also generates HTML reports for ease of analysis. The following parameters are checked during the quality checking process:

- Summary (The PASS/WARN/FAIL status of each following parameters)
- Basic_Statistics
- Per_base_sequence_quality
- Per_sequence_quality_scores
- Per_base_sequence_content
- Per_sequence_GC_content
- Per_base_N_content
- Sequence_Length_Distribution
- Sequence_Duplication_Levels
- Overrepresented_sequences

Steps involved in the ngsReports R package to generate a FASTQC plot using plant transcriptome data are given as an example.

ngsReports

ngsReports package is an R package mainly used to produce the combined plots using the multiple reports produced for the different libraries. Sample from FastQC (Andrews 2010), a Java-based tool and to parse the log files produced from several NGS tools like STAR, hisat2, bowtie, trimmomatic, cutadapt, BUSCO, quast. The combined FastQC file is useful to visualize the quality of the different libraries of the same sample and different libraries of the different samples.

ngsReports package requires R (>= 4.1.0), BiocGenerics (Huber *et al*., 2015), ggplot2 (>= 3.3.5) (Wickham, 2011) and tibble (>= 1.3.1) (Mailund, 2019). Plots created by ngsReportsare standard ggplot2 objects or interactive objects by using plotly

Example workflow

1. Installation of ngsReports:

Command to install the ngsReports package using R

```
>if (!requireNamespace("BiocManager", quietly = TRUE))
```

```
>install.packages("BiocManager")
```

This *install.packages()* is a function typically used to download the packages in R from cran library or from other local files. In this first step, BiocManager is installed by using the BiocManager, ngsReport package has to be installed in R by using the following command,

```
>BiocManager::install("ngsReports")
```

After installation of the ngsRports, this package to be loaded in the R working environment in order use this package, command to simply load the package in R,

```
>library(ngsReports)
```

2. Data for the analysis

Trasncriptome data of Nagina 22 an indica rice variety is used in this example analysis. Paired-end sequenced data such as SRR15216084, SRR15234654, and SRR15234669 downloaded from the SRA database of the NCBI (https://www.ncbi.nlm.nih.gov/sra/?term=SRR15216084).

3. Quality checking using FastQC software

FastQC, a standalone java-based tool is used to check the quality of the data. This FastQC can be run in user interface or in command line, to check the quality of the data, if user interface is used to check the quality, file has to be saved in html format, the output folder will contain html file and zip files of each library. In case the FastQC is run in the command line, a single command will automatically save these files in the desired output folder. These zip files saved in the output directory are the input file for the ngsReports package.

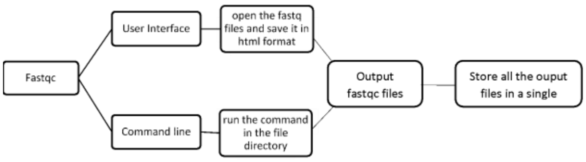
Fastqc
User Interface
open the fastq files and save it in html format
Command line
run the command in the file directory
Output fastqc files
Store all the ouput files in a single

Figure 1: FastQC workflow

Command for FastQC

```
>FastQC *.fastq.gz
```

(* denotes all the files with format fastq.gz files in the working directory)

1. Plotting the qc report produced by FastQC using ngsReports

Load the library:

```
>library(ngsReports)
```

File directory of the FASTQC files must be set and these files are taken for generating reports:

```
>setwd("path/to/FastQCfiles/")
```

```
>fileDir<- system.file("filepath")
```

This filedir variable holds the path of the FastQC files

```
>files<- list.files(fileDir, pattern = "FastQC.zip$", full.names = TRUE)
```

Now variable files holds the FastQC files which are available in the specified path

```
>fdl<- FastQCDataList(files)
```

Variable fdl has the FastQC files list

Total reads of the input file will be shown as ouput:

```
reads<- readTotals(fdl)
```

```
      # A tibble: 4 x 2
       Filename          Total_Sequences
      <chr><int>
      1 SRR15234654_1.FastQ      16016834
      2 SRR15234654_2.FastQ      16016834
      3 SRR15234669_1.FastQ      19084322
     4 SRR15234669_2.FastQ      19084322
     5 SRR15216084_1.FastQ      18128354
     6 SRR15216084_2.FastQ      18128354
```

To plot the summary of the FastQ files quality

```
>plotSummary(fdl)
```

This command will plot the full information of each quality checking category of all the libraries with four different colours. Explanation of the colors will be auto displayed as legends.

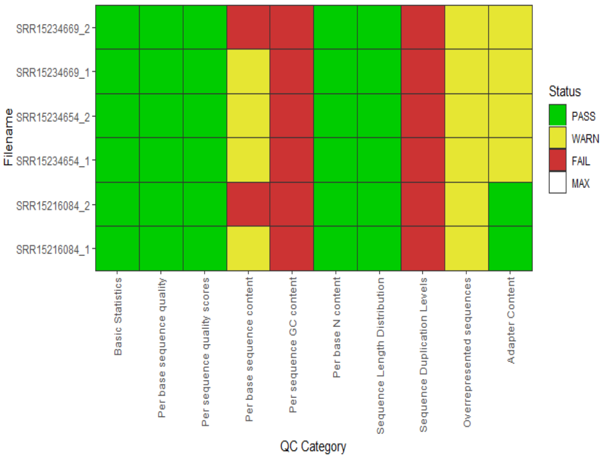

Fig 2. Summary of quality check report produced by ngsReports

>plotReadTotals(fdl)

Total reads can be generated as a graph which shows the unique and duplicated number of reads in the FastQfiles:

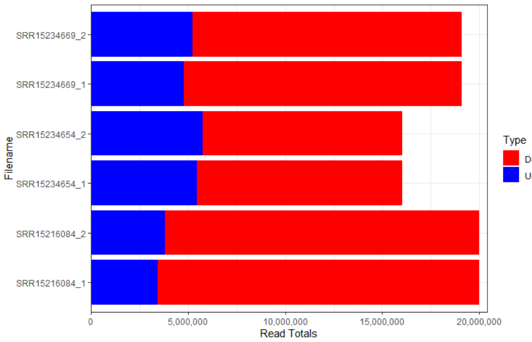

Fig 3. Chart describes the number of unique and duplicate reads

>plotBaseQuals(fdl[1:4], plotType = "boxplot")

Command to check the quality of each base, this is the most important qc check of the FastQfiles.

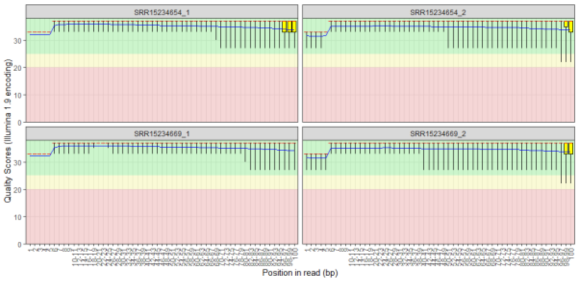

Fig 4.Combined per base sequence quality chart of different samples

>plotBaseQuals(fdl[1])

To check the per base sequence quality of the individual library,

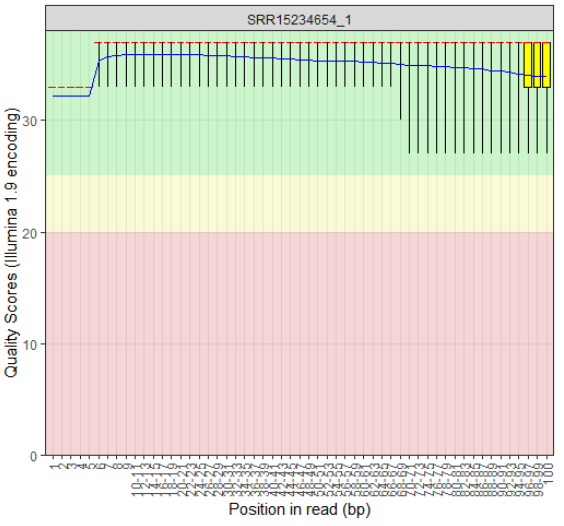

Fig. 5. Per base sequence quality chart of individual samples

Table 1: There are several other options available to check the quality of each category separately, commands and uses of the command are listed in the below table.

Command	Use
>plotBaseQuals(fdl)	Plots the heatmap of mean quality score at each position for multiple FastQC reports
>plotSeqQuals(fdl)	Plots the mean sequence quality as heatmap
>plotSeqContent(fdl)	Plots the heatmap of per base sequence content with individual base colour, A,C, G and T being represented by green, blue, black and red
>plotAdapterContent(fdl)	Plots the adapter content in the library
>plotDupLevels(fdl)	Plots the Sequence Duplication Levels
>plotGcContent(fdl)	Plots the GC content as heatmap for all the library
>plotOverrep(fdl)	Plots the Overrepresented sequences

Writing the outputs

After plotting the data each data can be individually exported in different formats like pdf, image, and HTML formats.

ggbio

ggbio (Yin *et al*., 2012) is a R based bioconductor package to visualize and explore genomics annotations and high-throughput data. This package has wider application in the creation of both typical and non-typical biological plots for genomic data generated from core bioconductor data structures by either the high-level autoplot function, or the combination of low-level components of the grammar of graphics. The ggbio R package is available at http://www.bioconductor.org/packages/2.11/bioc/html/ggbio.html. ggbio plots provide high quality publication figures that can be used for visualization of genomic regions, summary views of sequence alignments and splicing patterns, and genome-wide overviews with karyogram, circular and grand linear layouts. Bioconductor data structures support the generation of plots within a specified modular framework. Various functionalities applicable with ggplot2 (discussed below) are also supported by the ggbio package.

Gene hummus

geneHummus package (https://github.com/NCBI-Hackathons/GeneHummus) (Die et al., 2019), an R-based pipeline is used for the identification and characterization of plant gene families. Dependencies (*dplyr* [15], *stringr* (Wickham et al., 2019), *rentrez* (Winter, 2017), httr, utils and curl packages) that are required by geneHummus package can be readily downloaded from the CRAN repository (Hornik, 2012). Identification and characterization of unknown genes is performed based on the proteins from the RefSeq database and conserved domain architectures based on SPARCLE. A case study reported on the auxin receptor factor gene (ARF) family in *Cicerarietinum* (chickpea) and other legumes using geneHummus has shown higher performance in the identification of ARF family genes.

Genes identified can be further characterized by downstream analysis such as phylogenetic constructions and gene expression profiles.

Hayai-Annotation Plants

Hayai-Annotation Plants (Ghelfiet *al*., 2019) is an ultra-fast and comprehensive functional gene annotation system in plants using R. This tool is mainly based on sequence-similarity searches, using USEARCH against UniProtKB (taxonomy Embryophyta), with 5 level of functional annotation step which includes: i) protein name; ii) gene ontology terms consisting of its three main domains (Biological Process, Molecular Function and Cellular Component); iii) enzyme commission number; iv) protein existence level; and v) evidence type. Hayai-Annotation mainly aims to increase the GO (Gene ontology) and EC (Enzyme commission) annotation assignments based on the Protein Existence Level algorithm. This program can be installed locally and run on a local machine without the dependency on public websites.

WGCNA: an R package for weighted correlation network analysis

The WGCNA R package (Langfelderet *al*., 2008) has been widely used for weighted correlation network analysis (WGCNA) of large, high-dimensional data sets such as gene expression profiles, image data, genetic marker data, proteomics data etc. Functions in the WGCNA package can be divided into the following categories: 1. network construction; 2. module detection; 3. module and gene selection; 4. calculations of topological properties; 5. data simulation; 6. visualization; 7. interfacing with external software packages. System-level analysis of correlation patterns among genes across microarray samples is made possible through weighted gene co-expression network analysis.

Di Leo *et al* 2011reported the application of WGCNA to tomato metabolomics data and identified three major modules of metabolites that were associated with ripening-related traits and genetic background where WGCNA performance was found to be more significant compared to more common statistical methods such as PCA and BL-SOM

ggplot2

ggplot2 (https://github.com/hadley/ggplot2) (Wickham, 2011) is a plotting package that can be used to create complex plots from data in a data frame with publication quality. Many of the packages in bioconductor implement ggplot2 for data visualization which ranges from bar plot to a scatterplot. Data, aesthetics, and geometry are the three different fundamental parts of ggplot2. Flexibility and customization of the plots are the key features that render wide applicability of this package in biological data analysis.

Conclusion

R being an open-source programming language encourages scientists to modulate, manipulate and develop new technology to analyze the exponentially growing biological data obtained from various experimental platforms. There remains a lot of scopes to develop new packages (for example count table creation based on RNA-seq data) and view the biological data on multidimensional scale which will open new horizons for the molecular level understanding of the biological system.

Funding and Acknowledgment

All the authors acknowledge the bioinformatics lab facility at the Department of Plant molecular Biology and Bioinformatics, Centre for Plant Molecular Biology and Biotechnology, Tamil Nadu Agricultural University, Coimbatore.

Ethics statement

No specific permits were required for the described field studies because no human or animal subjects were involved in this research.

Originality and plagiarism

This is review article of R packages for biological data analysis and any work and/or words of others, has been appropriately cited

Consent for publication

All the authors agreed to publish the content.

Competing interests

There was no conflict of interest in the publication of this content.

Data availability

All the data of this manuscript are included in the MS. No separate external data source is required. If anything is required from the MS, certainly, this will be extended by communicating with the corresponding author through corresponding official mail; saranya.n@tnau.ac.in

Author contributions

Idea conceptualization and Guidance – SN, AV, Experiments - SS, KA, RV, Writing original draft and revision – SS, SN, MMI;

REFERENCES

A Grammar of Data Manipulation [R package dplyr version 0.8.0.1]. https://cran.r-project.org/web/packages/dplyr/index.html. Accessed 23 Mar 2019.

Andrews, S. 2010. FastQC: A Quality Control Tool for High Throughput Sequence Data [Online]. at: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., and Wheeler, D. L. 2005. GenBank. *Nucleic Acids Res.*, ***33*****(suppl_1)**: D34-D38.

Die, J. V., Elmassry, M. M., LeBlanc, K. H., Awe, O. I., Dillman, A. and Busby, B. 2019. geneHummus: an R package to define gene families and their expression in legumes and beyond. *BMC genomics*, ***20*****(1)**: 1-9.

DiLeo, M. V., Strahan, G. D., den Bakker, M. and Hoekenga, O. A. 2011. Weighted correlation network analysis (WGCNA) applied to the tomato fruit metabolome. *PLoS One.*, ***6*****(10)**: e26683.

Ghelfi, A., Shirasawa, K., Hirakawa, H. and Isobe, S. 2019. Hayai-Annotation Plants: an ultra-fast and comprehensive functional gene annotation system in plants. *Bioinformatics.*, ***35*****(21)**: 4427-4429.

Hornik, K. 2012. The comprehensive R archive network. *Wiley interdisciplinary reviews: Comput. Stat.*, ***4*****(4)**: 394-398.

Huber, W., Carey, V. J., Gentleman, R., Anders, S., Carlson, M., Carvalho, B. S. and Morgan, M. 2015. Orchestrating high-throughput genomic analysis with Bioconductor. *Nat. Methods.*, ***12*****(2)**: 115-121.

Langfelder, P. and Horvath, S. 2008. WGCNA: an R package for weighted correlation network analysis. *BMC Bioinform.*, ***9*****(1)**: 1-13.

Law, C. W., Alhamdoosh, M., Su, S., Dong, X., Tian, L., Smyth, G. K., and Ritchie, M. E. 2016. RNA-seq analysis is easy as 1-2-3 with limma, Glimma and edgeR. *F1000.*, ***5***.

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. 2010. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics.*, **26(1)**: 139-140.

Ritchie, M. E., Phipson, B., Wu, D. I., Hu, Y., Law, C. W., Shi, W., and Smyth, G. K. 2015. limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res.*, ***43*****(7)**: e47-e47.

Su, S., Law, C. W., Ah-Cann, C., Asselin-Labat, M. L., Blewitt, M. E., & Ritchie, M. E. 2017. Glimma: interactive graphics for gene expression analysis. *Bioinformatics.*, ***33*****(13)**: 2050-2052.

Maglott, D., Ostell, J., Pruitt, K. D., &Tatusova, T. 2010. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Res.*, ***39*****(suppl_1)**: D52-D57.

Mailund T. 2019 Representing Tables: tibble. In: R Data Science Quick Reference. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-4894-2_3

Wheeler, D. L., Church, D. M., Edgar, R., Federhen, S., Helmberg, W., Madden, T. L., and Wagner, L. 2004. Database resources of the National Center for Biotechnology Information: update. *Nucleic Acids Res.*, ***32*****(suppl_1)**: D35-D40.

Ward, C. M., To, T. H. and Pederson, S. M. 2020. ngsReports: a Bioconductor package for managing FastQC reports and other NGS related log files. *Bioinformatics.*, ***36*****(8)**: 2587-2588.

Wickham, H. 2011. ggplot2. *Wiley Interdisciplinary Reviews:Comput. Stat.*, ***3*****(2)**: 180-185.

Yin, T., Cook, D. and Lawrence, M. 2012. ggbio: an R package for extending the grammar of graphics for genomic data. *Genome Biol.*, ***13*****(8)**: 1-14.

Wickham, H. and Wickham, M. H. 2019. Package ‘stringr’.

Winter, D. J. 2017. *rentrez: An R package for the NCBI eUtils API* (No. e3179v2). PeerJ Preprints.