systemPipeRdata: NGS workflow templates and sample data

Author: Daniela Cassol (danielac@ucr.edu) and Thomas Girke (thomas.girke@ucr.edu)

Last update: 20 April, 2019

Package

systemPipeRdata 1.11.2

Contents

Note: the most recent version of this vignette can be found here and a short overview slide show here.

Note: if you use `systemPipeR` and `systemPipeRdata` in published research, please cite:

Backman, T.W.H and Girke, T. (2016). *systemPipeR*: NGS Workflow and Report Generation Environment. *BMC Bioinformatics*, 17: 388. 10.1186/s12859-016-1241-0.

1 Introduction

`systemPipeRdata` is a helper package to generate with a single command NGS workflow templates that are intended to be used by its parent package `systemPipeR` (H Backman and Girke 2016). The latter is an environment for building *end-to-end* analysis pipelines with automated report generation for next generation sequence (NGS) applications such as RNA-Seq, Ribo-Seq, ChIP-Seq, VAR-Seq and many others. The directory structure of the workflow templates and the sample data used by `systemPipeRdata` are described here.

2 Getting Started

2.1 Installation

The R software for using `systemPipeRdata` can be downloaded from CRAN. The `systemPipeRdata` package can be installed from within R as follows:

```r
if (!requireNamespace("BiocManager", quietly = TRUE)) install.packages("BiocManager")
BiocManager::install("systemPipeRdata")  # Installs from Bioconductor once
# available there
BiocManager::install("tgirke/systemPipeR", build_vignettes = TRUE,
    dependencies = TRUE)  # Installs from github
```

2.2 Loading package and documentation

```r
library("systemPipeRdata")  # Loads the package
```

```r
library(help = "systemPipeRdata")  # Lists package info
vignette("systemPipeRdata")  # Opens vignette
```

2.3 Generate workflow template

Load one of the available NGS workflows into your current working directory. The following does this for the `varseq` template. The name of the resulting workflow directory can be specified under the `mydirname` argument. The default `NULL` uses the name of the chosen workflow. An error is issued if a directory of the same name and path exists already. Besides, it is possible to choose different version of the workflow template. Please check the available options here, or provide the download URL to your template. The URL can be specified under `url` argument and the file name in the `urlname` argument. The default `NULL` copies the current version available in the systemPipeRdata.

```
genWorkenvir(workflow = "varseq", mydirname = NULL, url = NULL,
    urlname = NULL)
setwd("varseq")
```

On Linux and OS X systems the same can be achieved from the command-line of a terminal with the following commands.

```
$ Rscript -e "systemPipeRdata::genWorkenvir(workflow='varseq', mydirname=NULL, url=NULL, urlname=NULL)"
```

The workflow templates generated by `genWorkenvir` contain the following preconfigured directory structure:

- ***workflow/*** (*e.g.* *rnaseq/*)
    - This is the directory of the R session running the workflow.
    - Run script (**.Rmd* or **.Rnw*) and sample annotation (*targets.txt*) files are located here.
    - Note, this directory can have any name (*e.g.* ***rnaseq***, ***varseq***). Changing its name does not require any modifications in the run script(s).
    - Important subdirectories:
        - ***param/***
            - Stores parameter files such as: **.param*, **.tmpl* and **_run.sh*.
        - ***data/***
            - FASTQ samples
            - Reference FASTA file
            - Annotations
            - etc.
        - ***results/***
            - Alignment, variant and peak files (BAM, VCF, BED)
            - Tabular result files
            - Images and plots
            - etc.

2.4 Run workflows

Next, run from within R the chosen sample workflow by executing the code provided in the corresponding `*.Rmd` template file. If preferred the corresponding `*.Rnw` or `*.R` versions can be used instead. Alternatively, one can run an entire workflow from start to finish with a single command by executing from the command-line `'make -B'` within the workflow directory (here `'varseq'`). Much more detailed information on running and customizing `systemPipeR` workflows is available in its overview vignette here. This vignette can also be opened from R with the following command.

```
library("systemPipeR")  # Loads systemPipeR which needs to be installed via BiocManager::install() from Bioc
```

```
vignette("systemPipeR", package = "systemPipeR")
```

2.5 Return paths to sample data

The location of the sample data provided by `systemPipeRdata` can be returned as a `list`.

```
pathList()
## $targets
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/param/targets.txt"
##
## $targetsPE
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/param/targetsPE.txt"
##
## $annotationdir
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/annotation/"
##
## $fastqdir
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/fastq/"
##
## $bamdir
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/bam/"
##
## $paramdir
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/param/"
##
## $workflows
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/workflows/"
##
## $chipseq
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/workflows/chipseq/"
##
## $rnaseq
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/workflows/rnaseq/"
##
## $riboseq
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/workflows/riboseq/"
##
## $varseq
## [1] "/home/dcassol/R/x86_64-pc-linux-gnu-library/3.7/systemPipeRdata/extdata/workflows/varseq/"
```

3 Version information

```
sessionInfo()
## R Under development (unstable) (2019-04-03 r76310)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 18.04.2 LTS
##
## Matrix products: default
## BLAS:   /usr/local/lib/R/lib/libRblas.so
## LAPACK: /usr/local/lib/R/lib/libRlapack.so
```

```
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [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] stats4    parallel  stats     graphics  grDevices
## [6] utils     datasets  methods   base
##
## other attached packages:
##  [1] systemPipeRdata_1.11.2      systemPipeR_1.17.9
##  [3] ShortRead_1.41.0            GenomicAlignments_1.19.1
##  [5] SummarizedExperiment_1.13.0 DelayedArray_0.9.9
##  [7] matrixStats_0.54.0          Biobase_2.43.1
##  [9] BiocParallel_1.17.18        Rsamtools_1.99.5
## [11] Biostrings_2.51.5           XVector_0.23.2
## [13] GenomicRanges_1.35.1        GenomeInfoDb_1.19.3
## [15] IRanges_2.17.4              S4Vectors_0.21.22
## [17] BiocGenerics_0.29.2         BiocStyle_2.11.0
##
## loaded via a namespace (and not attached):
##  [1] Category_2.49.1           bitops_1.0-6
##  [3] bit64_0.9-7               RColorBrewer_1.1-2
##  [5] progress_1.2.0            httr_1.4.0
##  [7] Rgraphviz_2.27.0          tools_3.7.0
##  [9] backports_1.1.3           R6_2.4.0
## [11] DBI_1.0.0                 lazyeval_0.2.2
## [13] colorspace_1.4-1          withr_2.1.2
## [15] prettyunits_1.0.2         bit_1.1-14
## [17] compiler_3.7.0            graph_1.61.1
## [19] formatR_1.6               rtracklayer_1.43.3
## [21] bookdown_0.9              scales_1.0.0
## [23] checkmate_1.9.1           genefilter_1.65.0
## [25] RBGL_1.59.5               rappdirs_0.3.1
## [27] stringr_1.4.0             digest_0.6.18
## [29] rmarkdown_1.12            AnnotationForge_1.25.0
## [31] pkgconfig_2.0.2           htmltools_0.3.6
## [33] BSgenome_1.51.0           limma_3.39.14
## [35] rlang_0.3.3               RSQLite_2.1.1
## [37] GOstats_2.49.0            hwriter_1.3.2
## [39] VariantAnnotation_1.29.25 RCurl_1.95-4.12
## [41] magrittr_1.5              GO.db_3.7.0
## [43] GenomeInfoDbData_1.2.1    Matrix_1.2-17
## [45] Rcpp_1.0.1                munsell_0.5.0
## [47] stringi_1.4.3             yaml_2.2.0
## [49] edgeR_3.25.3              zlibbioc_1.29.0
## [51] plyr_1.8.4                grid_3.7.0
## [53] blob_1.1.1                 crayon_1.3.4
## [55] lattice_0.20-38            splines_3.7.0
## [57] GenomicFeatures_1.35.9     annotate_1.61.1
## [59] hms_0.4.2                  batchtools_0.9.11
## [61] locfit_1.5-9.1             knitr_1.22
## [63] pillar_1.3.1               rjson_0.2.20
## [65] base64url_1.4              codetools_0.2-16
## [67] biomaRt_2.39.2             XML_3.98-1.19
## [69] evaluate_0.13              latticeExtra_0.6-28
## [71] data.table_1.12.0          BiocManager_1.30.4
## [73] gtable_0.3.0               assertthat_0.2.1
## [75] ggplot2_3.1.0              xfun_0.6
## [77] xtable_1.8-3               survival_2.44-1.1
## [79] tibble_2.1.1               pheatmap_1.0.12
## [81] AnnotationDbi_1.45.1       memoise_1.1.0
## [83] brew_1.0-6                 GSEABase_1.45.0
```

4 Funding

This project was supported by funds from the National Institutes of Health (NIH) and the National Science Foundation (NSF).

References

H Backman, Tyler W, and Thomas Girke. 2016. "systemPipeR: NGS workflow and report generation environment." *BMC Bioinformatics* 17 (1): 388. doi:10.1186/s12859-016-1241-0.