systemPipeRdata: NGS workflow templates and sample data

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

Last update: 24 October, 2018

Package

systemPipeRdata 1.8.0

Contents

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

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:

```
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

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

```
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.

```
genWorkenvir(workflow="varseq", mydirname=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. `{.sh generate_workenvir_from_shell, eval=FALSE} $ Rscript -e "systemPipeRdata::genWorkenvir(workflow='varseq', mydirname=NULL)"`

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

```
workflow_name/ # *.Rnw/*.Rmd scripts and targets file
param/ # parameter files for command-line software
data/ # inputs e.g. FASTQ, reference, annotations
results/ # analysis result files
```

2.4 Run workflows

Next, run from within R the chosen sample workflow by executing the code provided in the corresponding **.Rnw* template file. If preferred the corresponding **.Rmd* 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 from Bioconductor
```

```
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] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/param/targets.txt"
##
## $targetsPE
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/param/targetsPE.txt"
##
## $annotationdir
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/annotation/"
##
## $fastqdir
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/fastq/"
##
## $bamdir
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/bam/"
##
## $paramdir
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/param/"
```

```
##
## $workflows
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/workflows/"
##
## $chipseq
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/workflows/chipseq/"
##
## $rnaseq
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/workflows/rnaseq/"
##
## $riboseq
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/workflows/riboseq/"
##
## $varseq
## [1] "/rhome/dcassol/R/x86_64-pc-linux-gnu-library/3.5/systemPipeRdata/extdata/workflows/varseq/"
```

3 Version information

```
sessionInfo()
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: CentOS Linux 7 (Core)
##
## Matrix products: default
## BLAS: /usr/lib64/libblas.so.3.4.2
## LAPACK: /usr/lib64/liblapack.so.3.4.2
##
## locale:
## [1] C
##
## attached base packages:
## [1] stats4    parallel  stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] systemPipeRdata_1.8.0       systemPipeR_1.13.0          ShortRead_1.38.0
##  [4] GenomicAlignments_1.16.0    SummarizedExperiment_1.10.1 DelayedArray_0.6.5
##  [7] matrixStats_0.54.0          Biobase_2.40.0              BiocParallel_1.14.2
## [10] Rsamtools_1.32.3            Biostrings_2.48.0           XVector_0.20.0
## [13] GenomicRanges_1.32.3        GenomeInfoDb_1.16.0         IRanges_2.14.10
## [16] S4Vectors_0.18.3            BiocGenerics_0.26.0         BiocStyle_2.8.2
##
## loaded via a namespace (and not attached):
##  [1] Category_2.46.0        bitops_1.0-6           bit64_0.9-7            RColorBrewer_1.1-2
##  [5] progress_1.2.0         httr_1.3.1             rprojroot_1.3-2        Rgraphviz_2.24.0
##  [9] tools_3.5.0            backports_1.1.2        R6_2.3.0               DBI_1.0.0
## [13] lazyeval_0.2.1         colorspace_1.3-2       withr_2.1.2            tidyselect_0.2.4
## [17] prettyunits_1.0.2      bit_1.1-14             compiler_3.5.0         graph_1.58.0
## [21] rtracklayer_1.40.6     bookdown_0.7           checkmate_1.8.5        scales_1.0.0
```

```
## [25] genefilter_1.62.0      RBGL_1.56.0            rappdirs_0.3.1         stringr_1.3.1
## [29] digest_0.6.18          rmarkdown_1.10         AnnotationForge_1.22.2 pkgconfig_2.0.2
## [33] htmltools_0.3.6        limma_3.36.5           rlang_0.2.1            RSQLite_2.1.1
## [37] bindr_0.1.1            GOstats_2.46.0         hwriter_1.3.2          dplyr_0.7.6
## [41] RCurl_1.95-4.10        magrittr_1.5           GO.db_3.6.0            GenomeInfoDbData_1.1.0
## [45] Matrix_1.2-14          Rcpp_0.12.19           munsell_0.5.0          stringi_1.2.4
## [49] yaml_2.2.0             edgeR_3.22.5           debugme_1.1.0          zlibbioc_1.26.0
## [53] plyr_1.8.4             grid_3.5.0             blob_1.1.1             crayon_1.3.4
## [57] lattice_0.20-35        splines_3.5.0          GenomicFeatures_1.32.3 annotate_1.58.0
## [61] hms_0.4.2              batchtools_0.9.11      locfit_1.5-9.1         knitr_1.20
## [65] pillar_1.3.0           rjson_0.2.20           base64url_1.4          codetools_0.2-15
## [69] biomaRt_2.36.1         XML_3.98-1.11          glue_1.3.0             evaluate_0.11
## [73] latticeExtra_0.6-28    data.table_1.11.8      gtable_0.2.0           purrr_0.2.5
## [77] assertthat_0.2.0       ggplot2_3.0.0          xfun_0.3               xtable_1.8-3
## [81] survival_2.42-6        tibble_1.4.2           pheatmap_1.0.10        AnnotationDbi_1.42.1
## [85] memoise_1.1.0          bindrcpp_0.2.2         brew_1.0-6             GSEABase_1.42.0
```

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.