Runtime benchmark

Fabricio Almeida-Silva [1] ***and Yves Van de Peer*** [1]

[1]VIB-UGent Center for Plant Systems Biology, Ghent University, Ghent, Belgium

9 February 2024

Contents

1 Introduction

Here, we will perform a runtime benchmark for functions related to duplicate classification and substitution rates calculation using model organisms.

To start, let's load the required data and packages.

```
set.seed(123) # for reproducibility

# Load required packages
library(doubletrouble)
library(here)
## here() starts at /home/faalm/Dropbox/package_benchmarks/doubletrouble_paper
library(tidyverse)
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
library(patchwork)

source(here("code", "utils.R"))

# Load sample metadata for Ensembl instances
load(here("products", "result_files", "metadata_all.rda"))
```

2 Benchmark 1: `classify_gene_pairs()`

Here, we will benchmark the performance of `classify_gene_pairs()` with model organisms.

First, let's get the genome and annotation data.

```
# Create a data frame with names of model species and their Ensembl instances
model_species <- data.frame(
    species = c(
        "arabidopsis_thaliana", "caenorhabditis_elegans",
        "homo_sapiens", "saccharomyces_cerevisiae",
        "drosophila_melanogaster", "danio_rerio"
    ),
    instance = c(
        "plants", "metazoa", "ensembl", "fungi", "metazoa", "ensembl"
    )
)

# For each organism, download data, and identify and classify duplicates
model_duplicates <- lapply(seq_len(nrow(model_species)), function(x) {
```

```
        species <- model_species$species[x]
        instance <- model_species$instance[x]

        # Get annotation
        annot <- get_annotation(model_species[x, ], instance)

        # Get proteome and keep only primary transcripts
        seq <- get_proteomes(model_species[x, ], instance)
        seq <- filter_sequences(seq, annot)

        # Process data
        pdata <- syntenet::process_input(seq, annot, filter_annotation = TRUE)

        # Perform DIAMOND search
        outdir <- file.path(tempdir(), paste0(species, "_intra"))
        diamond <- syntenet::run_diamond(
            seq = pdata$seq,
            compare = "intraspecies",
            outdir = outdir,
            threads = 4,
            ... = "--sensitive"
        )

        fs::dir_delete(outdir)

        # Classify duplicates - standard mode
        start <- Sys.time()
        duplicate_pairs <- classify_gene_pairs(
                blast_list = diamond,
                annotation = pdata$annotation,
                scheme = "standard"
            )[[1]]
        end <- Sys.time()
        runtime <- end - start

        return(runtime)
    })
    names(model_duplicates) <- gsub("_", " ", str_to_title(model_species$species))

    # Summarize results in a table
    benchmark_classification <- data.frame(
        species = names(model_duplicates),
        time_seconds = as.numeric(unlist(model_duplicates))
    )

    # Save results
    save(
        benchmark_classification, compress = "xz",
        file = here("products", "result_files", "benchmark_classification.rda")
    )
```

3 Benchmark 2: `pairs2kaks()`

Next, we will benchmark the performance of `pairs2kaks()` for duplicate pairs in the *Saccharomyces cerevisiae* genome. We will do it using a single thread, and using parallelization (with 4 and 8 threads).

First of all, let's get the required data for `pairs2kaks()`.

```
# Load duplicate pairs for S. cerevisiae
load(here("products", "result_files", "fungi_duplicates.rda"))
scerevisiae_pairs <- fungi_duplicates["saccharomyces_cerevisiae"]

# Get CDS for S. cerevisiae
scerevisiae_cds <- get_cds_ensembl("saccharomyces_cerevisiae", "fungi")
```

Now, we can do the benchmark.

```
# Parallel back-end: SerialParam (1 thread)
start <- Sys.time()
kaks <- pairs2kaks(
    scerevisiae_pairs,
    scerevisiae_cds,
    bp_param = BiocParallel::SerialParam()
)
end <- Sys.time()
runtime_serial <- end - start

# Parallel back-end: SnowParam, 4 threads
start <- Sys.time()
kaks <- pairs2kaks(
    scerevisiae_pairs,
    scerevisiae_cds,
    bp_param = BiocParallel::SnowParam(workers = 4)
)
end <- Sys.time()
runtime_snow4 <- end - start

# Parallel back-end: SnowParam, 8 threads
start <- Sys.time()
kaks <- pairs2kaks(
    scerevisiae_pairs,
    scerevisiae_cds,
    bp_param = BiocParallel::SnowParam(workers = 8)
)
end <- Sys.time()
runtime_snow8 <- end - start

# Summarize results in a table
benchmark_kaks <- data.frame(
    `Back-end` = c("Serial", "Snow, 4 threads", "Snow, 8 threads"),
    Time_minutes = as.numeric(c(runtime_serial, runtime_snow4, runtime_snow8))
) |>
```

```
        dplyr::mutate(
            Pairs_per_minute = nrow(scerevisiae_pairs[[1]]) / Time_minutes,
            Pairs_per_second = nrow(scerevisiae_pairs[[1]]) / (Time_minutes * 60)
        )

    save(
        benchmark_kaks, compress = "xz",
        file = here("products", "result_files", "benchmark_kaks.rda")
    )
```

Session info

This document was created under the following conditions:

```
## - Session info ---------------------------------------------------------------
##  setting  value
##  version  R version 4.3.2 (2023-10-31)
##  os       Ubuntu 22.04.3 LTS
##  system   x86_64, linux-gnu
##  ui       X11
##  language (EN)
##  collate  en_US.UTF-8
##  ctype    en_US.UTF-8
##  tz       Europe/Brussels
##  date     2024-02-09
##  pandoc   3.1.1 @ /usr/lib/rstudio/resources/app/bin/quarto/bin/tools/ (via rmarkdown)
##
## - Packages -------------------------------------------------------------------
##  package             * version   date (UTC) lib source
##  abind                 1.4-5     2016-07-21 [1] CRAN (R 4.3.2)
##  ade4                  1.7-22    2023-02-06 [1] CRAN (R 4.3.2)
##  AnnotationDbi         1.64.1    2023-11-03 [1] Bioconductor
##  ape                   5.7-1     2023-03-13 [1] CRAN (R 4.3.2)
##  Biobase               2.62.0    2023-10-24 [1] Bioconductor
##  BiocFileCache         2.10.1    2023-10-26 [1] Bioconductor
##  BiocGenerics          0.48.1    2023-11-01 [1] Bioconductor
##  BiocIO                1.12.0    2023-10-24 [1] Bioconductor
##  BiocManager           1.30.22   2023-08-08 [1] CRAN (R 4.3.2)
##  BiocParallel          1.37.0    2024-01-19 [1] Github (Bioconductor/BiocParallel@79a1b2d)
##  BiocStyle           * 2.30.0    2023-10-24 [1] Bioconductor
##  biomaRt               2.58.0    2023-10-24 [1] Bioconductor
##  Biostrings            2.70.1    2023-10-25 [1] Bioconductor
##  bit                   4.0.5     2022-11-15 [1] CRAN (R 4.3.2)
##  bit64                 4.0.5     2020-08-30 [1] CRAN (R 4.3.2)
##  bitops                1.0-7     2021-04-24 [1] CRAN (R 4.3.2)
##  blob                  1.2.4     2023-03-17 [1] CRAN (R 4.3.2)
##  bookdown              0.37      2023-12-01 [1] CRAN (R 4.3.2)
##  cachem                1.0.8     2023-05-01 [1] CRAN (R 4.3.2)
##  cli                   3.6.2     2023-12-11 [1] CRAN (R 4.3.2)
```

```
##  coda                  0.19-4    2020-09-30 [1] CRAN (R 4.3.2)
##  codetools             0.2-19    2023-02-01 [4] CRAN (R 4.2.2)
##  colorspace            2.1-0     2023-01-23 [1] CRAN (R 4.3.2)
##  crayon                1.5.2     2022-09-29 [1] CRAN (R 4.3.2)
##  curl                  5.2.0     2023-12-08 [1] CRAN (R 4.3.2)
##  DBI                   1.1.3     2022-06-18 [1] CRAN (R 4.3.2)
##  dbplyr                2.4.0     2023-10-26 [1] CRAN (R 4.3.2)
##  DelayedArray          0.28.0    2023-10-24 [1] Bioconductor
##  digest                0.6.33    2023-07-07 [1] CRAN (R 4.3.2)
##  doParallel            1.0.17    2022-02-07 [1] CRAN (R 4.3.2)
##  doubletrouble       * 1.3.4     2024-02-05 [1] Bioconductor
##  dplyr               * 1.1.4     2023-11-17 [1] CRAN (R 4.3.2)
##  evaluate              0.23      2023-11-01 [1] CRAN (R 4.3.2)
##  fansi                 1.0.6     2023-12-08 [1] CRAN (R 4.3.2)
##  fastmap               1.1.1     2023-02-24 [1] CRAN (R 4.3.2)
##  filelock              1.0.3     2023-12-11 [1] CRAN (R 4.3.2)
##  forcats             * 1.0.0     2023-01-29 [1] CRAN (R 4.3.2)
##  foreach               1.5.2     2022-02-02 [1] CRAN (R 4.3.2)
##  generics              0.1.3     2022-07-05 [1] CRAN (R 4.3.2)
##  GenomeInfoDb          1.38.2    2023-12-13 [1] Bioconductor 3.18 (R 4.3.2)
##  GenomeInfoDbData      1.2.11    2023-12-21 [1] Bioconductor
##  GenomicAlignments     1.38.0    2023-10-24 [1] Bioconductor
##  GenomicFeatures       1.54.1    2023-10-29 [1] Bioconductor
##  GenomicRanges         1.54.1    2023-10-29 [1] Bioconductor
##  ggnetwork             0.5.12    2023-03-06 [1] CRAN (R 4.3.2)
##  ggplot2             * 3.4.4     2023-10-12 [1] CRAN (R 4.3.2)
##  glue                  1.6.2     2022-02-24 [1] CRAN (R 4.3.2)
##  gtable                0.3.4     2023-08-21 [1] CRAN (R 4.3.2)
##  here                * 1.0.1     2020-12-13 [1] CRAN (R 4.3.2)
##  hms                   1.1.3     2023-03-21 [1] CRAN (R 4.3.2)
##  htmltools             0.5.7     2023-11-03 [1] CRAN (R 4.3.2)
##  htmlwidgets           1.6.4     2023-12-06 [1] CRAN (R 4.3.2)
##  httr                  1.4.7     2023-08-15 [1] CRAN (R 4.3.2)
##  igraph                2.0.1.1   2024-01-30 [1] CRAN (R 4.3.2)
##  intergraph            2.0-3     2023-08-20 [1] CRAN (R 4.3.2)
##  IRanges               2.36.0    2023-10-24 [1] Bioconductor
##  iterators             1.0.14    2022-02-05 [1] CRAN (R 4.3.2)
##  KEGGREST              1.42.0    2023-10-24 [1] Bioconductor
##  knitr                 1.45      2023-10-30 [1] CRAN (R 4.3.2)
##  lattice               0.22-5    2023-10-24 [4] CRAN (R 4.3.1)
##  lifecycle             1.0.4     2023-11-07 [1] CRAN (R 4.3.2)
##  lubridate           * 1.9.3     2023-09-27 [1] CRAN (R 4.3.2)
##  magrittr              2.0.3     2022-03-30 [1] CRAN (R 4.3.2)
##  MASS                  7.3-60    2023-05-04 [4] CRAN (R 4.3.1)
##  Matrix                1.6-3     2023-11-14 [4] CRAN (R 4.3.2)
##  MatrixGenerics        1.14.0    2023-10-24 [1] Bioconductor
##  matrixStats           1.2.0     2023-12-11 [1] CRAN (R 4.3.2)
##  mclust                6.0.1     2023-11-15 [1] CRAN (R 4.3.2)
##  memoise               2.0.1     2021-11-26 [1] CRAN (R 4.3.2)
##  MSA2dist              1.6.0     2023-10-24 [1] Bioconductor
##  munsell               0.5.0     2018-06-12 [1] CRAN (R 4.3.2)
```

```
##  network               1.18.2    2023-12-05 [1] CRAN (R 4.3.2)
##  networkD3             0.4       2017-03-18 [1] CRAN (R 4.3.2)
##  nlme                  3.1-163   2023-08-09 [4] CRAN (R 4.3.1)
##  patchwork           * 1.2.0     2024-01-08 [1] CRAN (R 4.3.2)
##  pheatmap              1.0.12    2019-01-04 [1] CRAN (R 4.3.2)
##  pillar                1.9.0     2023-03-22 [1] CRAN (R 4.3.2)
##  pkgconfig             2.0.3     2019-09-22 [1] CRAN (R 4.3.2)
##  png                   0.1-8     2022-11-29 [1] CRAN (R 4.3.2)
##  prettyunits           1.2.0     2023-09-24 [1] CRAN (R 4.3.2)
##  progress              1.2.3     2023-12-06 [1] CRAN (R 4.3.2)
##  purrr               * 1.0.2     2023-08-10 [1] CRAN (R 4.3.2)
##  R6                    2.5.1     2021-08-19 [1] CRAN (R 4.3.2)
##  rappdirs              0.3.3     2021-01-31 [1] CRAN (R 4.3.2)
##  RColorBrewer          1.1-3     2022-04-03 [1] CRAN (R 4.3.2)
##  Rcpp                  1.0.11    2023-07-06 [1] CRAN (R 4.3.2)
##  RCurl                 1.98-1.13 2023-11-02 [1] CRAN (R 4.3.2)
##  readr               * 2.1.4     2023-02-10 [1] CRAN (R 4.3.2)
##  restfulr              0.0.15    2022-06-16 [1] CRAN (R 4.3.2)
##  rjson                 0.2.21    2022-01-09 [1] CRAN (R 4.3.2)
##  rlang                 1.1.2     2023-11-04 [1] CRAN (R 4.3.2)
##  rmarkdown             2.25      2023-09-18 [1] CRAN (R 4.3.2)
##  rprojroot             2.0.4     2023-11-05 [1] CRAN (R 4.3.2)
##  Rsamtools             2.18.0    2023-10-24 [1] Bioconductor
##  RSQLite               2.3.4     2023-12-08 [1] CRAN (R 4.3.2)
##  rstudioapi            0.15.0    2023-07-07 [1] CRAN (R 4.3.2)
##  rtracklayer           1.62.0    2023-10-24 [1] Bioconductor
##  S4Arrays              1.2.0     2023-10-24 [1] Bioconductor
##  S4Vectors             0.40.2    2023-11-23 [1] Bioconductor 3.18 (R 4.3.2)
##  scales                1.3.0     2023-11-28 [1] CRAN (R 4.3.2)
##  seqinr                4.2-36    2023-12-08 [1] CRAN (R 4.3.2)
##  sessioninfo           1.2.2     2021-12-06 [1] CRAN (R 4.3.2)
##  SparseArray           1.2.2     2023-11-07 [1] Bioconductor
##  statnet.common        4.9.0     2023-05-24 [1] CRAN (R 4.3.2)
##  stringi               1.8.3     2023-12-11 [1] CRAN (R 4.3.2)
##  stringr             * 1.5.1     2023-11-14 [1] CRAN (R 4.3.2)
##  SummarizedExperiment  1.32.0    2023-10-24 [1] Bioconductor
##  syntenet              1.4.0     2023-10-24 [1] Bioconductor
##  tibble              * 3.2.1     2023-03-20 [1] CRAN (R 4.3.2)
##  tidyr               * 1.3.0     2023-01-24 [1] CRAN (R 4.3.2)
##  tidyselect            1.2.0     2022-10-10 [1] CRAN (R 4.3.2)
##  tidyverse           * 2.0.0     2023-02-22 [1] CRAN (R 4.3.2)
##  timechange            0.2.0     2023-01-11 [1] CRAN (R 4.3.2)
##  tzdb                  0.4.0     2023-05-12 [1] CRAN (R 4.3.2)
##  utf8                  1.2.4     2023-10-22 [1] CRAN (R 4.3.2)
##  vctrs                 0.6.5     2023-12-01 [1] CRAN (R 4.3.2)
##  withr                 2.5.2     2023-10-30 [1] CRAN (R 4.3.2)
##  xfun                  0.41      2023-11-01 [1] CRAN (R 4.3.2)
##  XML                   3.99-0.16 2023-11-29 [1] CRAN (R 4.3.2)
##  xml2                  1.3.6     2023-12-04 [1] CRAN (R 4.3.2)
##  XVector               0.42.0    2023-10-24 [1] Bioconductor
##  yaml                  2.3.8     2023-12-11 [1] CRAN (R 4.3.2)
```

Runtime benchmark

```
##  zlibbioc               1.48.0    2023-10-24 [1] Bioconductor
##
##  [1] /home/faalm/R/x86_64-pc-linux-gnu-library/4.3
##  [2] /usr/local/lib/R/site-library
##  [3] /usr/lib/R/site-library
##  [4] /usr/lib/R/library
##
## ------------------------------------------------------------------------------
```