Supplemental File of

clusterProfiler 4.0: A universal enrichment tool for functional and comparative study

Tianzhi Wu[§], Erqiang Hu[§], Meijun Chen, Pingfan Guo, Zehan Dai, Tingze Feng, Shuangbin Xu, Lang Zhou, Wenli Tang, Li Zhan, Xiaocong Fu, Shanshan Liu, Xiaochen Bo[*] and Guangchuang Yu[*]

[*]correspondence: Guangchuang Yu <gcyu1@smu.edu.cn> and Xiaochen Bo <boxc@bmi.ac.cn>

1 Installation

To install `clusterProfiler` package, please enter the following command in R:

```
if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
BiocManager::install("clusterProfiler")
```

To reproduce examples in this document, you need to install several extra packages:

```
install.packages(c("forcats", "ggplot2", "ggnewscale", "ggupset"))
BiocManager::install(c("org.Hs.eg.db", "enrichplot",
            "ChIPseeker", "TxDb.Hsapiens.UCSC.hg19.knownGene"))
```

2 R packages that depends on clusterProfiler

The `clusterProfiler` library is one of the fundamental packages and it had been incorporated in more than twenty R packages (in CRAN or Bioconductor) to perform functional enrichment analysis for different topics, especially for cancer research.

```
db <- utils::available.packages(repo=BiocManager::repositories())
pkgs <- tools::package_dependencies('clusterProfiler', db=db,
                            which = c("Depends", "Imports"),
                            reverse=TRUE)[[1]]
pkgs
```

```
##  [1] "bioCancer"       "CEMiTool"        "CeTF"            "conclus"
##  [5] "DAPAR"           "debrowser"       "eegc"            "enrichTF"
##  [9] "esATAC"          "famat"           "fcoex"           "GDCRNATools"
## [13] "IRISFGM"         "MAGeCKFlute"     "methylGSA"       "miRspongeR"
## [17] "MoonlightR"      "multiSight"      "netboxr"         "PFP"
## [21] "RNASeqR"         "signatureSearch" "TCGAbiolinksGUI" "TimiRGeN"
## [25] "ExpHunterSuite"  "maEndToEnd"      "recountWorkflow" "TCGAWorkflow"
## [29] "AutoPipe"        "immcp"           "RVA"
```

3 Examples of using clusterProfiler

This session provides source codes to reproduce the figures presented in the manuscript.

3.1 GO enrichment analysis

```
library(clusterProfiler)
library(enrichplot)

## geneList for GSEA examples
data(geneList, package="DOSE")

## fold change > 2 as DE genes, for ORA examples
de <- names(geneList)[abs(geneList) > 2]


ego <- enrichGO(de, OrgDb = "org.Hs.eg.db", ont="BP", readable=TRUE)
```

```
## use simplify to remove redundant terms
ego2 <- simplify(ego, cutoff=0.7, by="p.adjust", select_fun=min)


## visualization
ego <- pairwise_termsim(ego)
ego2 <- pairwise_termsim(ego2)

p1 <- emapplot(ego, cex_label_category=.8, cex_line=.5)
p2 <- emapplot(ego2, cex_label_category=.8, cex_line=.5)
cowplot::plot_grid(p1, p2, labels=c("A", "B"))
```

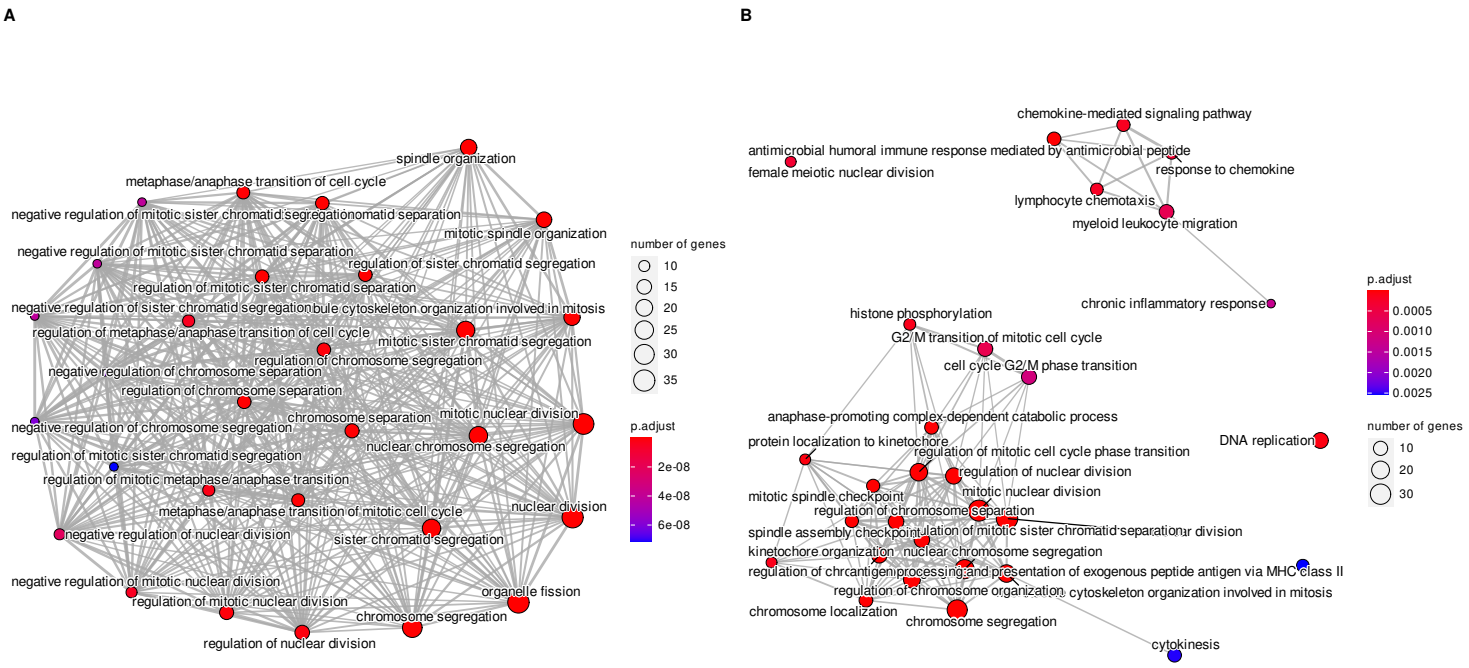

Fig. 1: Gene ontology enrichment analysis.

3.2 KEGG enrichment analysis

```
kk <- gseKEGG(geneList, organism = "hsa")

## sorted by absolute values of NES
kk2 <- arrange(kk, abs(NES))

## visualization
kp1 <- gseaplot2(kk2, 1:5, pvalue_table=F, base_size=14)
kp2 <- upsetplot(kk2, n=5)
cowplot::plot_grid(kp1, kp2, rel_widths=c(1, .5), labels=c("A", "B"))
```

3.3 Universal interface for biomedical gene sets

```
## downloaded from https://wikipathways-data.wmcloud.org/current/gmt/
gmt <- 'wikipathways-20210310-gmt-Homo_sapiens.gmt'
wp <- read.gmt.wp(gmt)
ewp <- GSEA(geneList, TERM2GENE=wp[,c("wpid", "gene")], TERM2NAME=wp[,c("wpid", "name")])
```

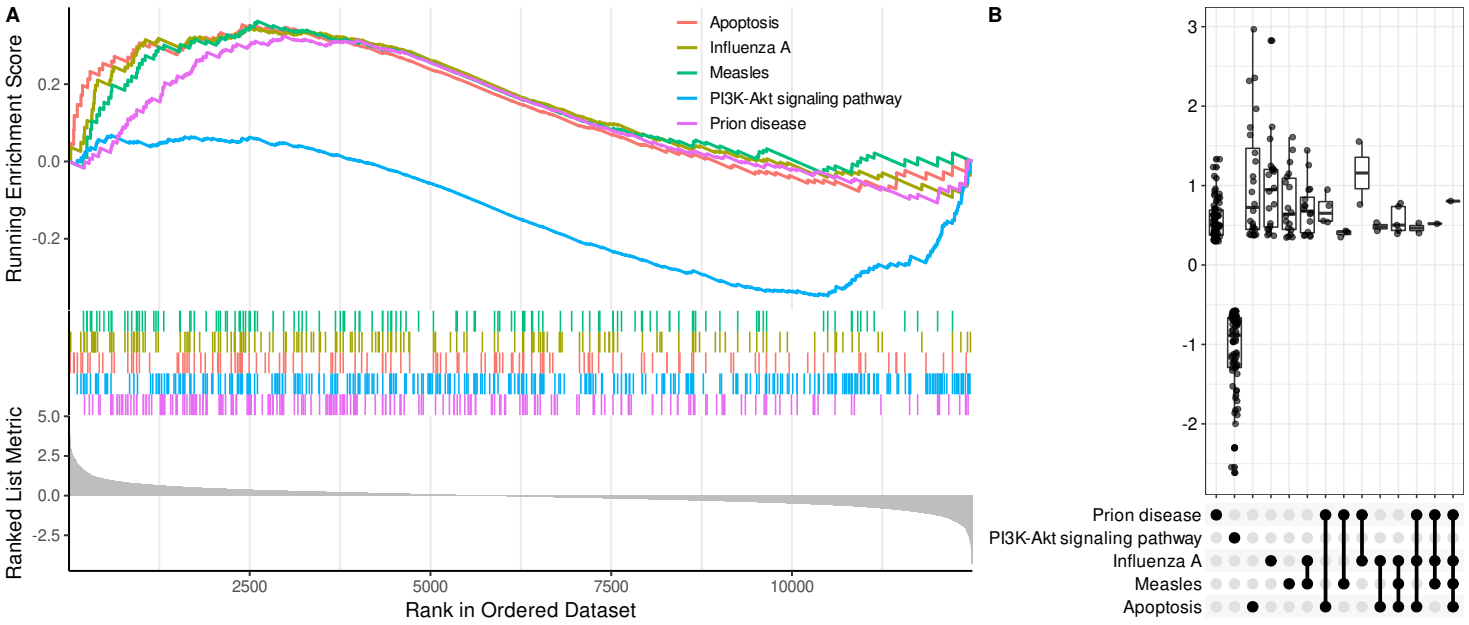

Fig. 2: KEGG pathway enrichment analysis.

3.4 Functional interpretation of genomic regions of interest

```
library(ChIPseeker)
## the file can be downloaded using `downloadGSMbedFiles("GSM1295076")`
file <- "GSM1295076_CBX6_BF_ChipSeq_mergedReps_peaks.bed.gz"
gr <- readPeakFile(file)

library(TxDb.Hsapiens.UCSC.hg19.knownGene)
TxDb <- TxDb.Hsapiens.UCSC.hg19.knownGene
genes <- seq2gene(gr, tssRegion=c(-1000, 1000), flankDistance = 3000, TxDb)

library(clusterProfiler)
## downloaded from https://maayanlab.cloud/Enrichr/geneSetLibrary?mode=text&libraryName=ENCODE_and_ChEA_Consens
encode <- read.gmt("ENCODE_and_ChEA_Consensus_TFs_from_ChIP-X.txt")
g <- bitr(genes, 'ENTREZID', 'SYMBOL', 'org.Hs.eg.db')
```

```
## Warning in bitr(genes, "ENTREZID", "SYMBOL", "org.Hs.eg.db"): 5.32% of input
## gene IDs are fail to map...
```

```
x <- enricher(g$SYMBOL, TERM2GENE=encode)
enrichplot::cnetplot(x, cex_label_gene=0.6)
```

3.5 Comparison for different conditions

```
data(DE_GSE8057)

xx <- compareCluster(Gene~time+treatment, data=DE_GSE8057, fun = enricher,
              TERM2GENE=wp[,c("wpid", "gene")], TERM2NAME=wp[,c("wpid", "name")])

pp <- dotplot(xx, x="time") + facet_grid(~treatment) +
    aes(x=fct_relevel(time, c('0h', '2h', '6h', '24h'))) + xlab(NULL)
```

```
print(pp)
```

3.6 Visualization using ggplot2

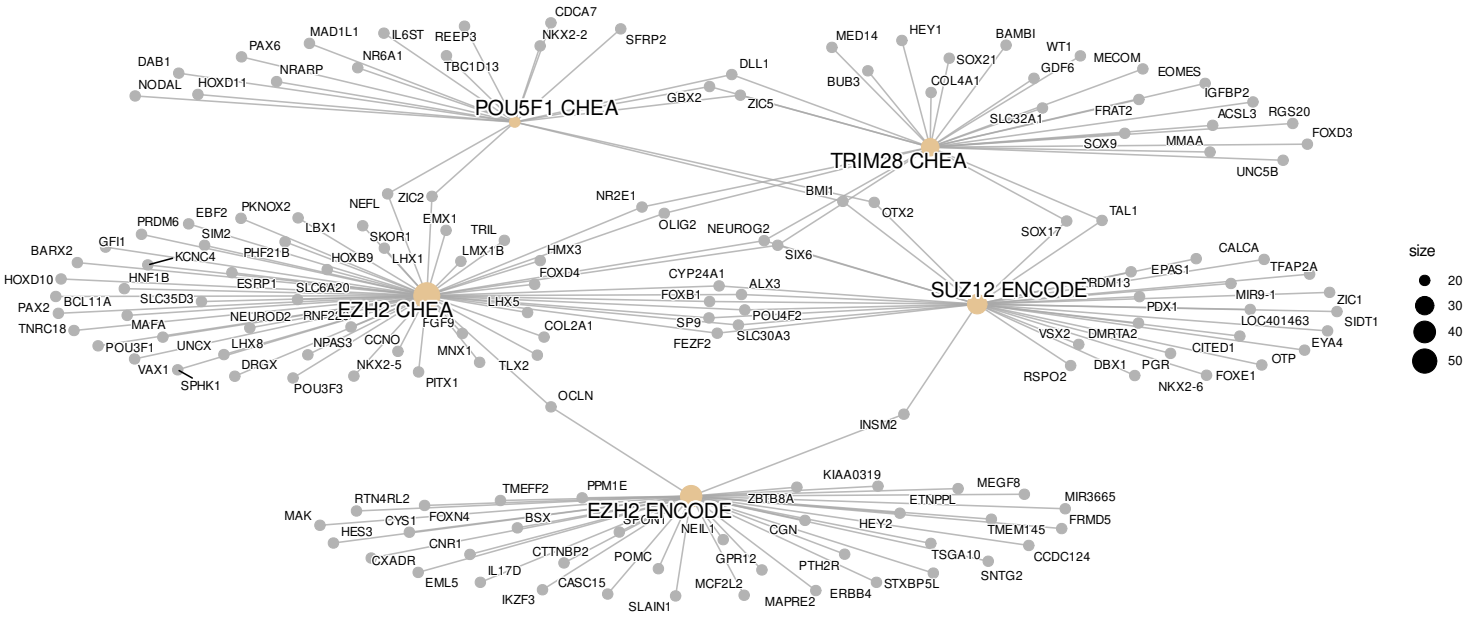

Fig. 3: Functional enrichment analysis of genomic regions of interest.

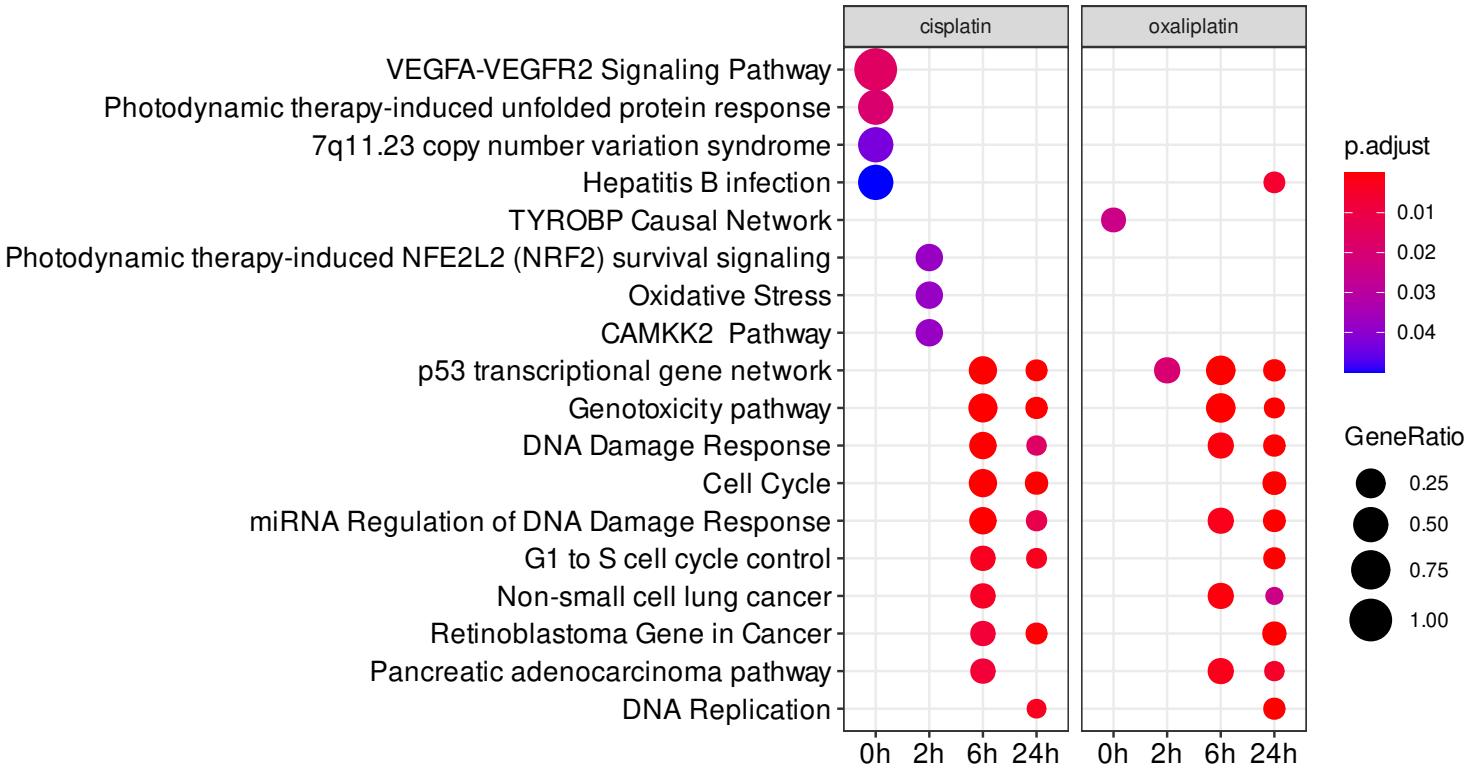

Fig. 4: Comparing functional profiles among different levels of conditions.

```
library(forcats)
library(ggplot2)

## downloaded from https://wikipathways-data.wmcloud.org/current/gmt/
gmt <- 'wikipathways-20210310-gmt-Homo_sapiens.gmt'
wp <- read.gmt.wp(gmt)
ewp <- GSEA(geneList, TERM2GENE=wp[,c("wpid", "gene")], TERM2NAME=wp[,c("wpid", "name")])

ewp2 <- arrange(ewp, abs(NES)) %>%
        group_by(sign(NES)) %>%
```

```
        slice(1:5)
ego3 <- mutate(ego, richFactor = Count / as.numeric(sub("/\\d+", "", BgRatio)))

g1 <- ggplot(ego3, showCategory = 10,
  aes(richFactor, fct_reorder(Description, richFactor))) +
  geom_segment(aes(xend=0, yend = Description)) +
  geom_point(aes(color=p.adjust, size = Count)) +
  scale_color_viridis_c(guide=guide_colorbar(reverse=TRUE)) +
  scale_size_continuous(range=c(2, 10)) +
  theme_dose(12) +
  xlab("Rich Factor") +
  ylab(NULL) +
  ggtitle("Biological Processes")

g2 <- ggplot(ewp2, showCategory=10,
        aes(NES, fct_reorder(Description, NES), fill=qvalues)) +
    geom_col() +
    scale_fill_continuous(low='red', high='blue',
                          guide=guide_colorbar(reverse=TRUE)) +
    theme_dose(12) +
    xlab("Normalized Enrichment Score") +
    ylab(NULL) +
    ggtitle("WikiPathways")

cowplot::plot_grid(g1, g2, labels=c("A", "B"))
```

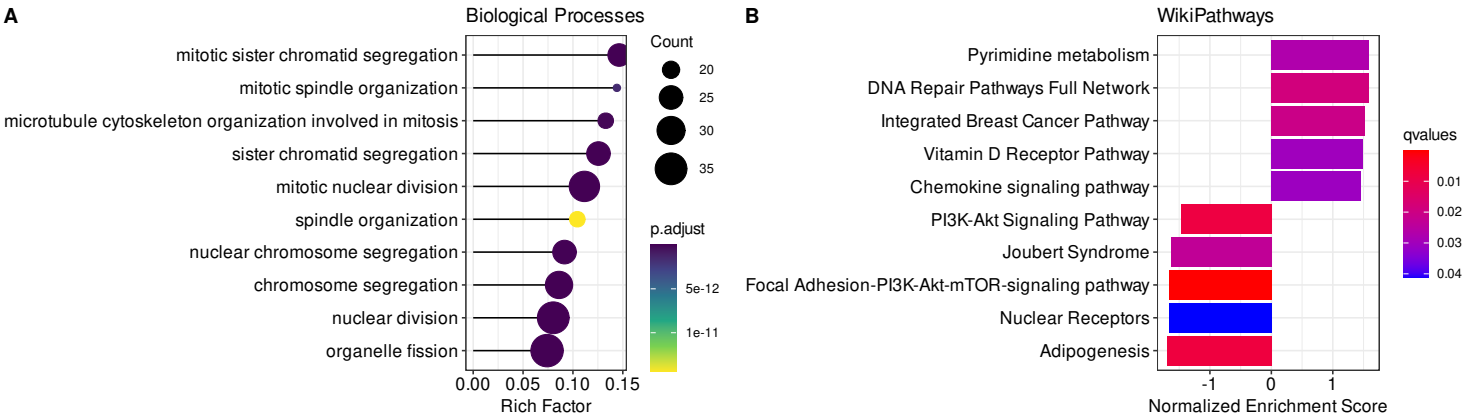

Fig. 5: Visualization enrichment results using ggplot2.

NOTE: source codes and datasets to produce this file can be obtained online[1].

4 Session information

Here is the output of `sessionInfo()` of the system on which the Supplemental file was compiled:

```
## R version 4.1.0 (2021-05-18)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Arch Linux
##
## Matrix products: default
## BLAS:   /usr/lib/libblas.so.3.9.1
## LAPACK: /usr/lib/liblapack.so.3.9.1
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
```

[1] https://github.com/YuLab-SMU/supplemental-clusterProfiler-v4

```
##  [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 utils     datasets
## [8] methods   base
##
## other attached packages:
##  [1] TxDb.Hsapiens.UCSC.hg19.knownGene_3.2.2
##  [2] GenomicFeatures_1.44.0
##  [3] GenomicRanges_1.44.0
##  [4] GenomeInfoDb_1.28.0
##  [5] ChIPseeker_1.28.0
##  [6] forcats_0.5.1
##  [7] ggplot2_3.3.3
##  [8] enrichplot_1.12.0
##  [9] clusterProfiler_4.0.0
## [10] DOSE_3.18.0
## [11] org.Hs.eg.db_3.13.0
## [12] AnnotationDbi_1.54.0
## [13] IRanges_2.26.0
## [14] S4Vectors_0.30.0
## [15] Biobase_2.52.0
## [16] BiocGenerics_0.38.0
## [17] kableExtra_1.3.4
## [18] magrittr_2.0.1
## [19] conflicted_1.0.4
## [20] rvcheck_0.1.8
## [21] wget_0.0.1
## [22] rmarkdown_2.8
##
## loaded via a namespace (and not attached):
##   [1] shadowtext_0.0.8           fastmatch_1.1-0
##   [3] BiocFileCache_2.0.0        systemfonts_1.0.2
##   [5] plyr_1.8.6                 igraph_1.2.6
##   [7] lazyeval_0.2.2             splines_4.1.0
##   [9] BiocParallel_1.26.0        digest_0.6.27
##  [11] htmltools_0.5.1.1          GOSemSim_2.18.0
##  [13] viridis_0.6.1              GO.db_3.13.0
##  [15] fansi_0.4.2                memoise_2.0.0
##  [17] Biostrings_2.60.0          graphlayouts_0.7.1
##  [19] matrixStats_0.58.0         svglite_2.0.0
##  [21] prettyunits_1.1.1          colorspace_2.0-1
##  [23] blob_1.2.1                 rvest_1.0.0
##  [25] rappdirs_0.3.3             ggrepel_0.9.1
##  [27] xfun_0.23                  dplyr_1.0.6
##  [29] crayon_1.4.1               RCurl_1.98-1.3
##  [31] jsonlite_1.7.2             scatterpie_0.1.6
##  [33] ape_5.5                    glue_1.4.2
##  [35] polyclip_1.10-0            gtable_0.3.0
##  [37] zlibbioc_1.38.0            XVector_0.32.0
##  [39] webshot_0.5.2              DelayedArray_0.18.0
##  [41] scales_1.1.1               DBI_1.1.1
##  [43] Rcpp_1.0.6                 plotrix_3.8-1
##  [45] viridisLite_0.4.0          progress_1.2.2
##  [47] tidytree_0.3.3             bit_4.0.4
##  [49] httr_1.4.2                 fgsea_1.18.0
##  [51] gplots_3.1.1               RColorBrewer_1.1-2
```

```
##  [53] ellipsis_0.3.2              pkgconfig_2.0.3
##  [55] XML_3.99-0.6                farver_2.1.0
##  [57] dbplyr_2.1.1                utf8_1.2.1
##  [59] labeling_0.4.2              tidyselect_1.1.1
##  [61] rlang_0.4.11                reshape2_1.4.4
##  [63] munsell_0.5.0               tools_4.1.0
##  [65] cachem_1.0.5                downloader_0.4
##  [67] generics_0.1.0              RSQLite_2.2.7
##  [69] evaluate_0.14               stringr_1.4.0
##  [71] fastmap_1.1.0               yaml_2.2.1
##  [73] ggtree_3.0.0                knitr_1.33
##  [75] bit64_4.0.5                 tidygraph_1.2.0
##  [77] caTools_1.18.2              purrr_0.3.4
##  [79] KEGGREST_1.32.0             ggraph_2.0.5
##  [81] nlme_3.1-152                aplot_0.0.6
##  [83] DO.db_2.9                   xml2_1.3.2
##  [85] biomaRt_2.48.0              compiler_4.1.0
##  [87] rstudioapi_0.13             filelock_1.0.2
##  [89] curl_4.3.1                  png_0.1-7
##  [91] treeio_1.16.0               tibble_3.1.2
##  [93] tweenr_1.0.2                stringi_1.6.2
##  [95] lattice_0.20-44             Matrix_1.3-3
##  [97] vctrs_0.3.8                 pillar_1.6.1
##  [99] lifecycle_1.0.0             BiocManager_1.30.15
## [101] data.table_1.14.0           cowplot_1.1.1
## [103] bitops_1.0-7                patchwork_1.1.1
## [105] rtracklayer_1.52.0          qvalue_2.24.0
## [107] R6_2.5.0                    BiocIO_1.2.0
## [109] bookdown_0.22               KernSmooth_2.23-20
## [111] gridExtra_2.3               codetools_0.2-18
## [113] gtools_3.8.2                boot_1.3-28
## [115] MASS_7.3-54                 assertthat_0.2.1
## [117] SummarizedExperiment_1.22.0 rjson_0.2.20
## [119] withr_2.4.2                 GenomicAlignments_1.28.0
## [121] Rsamtools_2.8.0             GenomeInfoDbData_1.2.6
## [123] hms_1.1.0                   ggupset_0.3.0
## [125] grid_4.1.0                  tidyr_1.1.3
## [127] MatrixGenerics_1.4.0        ggnewscale_0.4.5
## [129] ggforce_0.3.3               tinytex_0.31
## [131] restfulr_0.0.13
```

5 References