IOBR (Immuno-Oncology Biological Research)

Dongqiang Zeng

2023-09-08

Contents

Introduction **5**
0.1 Introduction . . . 6
0.2 License . . . 6
0.3 Publishment . . . 6
0.4 Major Updates . . . 7
0.5 Reporting bugs . . . 7

1 How to install IOBR **9**
1.1 Install Dependency Packages . . . 9
1.2 Install IOBR package . . . 9
1.3 How to update IOBR . . . 10
1.4 The main pipeline of IOBR . . . 10
1.5 Main Functions of IOBR . . . 10
1.6 Current working environment . . . 13

2 RNA Data preprocessing **17**
2.1 Loading packages . . . 17
2.2 Download array data using `GEOquery` . . . 17
2.3 Gene Annotation . . . 18
2.4 Identifying outlier samples . . . 20
2.5 PCA analysis of molecular subtypes . . . 21
2.6 Batch effect correction . . . 23
2.7 References . . . 25

3 Tumor ecosystem analysis **27**
3.1 Loading packages . . . 27
3.2 Downloading data for example . . . 27
3.3 Gene Annotation: HGU133PLUS-2 (Affaymetrix) . . . 28
3.4 Determine TME subtype of gastric cancer using TMEclassifier R package . . . 28
3.5 DEG analysis: method1 . . . 29
3.6 GSEA analysis based on differential express gene analysis results . . . 30
3.7 DEG analysis: method2 . . . 36
3.8 Identifying signatures associated with TME clusters . . . 39

4 Signature and relevant phenotypes **47**
4.1 Loading packages . . . 47
4.2 Downloading data for example . . . 47
4.3 Signature score estimation . . . 48
4.4 Identifying features associated with survival . . . 53
4.5 Visulization using heatmap . . . 54
4.6 Focus on target signatures . . . 55
4.7 Survival analysis . . . 57
4.8 Batch correlation analysis . . . 60
4.9 Visulization of correlations . . . 63

5 TME deconvolution **65**
5.1 Loading packages . . . 65
5.2 Downloading data for example . . . 65
5.3 Available Methods to Decode TME Contexture . . . 66
5.4 Method 1: CIBERSORT . . . 67
5.5 Method 2: EPIC . . . 68
5.6 Method 3: MCPcounter . . . 69
5.7 Method 4: xCELL . . . 70
5.8 Method 5: ESTIMATE . . . 70
5.9 Method 6: TIMER . . . 71
5.10 Method 7: quanTIseq . . . 71
5.11 Method 8: IPS . . . 73
5.12 Combination of above deconvolution results . . . 74

6 References **77**
6.1 TME deconvolution . . . 77
6.2 TME Signatures . . . 78
6.3 Data sets . . . 78
6.4 Others . . . 79

Introduction

Preface

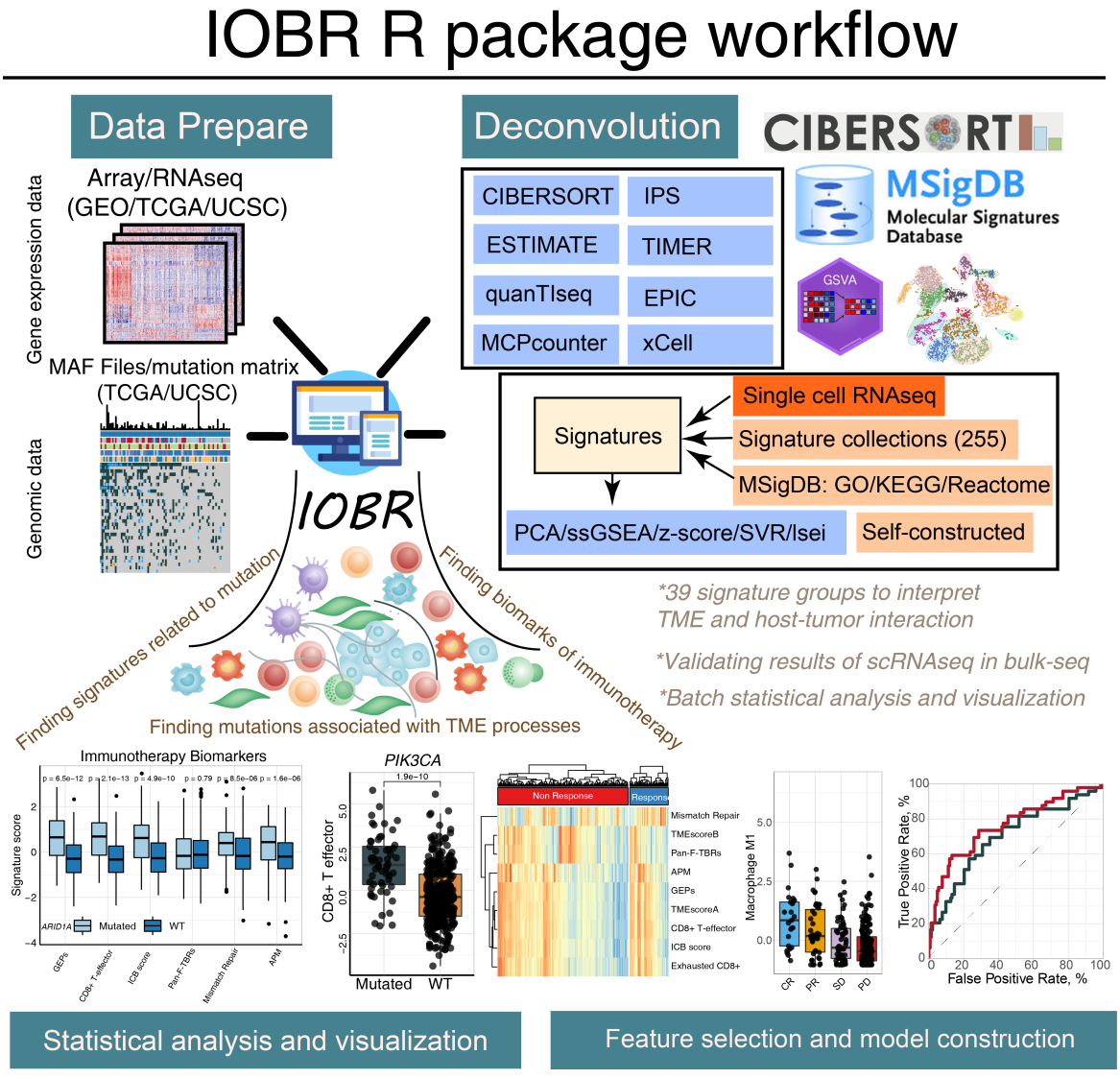


Figure 1: The workflow of IOBR

0.1 Introduction

IOBR is design for Immuno-Oncology Biological Research. Recent advance in next-generation sequencing has triggered the rapidly accumulating publicly available multi-omics data. The application of integrated omics to exploring robust signatures for clinical translation is increasingly highlighted in immuno-oncology but raises computational and biological challenges. This vignette aims to demonstrate how to utilize the package named IOBR to perform multi-omics immuno-oncology biological research to decode tumor microenvironment and signatures for clinical translation.

This R package integrates 8 published methodologies for decoding tumor microenvironment (TME) contexture: `CIBERSORT`, `TIMER`, `xCell`, `MCPcounter`, `ESITMATE`, `EPIC`, `IPS`, `quanTIseq`. Moreover, 255 published signature gene sets were collected by IOBR, involving tumor microenvironment, tumor metabolism, m6A, exosomes, microsatellite instability, and tertiary lymphoid structure. Run the function `signature_collection_citation` to obtain the source papers, and the function `signature_collection` returns the detail signature genes of all given signatures. Subsequently, IOBR adopts three computational methods to calculate the signature score, comprising `PCA`, `z-score`, and `ssGSEA`. To note, IOBR collected and employed multiple approaches for variable transition, visualization, batch survival analysis, feature selection, and statistical analysis. Batch analysis and visualization of corresponding results are supported. The details of how IOBR works are described below.

0.2 License

0.3 Publishment

Zeng D, Ye Z, Shen R, Yu G, Wu J, Xiong Y,..., Liao W (2021) **IOBR**: Multi-Omics Immuno-Oncology Biological Research to Decode Tumor Microenvironment and Signatures. *Frontiers in Immunology.* 12:687975. doi: 10.3389/fimmu.2021.687975

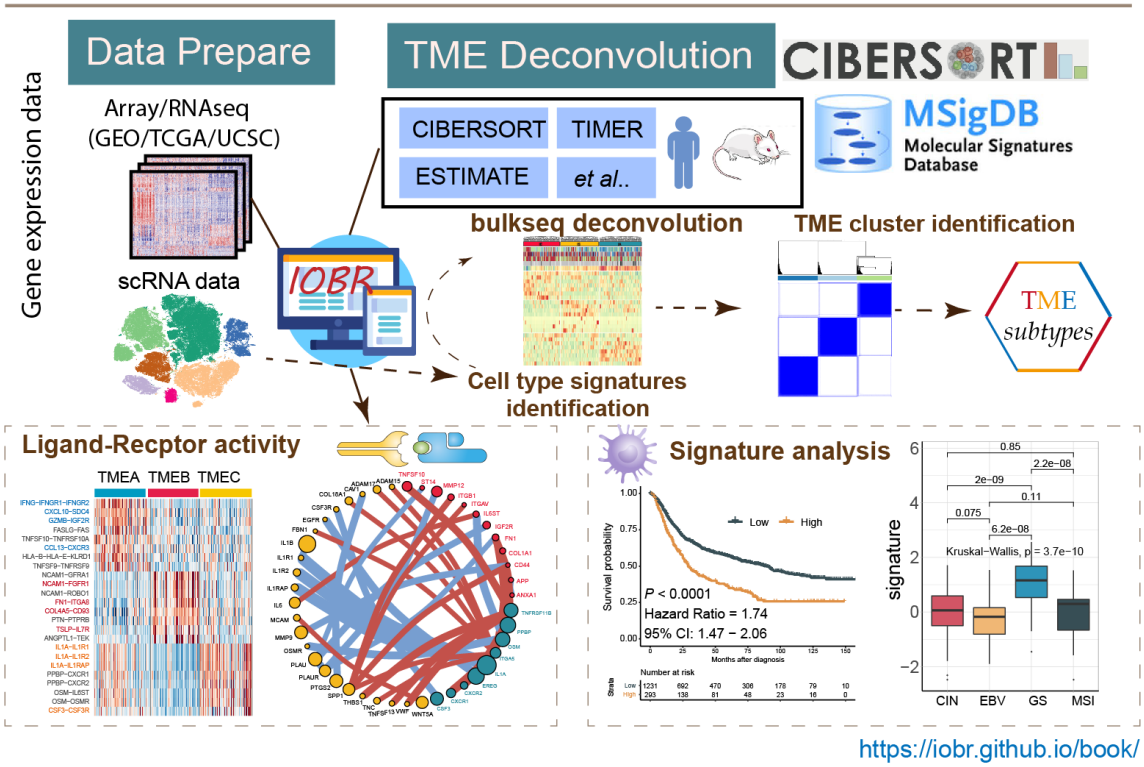


Figure 2: The workflow of IOBR

0.4 Major Updates

0.5 Reporting bugs

Please report bugs to the Github issues page

E-mail any questions to dongqiangzeng0808@gmail.com

Chapter 1

How to install IOBR

1.1 Install Dependency Packages

It is essential that you have R 3.6.3 or above already installed on your computer or server. IOBR is a pipeline that utilizes many other R packages that are currently available from CRAN, Bioconductor and GitHub.

```
if (!requireNamespace("BiocManager", quietly = TRUE)) install.packages("BiocManager")
depens<-c('tibble', 'survival', 'survminer', 'limma', "DESeq2","devtools", 'limSolve', '
          "devtools", "tidyHeatmap", "caret", "glmnet", "ppcor",  "timeROC", "pracma", "
          "FactoMineR", "WGCNA", "patchwork", 'ggplot2', "biomaRt", 'ggpubr')
for(i in 1:length(depens)){
  depen<-depens[i]
  if (!requireNamespace(depen, quietly = TRUE))  BiocManager::install(depen,update = FA
}
```

1.2 Install IOBR package

When the dependent environments are built, users are able to install IOBR from github by typing the following code into your R session:

```
if (!requireNamespace("IOBR", quietly = TRUE))  devtools::install_github("IOBR/IOBR")

library(IOBR)
```

1.3 How to update IOBR

```
detach("package:IOBR")
path<-.libPaths()
remove.packages(c('IOBR'), lib=file.path(path))
devtools::install_github("IOBR/IOBR")
```

1.4 The main pipeline of IOBR

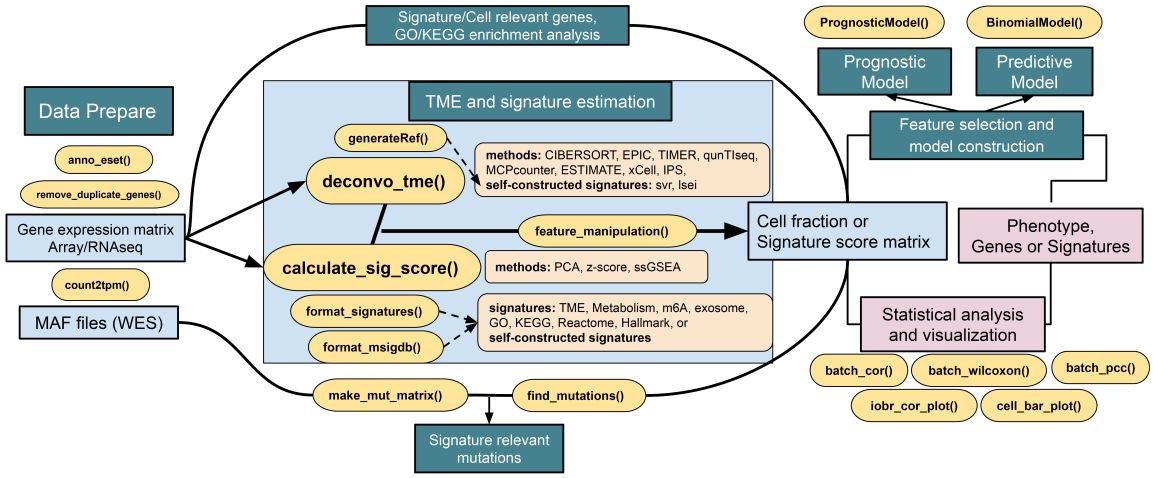


Figure 1.1: The main pipeline of IOBR

1.5 Main Functions of IOBR

- **Data Preparation: data annotation and transformation**
  - `count2tpm()`: transform count data of RNA sequencing into TPM data.
  - `anno_eset()`: annotate the normalized genes expression matrix, including RNAseq and array (Affymetrix or Illumina).
  - `remove_duplicate_genes()`: remove the genes annotated with the duplicated symbol after normalization and retain only the symbol with highest expression level.

  - `mouse2human_eset()`: Converting muouse gene symbol to human gene symbol of expression set.
  - `find_outlier_samples()`: Waiting for updates…
  - `remove_batcheffect()`: Waiting for updates…

- **TME Deconvolution Module: integrate multiple algorithms to decode immune contexture**

  - `deconvo_tme()`: decode the TME infiltration with different deconvolution methodologies, based on bulk RNAseq, microarray or single cell RNAseq data.
  - `generateRef()`: generate a novel gene reference matrix for a specific feature such as infiltrating cell, through the SVR and lsei algorithm.

- **Signature Module: calculatint signature scores, estimate phenotype related signatures and corresponding genes, and evaluate signatures generated from single-cell RNA sequencing data**

  - `calculate_sig_score()`: estimate the interested signatures enrolled in IOBR R package, which involves TME-associated, tumor-metabolism, and tumor-intrinsic signatures.
  - `feature_manipulation()`: manipulate features including the cell fraction and signatures generated from multi-omics data for latter analysis and model construction. Remove missing values, outliers and variables without significant variance.
  - `format_signatures()`: generate the object for `calculate_sig_score()` function, by inputting a data frame with signatures as column names of corresponding gene sets, and return a list contain the signature information for calculating multiple signature scores.
  - `format_msigdb()`: transform the signature gene sets data with gmt format, which is not included in the signature collection and might be downloaded in the MSgiDB website, into the object of `calculate_sig_score()`function.
  - `sig_gsea()`: Waiting for updates…

- **Batch Analysis and Visualization: batch survival analysis and batch correlation analysis and other batch statistical analyses**

  - `batch_surv()`: batch survival analysis of multiple continuous variables including varied signature scores.
  - `subgroup_survival()`: batch survival analysis of multiple categorized variables with different number of subgroups.
  - `batch_cor()`: batch analysis of correlation between two continuous variables using Pearson correlation coefficient or Spearman's rank correlation coefficient .

  - `batch_wilcoxon()`: conduct batch wilcoxon analyses of binary variables.
  - `batch_pcc()`: batch analyses of Partial Correlation coefficient(PCC) between continuous variables and minimize the interference derived from confounding factors.
  - `iobr_cor_plot()`: visualization of batch correlation analysis of signatures from 'sig_group'. Visualize the correlation between signature or phenotype with expression of gene sets in target signature is also supported.
  - `cell_bar_plot()`: batch visualization of TME cell fraction, supporting input of deconvolution results from 'CIBERSORT', 'EPIC' and 'quanTIseq' methodologies to further compare the TME cell distributions within one sample or among different samples.
  - `iobr_pca()`: The iobr_pca function performs Principal Component Analysis (PCA), which reduces the dimensionality of data while maintaining most of the original variance, and visualizes the PCA results on a scatter plot.
  - `iobr_cor_plot()`: Integrative correlation between phenotype and features.
  - `iobr_deg()`: Waiting for updates…
  - `get_cor()`: Waiting for updates…
  - `roc_time()`: Waiting for updates…
  - `sig_box()`: Waiting for updates…
  - `sig_heatmap()`: Waiting for updates…
  - `sig_forest()`: Waiting for updates…
  - `sig_roc()`: Waiting for updates…
  - `sig_surv_plot()`: Waiting for updates…
  - `find_markers_in_bulk()`: Waiting for updates…

- **Signature Associated Mutation Module: identify and analyze mutations relevant to targeted signatures**

  - `make_mut_matrix()`: transform the mutation data with MAF format(contain the columns of gene ID and the corresponding gene alterations which including SNP, indel and frameshift) into a mutation matrix in a suitable manner for further investigating signature relevant mutations.
  - `find_mutations()`: identify mutations associated with a distinct phenotype or signature.

- **Model Construction Module: feature selection and fast model construct to predict clinical phenotype**

  - `BinomialModel()`: select features and construct a model to predict a binary phenotype.

- `PrognosticMode()`: select features and construct a model to predict clinical survival outcome.

1.6 Current working environment

```
sessionInfo()
```

```
## R version 4.2.0 (2022-04-22 ucrt)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 19045)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=Chinese (Simplified)_China.utf8
## [2] LC_CTYPE=Chinese (Simplified)_China.utf8
## [3] LC_MONETARY=Chinese (Simplified)_China.utf8
## [4] LC_NUMERIC=C
## [5] LC_TIME=Chinese (Simplified)_China.utf8
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] IOBR_0.99.9           survminer_0.4.9       patchwork_1.1.1
##  [4] clusterProfiler_4.4.3 survival_3.3-1        ggpubr_0.4.0
##  [7] ggplot2_3.4.2         dplyr_1.1.2           tibble_3.2.1
## [10] tidyHeatmap_1.8.1     ComplexHeatmap_2.15.4
##
## loaded via a namespace (and not attached):
##   [1] utf8_1.2.2                 tidyselect_1.2.0
##   [3] RSQLite_2.2.14             AnnotationDbi_1.58.0
##   [5] BiocParallel_1.30.3        lpSolve_5.6.15
##   [7] scatterpie_0.1.7           ScaledMatrix_1.4.0
##   [9] munsell_0.5.0              codetools_0.2-18
##  [11] preprocessCore_1.58.0      withr_2.5.0
```

```
##  [13] colorspace_2.0-3            GOSemSim_2.22.0
##  [15] Biobase_2.56.0              limSolve_1.5.6
##  [17] knitr_1.39                  rstudioapi_0.13
##  [19] SingleCellExperiment_1.18.0 stats4_4.2.0
##  [21] ggsignif_0.6.3              DOSE_3.22.0
##  [23] MatrixGenerics_1.8.0        GenomeInfoDbData_1.2.8
##  [25] KMsurv_0.1-5                polyclip_1.10-0
##  [27] bit64_4.0.5                 farver_2.1.0
##  [29] rhdf5_2.40.0                downloader_0.4
##  [31] treeio_1.21.2               vctrs_0.6.2
##  [33] generics_0.1.3              xfun_0.40
##  [35] R6_2.5.1                    doParallel_1.0.17
##  [37] GenomeInfoDb_1.34.4         clue_0.3-61
##  [39] graphlayouts_0.8.0          rsvd_1.0.5
##  [41] locfit_1.5-9.5              rhdf5filters_1.8.0
##  [43] gridGraphics_0.5-1          bitops_1.0-7
##  [45] cachem_1.0.6                fgsea_1.22.0
##  [47] DelayedArray_0.22.0         assertthat_0.2.1
##  [49] scales_1.2.0                ggraph_2.0.5
##  [51] enrichplot_1.16.1           googlesheets4_1.0.0
##  [53] gtable_0.3.1                beachmat_2.12.0
##  [55] tidygraph_1.2.1             rlang_1.1.0
##  [57] genefilter_1.78.0           GlobalOptions_0.1.2
##  [59] splines_4.2.0               lazyeval_0.2.2
##  [61] rstatix_0.7.0               gargle_1.2.0
##  [63] broom_0.8.0                 yaml_2.3.5
##  [65] reshape2_1.4.4              abind_1.4-5
##  [67] modelr_0.1.8                backports_1.4.1
##  [69] qvalue_2.28.0               tools_4.2.0
##  [71] bookdown_0.35               ggplotify_0.1.0
##  [73] ellipsis_0.3.2              RColorBrewer_1.1-3
##  [75] proxy_0.4-27                BiocGenerics_0.42.0
##  [77] Rcpp_1.0.9                  plyr_1.8.7
##  [79] sparseMatrixStats_1.8.0     zlibbioc_1.42.0
##  [81] purrr_0.3.4                 RCurl_1.98-1.7
##  [83] GetoptLong_1.0.5            viridis_0.6.2
##  [85] cowplot_1.1.1               S4Vectors_0.34.0
```

```
##  [87] zoo_1.8-10                 SummarizedExperiment_1.26.1
##  [89] haven_2.5.0                ggrepel_0.9.1
##  [91] cluster_2.1.3              fs_1.5.2
##  [93] magrittr_2.0.3             data.table_1.14.2
##  [95] DO.db_2.9                  circlize_0.4.15
##  [97] reprex_2.0.1               googledrive_2.0.0
##  [99] matrixStats_0.62.0         GSVA_1.44.2
## [101] hms_1.1.1                  evaluate_0.15
## [103] xtable_1.8-4               XML_3.99-0.10
## [105] readxl_1.4.0               IRanges_2.30.0
## [107] gridExtra_2.3              shape_1.4.6
## [109] compiler_4.2.0             shadowtext_0.1.2
## [111] crayon_1.5.2               htmltools_0.5.2
## [113] ggfun_0.0.6                tzdb_0.3.0
## [115] tidyr_1.2.0                geneplotter_1.74.0
## [117] aplot_0.1.6                lubridate_1.8.0
## [119] DBI_1.1.2                  tweenr_1.0.2
## [121] corrplot_0.92              dbplyr_2.2.0
## [123] MASS_7.3-56                Matrix_1.5-4.1
## [125] car_3.1-0                  readr_2.1.2
## [127] cli_3.4.1                  quadprog_1.5-8
## [129] parallel_4.2.0             igraph_1.3.2
## [131] GenomicRanges_1.48.0       forcats_0.5.1
## [133] pkgconfig_2.0.3            km.ci_0.5-6
## [135] xml2_1.3.3                 foreach_1.5.2
## [137] ggtree_3.4.4               annotate_1.74.0
## [139] XVector_0.36.0             rvest_1.0.2
## [141] yulab.utils_0.0.4          stringr_1.4.0
## [143] digest_0.6.29              graph_1.74.0
## [145] Biostrings_2.64.0          rmarkdown_2.14
## [147] cellranger_1.1.0           fastmatch_1.1-3
## [149] tidytree_0.3.9             survMisc_0.5.6
## [151] dendextend_1.15.2          DelayedMatrixStats_1.18.0
## [153] GSEABase_1.58.0            rjson_0.2.21
## [155] nlme_3.1-157               lifecycle_1.0.3
## [157] jsonlite_1.8.0             Rhdf5lib_1.18.2
## [159] carData_3.0-5              viridisLite_0.4.1
```

```
## [161] limma_3.52.1              fansi_1.0.3
## [163] pillar_1.9.0              lattice_0.20-45
## [165] KEGGREST_1.36.2           fastmap_1.1.0
## [167] httr_1.4.3                GO.db_3.15.0
## [169] glue_1.6.2                png_0.1-7
## [171] iterators_1.0.14          glmnet_4.1-4
## [173] bit_4.0.4                 HDF5Array_1.24.1
## [175] ggforce_0.3.3             class_7.3-20
## [177] stringi_1.7.6             blob_1.2.3
## [179] BiocSingular_1.12.0       DESeq2_1.36.0
## [181] memoise_2.0.1             irlba_2.3.5
## [183] ape_5.6-2                 tidyverse_1.3.2
## [185] e1071_1.7-11
```

Chapter 2

RNA Data preprocessing

2.1 Loading packages

Load the IOBR package in your R session after the installation is complete:

```
library(IOBR)
library(tidyverse)
library(clusterProfiler)
```

2.2 Download array data using GEOquery

Obtaining data set from GEO Gastric cancer: GSE62254 using GEOquery R package.

```
if (!requireNamespace("GEOquery", quietly = TRUE))  BiocManager::install("GEOquery")
library("GEOquery")
# NOTE: This process may take a few minutes which depends on the internet connection s
eset_geo<-getGEO(GEO     = "GSE62254", getGPL  = F, destdir = "./")
eset    <-eset_geo[[1]]
eset    <-exprs(eset)
eset[1:5,1:5]
```

```
##           GSM1523727 GSM1523728 GSM1523729 GSM1523744 GSM1523745
## 1007_s_at  3.2176645  3.0624323  3.0279131   2.921683  2.8456013
## 1053_at    2.4050109  2.4394879  2.2442708   2.345916  2.4328582
## 117_at     1.4933412  1.8067380  1.5959665   1.839822  1.8326058
## 121_at     2.1965561  2.2812181  2.1865556   2.258599  2.1874363
```

```
## 1255_g_at  0.8698382  0.9502466  0.8125414   1.012860  0.9441993
```

2.3 Gene Annotation

Annotation of genes in the expression matrix and removal of duplicate genes.

```
# Load the annotation file `anno_hug133plus2` in IOBR.
head(anno_hug133plus2)
```

```
## # A tibble: 6 x 2
##   probe_id  symbol
##   <fct>     <fct>
## 1 1007_s_at MIR4640
## 2 1053_at   RFC2
## 3 117_at    HSPA6
## 4 121_at    PAX8
## 5 1255_g_at GUCA1A
## 6 1294_at   MIR5193
```

```
# Load the annotation file `anno_grch38` in IOBR.
head(anno_grch38)
```

```
##                id eff_length        gc entrez   symbol chr     start       end
## 1 ENSG00000000003       4536 0.3992504   7105   TSPAN6   X 100627109 100639991
## 2 ENSG00000000005       1476 0.4241192  64102     TNMD   X 100584802 100599885
## 3 ENSG00000000419       9276 0.4252911   8813     DPM1  20  50934867  50958555
## 4 ENSG00000000457       6883 0.4117391  57147    SCYL3   1 169849631 169894267
## 5 ENSG00000000460       5970 0.4298157  55732 C1orf112   1 169662007 169854080
## 6 ENSG00000000938       3382 0.5644589   2268      FGR   1  27612064  27635277
##   strand        biotype
## 1     -1 protein_coding
## 2      1 protein_coding
## 3     -1 protein_coding
## 4     -1 protein_coding
## 5      1 protein_coding
## 6     -1 protein_coding
##
## 1                                                    tetraspanin 6 [Source:HGNC
## 2                                                      tenomodulin [Source:HGNC
```

```
## 3 dolichyl-phosphate mannosyltransferase polypeptide 1, catalytic subunit [Source:HGN
## 4                                                SCY1-like, kinase-like 3 [Source:HGNC
## 5                                     chromosome 1 open reading frame 112 [Source:HGNC
## 6                            FGR proto-oncogene, Src family tyrosine kinase [Source:HGN
```

```
# Load the annotation file `anno_gc_vm32` in IOBR for mouse RNAseq data
head(anno_gc_vm32)
```

```
##                   id eff_length        gc symbol      mgi_id      gene_type
## 1 ENSMUSG00000000001       3262 0.4350092  Gnai3   MGI:95773 protein_coding
## 2 ENSMUSG00000000003        902 0.3481153   Pbsn MGI:1860484 protein_coding
## 3 ENSMUSG00000000028       3506 0.4962921  Cdc45 MGI:1338073 protein_coding
## 4 ENSMUSG00000000031       2625 0.5588571    H19   MGI:95891         lncRNA
## 5 ENSMUSG00000000037       6397 0.4377052  Scml2 MGI:1340042 protein_coding
## 6 ENSMUSG00000000049       1594 0.5050188   Apoh   MGI:88058 protein_coding
##       start       end transcript_id  ont
## 1 108014596 108053462          <NA> <NA>
## 2  76881507  76897229          <NA> <NA>
## 3  18599197  18630737          <NA> <NA>
## 4 142129262 142131886          <NA> <NA>
## 5 159865521 160041209          <NA> <NA>
## 6 108234180 108305222          <NA> <NA>
```

2.3.1 For Array data: HGU133PLUS-2 (Affaymetrix)

```
# Conduct gene annotation using `anno_hug133plus2` file; If identical gene symbols exi

eset<-anno_eset(eset       = eset,
                annotation = anno_hug133plus2,
                symbol     = "symbol",
                probe      = "probe_id",
                method     = "mean")
eset[1:5, 1:3]
```

```
##              GSM1523727 GSM1523728 GSM1523729
## SH3KBP1        4.327974   4.316195   4.351425
## RPL41          4.246149   4.246808   4.257940
## EEF1A1         4.293762   4.291038   4.262199
## COX2           4.250288   4.283714   4.270508
```

```
## LOC101928826    4.219303    4.219670    4.213252
```

2.3.2 For RNAseq data

Download RNAseq data using `UCSCXenaTools`

```
if (!requireNamespace("UCSCXenaTools", quietly = TRUE))   BiocManager::install("UCSCXena
library(UCSCXenaTools)
# NOTE: This process may take a few minutes which depends on the internet connection s
eset_stad<-XenaGenerate(subset = XenaCohorts =="GDC TCGA Stomach Cancer (STAD)") %>%
  XenaFilter(filterDatasets    = "TCGA-STAD.htseq_counts.tsv") %>%
  XenaQuery() %>%
  XenaDownload() %>%
  XenaPrepare()
eset_stad[1:5, 1:3]
```

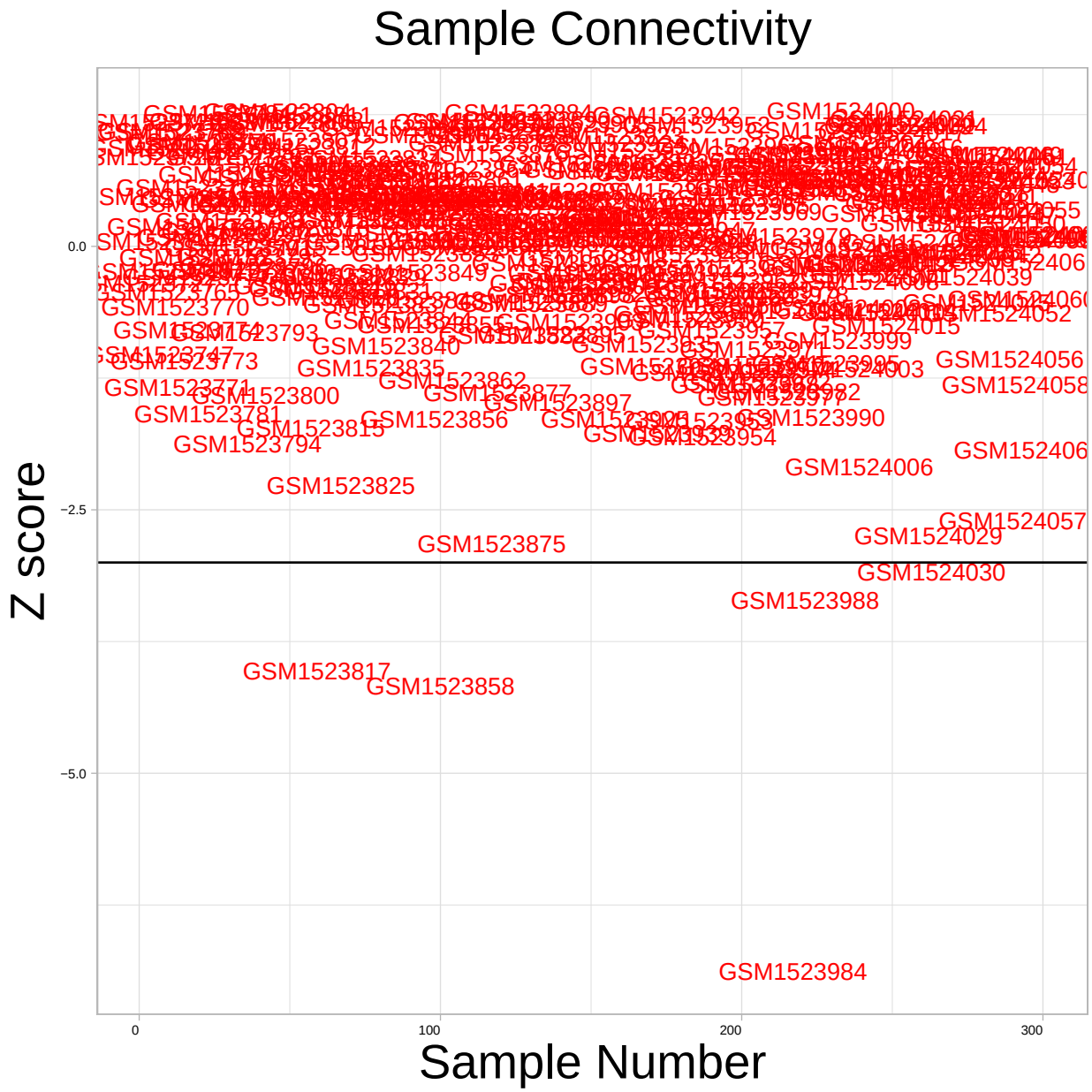
Transform gene expression matrix into TPM format, and conduct subsequent annotation.

```
# Remove the version numbers in Ensembl ID.
eset_stad$Ensembl_ID<-substring(eset_stad$Ensembl_ID, 1, 15)
eset_stad<-column_to_rownames(eset_stad, var = "Ensembl_ID")

# Revert back to original format because the data from UCSC was log2(x+1)transformed.
eset_stad<-(2^eset_stad)+1

eset_stad<-count2tpm(countMat = eset_stad, idType = "Ensembl", org="hsa", source = "loca

eset_stad[1:5,1:5]
```

2.4 Identifying outlier samples

Take ACRG microarray data for example

```
res <- find_outlier_samples(eset = eset, project = "ACRG", show_plot = TRUE)
```




```
## [1] "GSM1523817" "GSM1523858" "GSM1523984" "GSM1523988" "GSM1524030"
```

Removing potential outlier samples

```
eset1 <- eset[, !colnames(eset)%in%res]
```

2.5 PCA analysis of molecular subtypes

```
data("pdata_acrg")
res<- iobr_pca(data       = eset1,
```

```
              is.matrix   = TRUE,
              scale       = TRUE,
              is.log      = FALSE,
              pdata       = pdata_acrg,
              id_pdata    = "ID",
              group       = "Subtype",
              geom.ind    = "point",
              cols        = "normal",
              palette     = "jama",
              repel       = FALSE,
              ncp         = 5,
              axes        = c(1, 2),
              addEllipses = TRUE)
```

```
##
##       CIN       EBV       EMT        GS       MSI MSS/TP53- MSS/TP53+
##         0         0        42         0        68       106        79
## [1] ">>-- colors for PCA: #374E55FF" ">>-- colors for PCA: #DF8F44FF"
## [3] ">>-- colors for PCA: #00A1D5FF" ">>-- colors for PCA: #B24745FF"
```

```
res
```

2.6 Batch effect correction

Obtaining another data set from GEO Gastric cancer: GSE57303 using `GEOquery` R package.

```
# NOTE: This process may take a few minutes which depends on the internet connection s
eset_geo<-getGEO(GEO      = "GSE57303", getGPL  = F, destdir = "./")
eset2    <-eset_geo[[1]]
eset2    <-exprs(eset2)
eset2[1:5,1:5]
```

```
##           GSM1379261 GSM1379262 GSM1379263 GSM1379264 GSM1379265
## 1007_s_at    8.34746    9.67994    8.62643    8.59301    8.63046
```

```
## 1053_at      5.07972    4.46377    5.29685    5.78983    4.33359
## 117_at       5.65558    4.48732    4.21615    5.47984    5.20816
## 121_at       5.95123    7.09056    6.19903    5.89872    5.91323
## 1255_g_at    1.66923    1.98758    1.73083    1.56687    1.63332
```

Annotation of genes in the expression matrix and removal of duplicate genes.

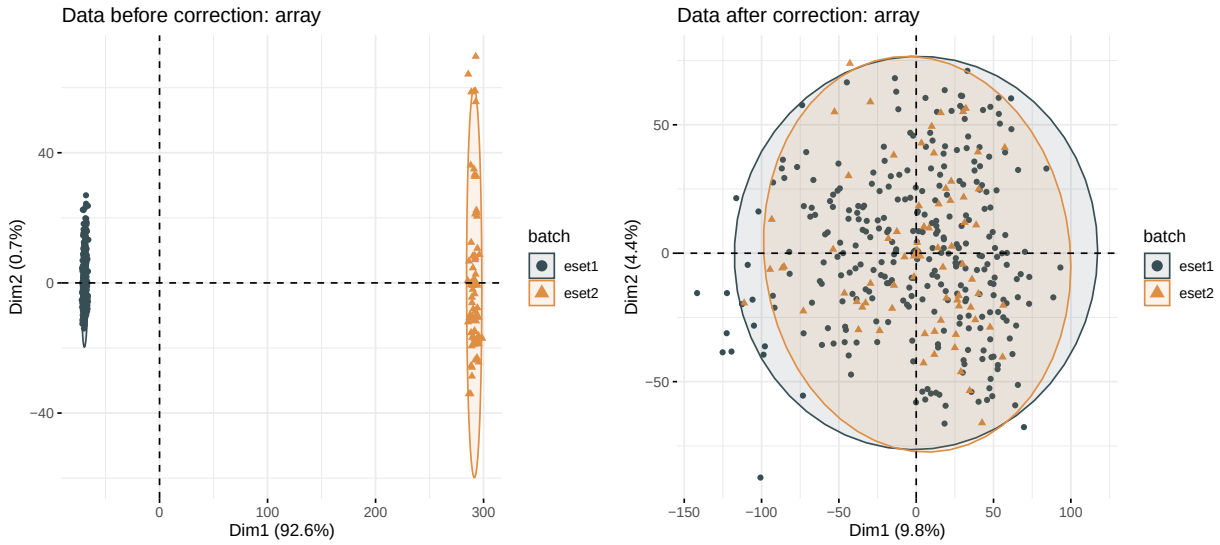
```
eset2<-anno_eset(eset       = eset2,
                 annotation = anno_hug133plus2,
                 symbol     = "symbol",
                 probe      = "probe_id",
                 method     = "mean")
eset2[1:5, 1:5]
```

```
##         GSM1379261 GSM1379262 GSM1379263 GSM1379264 GSM1379265
## ND4        13.1695    13.1804    13.0600    12.4544    13.0457
## ATP6       13.1433    13.0814    13.0502    12.4831    13.1168
## SH3KBP1    12.9390    13.1620    12.9773    12.8745    13.1169
## COX2       13.0184    13.0489    12.8621    12.7489    12.9732
## RPL41      13.0201    12.6034    12.7929    13.0153    12.9404
```

```
eset_com <- remove_batcheffect( eset1       = eset1,
                                eset2       = eset2,
                                eset3       = NULL,
                                id_type     = "symbol",
                                data_type   = "array",
                                cols        = "normal",
                                palette     = "jama",
                                log2        = TRUE,
                                check_eset  = TRUE,
                                adjust_eset = TRUE,
                                repel       = FALSE,
                                path        = "result")
```

```
##
## eset1 eset2
##   295    70
## [1] ">>-- colors for PCA: #374E55FF" ">>-- colors for PCA: #DF8F44FF"
##
## eset1 eset2
```

```
##   295    70
## [1] ">>-- colors for PCA: #374E55FF" ">>-- colors for PCA: #DF8F44FF"
```




```
dim(eset_com)
```

```
## [1] 21752   365
```

Waiting for updates: Removing batch effect of RNAseq datasets: count, combat-seq

2.7 References

Wang et al., (2019). The UCSCXenaTools R package: a toolkit for accessing genomics data from UCSC Xena platform, from cancer multi-omics to single-cell RNA-seq. Journal of Open Source Software, 4(40), 1627, https://doi.org/10.21105/joss.01627

Yuqing Zhang and others, ComBat-seq: batch effect adjustment for RNA-seq count data, NAR Genomics and Bioinformatics, Volume 2, Issue 3, September 2020, lqaa078, https://doi.org/10.1093/nargab/lqaa078

Leek, J. T., Johnson, W. E., Parker, H. S., Jaffe, A. E., & Storey, J. D. (2012). The sva package for removing batch effects and other unwanted variation in high-throughput experiments. Bioinformatics, 28(6), 882-883.

Chapter 3

Tumor ecosystem analysis

3.1 Loading packages

```
library(IOBR)
```

3.2 Downloading data for example

Obtaining data set from GEO Gastric cancer: GSE62254 using `GEOquery` R package.

```
if (!requireNamespace("GEOquery", quietly = TRUE))  BiocManager::install("GEOquery")
library("GEOquery")
# NOTE: This process may take a few minutes which depends on the internet connection s
eset_geo<-getGEO(GEO     = "GSE62254", getGPL  = F, destdir = "./")
eset    <-eset_geo[[1]]
eset    <-exprs(eset)
eset[1:5,1:5]
```

```
##           GSM1523727 GSM1523728 GSM1523729 GSM1523744 GSM1523745
## 1007_s_at  3.2176645  3.0624323  3.0279131   2.921683  2.8456013
## 1053_at    2.4050109  2.4394879  2.2442708   2.345916  2.4328582
## 117_at     1.4933412  1.8067380  1.5959665   1.839822  1.8326058
## 121_at     2.1965561  2.2812181  2.1865556   2.258599  2.1874363
## 1255_g_at  0.8698382  0.9502466  0.8125414   1.012860  0.9441993
```

3.3 Gene Annotation: HGU133PLUS-2 (Affaymetrix)

```
# Conduct gene annotation using `anno_hug133plus2` file; If identical gene symbols exi

eset<-anno_eset(eset       = eset,
                annotation = anno_hug133plus2,
                symbol     = "symbol",
                probe      = "probe_id",
                method     = "mean")
eset[1:5, 1:3]
```

```
##              GSM1523727 GSM1523728 GSM1523729
## SH3KBP1        4.327974   4.316195   4.351425
## RPL41          4.246149   4.246808   4.257940
## EEF1A1         4.293762   4.291038   4.262199
## COX2           4.250288   4.283714   4.270508
## LOC101928826   4.219303   4.219670   4.213252
```

3.4 Determine TME subtype of gastric cancer using TMEclassifier R package

```
library(TMEclassifier)
tme <- tme_classifier(eset = eset, scale = TRUE)
```

```
## Step-1: Expression data preprocessing...
## Step-2: TME deconvolution...
## Step-3: Predicting TME phenotypes...
## [19:19:48] WARNING: amalgamation/../src/learner.cc:1040:
##   If you are loading a serialized model (like pickle in Python, RDS in R) generated b
##   older XGBoost, please export the model by calling `Booster.save_model` from that ve
##   first, then load it back in current version. See:
##
##     https://xgboost.readthedocs.io/en/latest/tutorials/saving_model.html
##
##   for more details about differences between saving model and serializing.
##
## [19:19:48] WARNING: amalgamation/../src/learner.cc:749: Found JSON model saved before
```

```
## >>>--- DONE!
```

```
table(tme$TMEcluster)
```

```
##
##  IA  IE  IS
## 107  96  97
```

```
head(tme)
```

```
##           ID          IE         IS         IA TMEcluster
## 1 GSM1523727 0.204623557 0.11212681 0.68324962         IA
## 2 GSM1523728 0.009599504 0.11179146 0.87860903         IA
## 3 GSM1523729 0.852615046 0.11369089 0.03369407         IE
## 4 GSM1523744 0.053842233 0.06994632 0.87621145         IA
## 5 GSM1523745 0.055973019 0.80839488 0.13563209         IS
## 6 GSM1523746 0.545343299 0.37437568 0.08028102         IE
```

```
table(tme$TMEcluster)
```

```
##
##  IA  IE  IS
## 107  96  97
```

```
head(tme)
```

```
##           ID          IE         IS         IA TMEcluster
## 1 GSM1523727 0.204623557 0.11212681 0.68324962         IA
## 2 GSM1523728 0.009599504 0.11179146 0.87860903         IA
## 3 GSM1523729 0.852615046 0.11369089 0.03369407         IE
## 4 GSM1523744 0.053842233 0.06994632 0.87621145         IA
## 5 GSM1523745 0.055973019 0.80839488 0.13563209         IS
## 6 GSM1523746 0.545343299 0.37437568 0.08028102         IE
```

3.5 DEG analysis: method1

Differential analysis of selected immune-activated and immune-expelled gastric cancers

```
pdata <- tme[!tme$TMEcluster=="IS", ]
deg  <-   iobr_deg(eset          = eset,
                   annotation     = NULL,
                   pdata         = pdata,
```

```
                    group_id     = "TMEcluster",
                    pdata_id     = "ID",
                    array        = TRUE,
                    method       = "limma",
                    contrast     = c("IA","IE"),
                    path         = NULL,
                    padj_cutoff  = 0.01,
                    logfc_cutoff = 0.5)
```

```
## >>>== Matching grouping information and expression matrix

## >>>== limma was selected for differential gene analysis of Array data

##
## Attaching package: 'limma'

## The following object is masked from 'package:BiocGenerics':
##
##     plotMA

## group1 = IE

## group2 = NA

## # A tibble: 6 x 11
##   symbol  log2FoldChange AveExpr     t   pvalue     padj     B sigORnot    label
##   <chr>            <dbl>   <dbl> <dbl>    <dbl>    <dbl> <dbl> <chr>       <chr>
## 1 TMEM100          0.774    1.84  13.9 2.47e-31 5.37e-27  60.4 Up_regulat~ Both
## 2 ABCA8            0.933    1.90  12.9 3.11e-28 3.38e-24  53.4 Up_regulat~ Both
## 3 HHIP             0.613    1.73  12.1 7.62e-26 4.46e-22  48.0 Up_regulat~ Both
## 4 LMNB2           -0.287    2.25 -12.1 9.28e-26 4.46e-22  47.8 NOT         Sign~
## 5 MCM6            -0.211    3.02 -12.1 1.02e-25 4.46e-22  47.7 NOT         Sign~
## 6 ADH1B            0.907    1.86  12.0 2.27e-25 7.04e-22  47.0 Up_regulat~ Both
## # i 2 more variables: IE <dbl>, `` <dbl>
```

3.6 GSEA analysis based on differential express gene analysis results

Select the gene set list in IOBR's signature collection.

```
head(deg)
```

```
## # A tibble: 6 x 11
##   symbol  log2FoldChange AveExpr     t   pvalue     padj     B sigORnot    label
##   <chr>            <dbl>   <dbl> <dbl>    <dbl>    <dbl> <dbl> <chr>       <chr>
## 1 TMEM100          0.774    1.84  13.9 2.47e-31 5.37e-27  60.4 Up_regulat~ Both
## 2 ABCA8            0.933    1.90  12.9 3.11e-28 3.38e-24  53.4 Up_regulat~ Both
## 3 HHIP             0.613    1.73  12.1 7.62e-26 4.46e-22  48.0 Up_regulat~ Both
## 4 LMNB2           -0.287    2.25 -12.1 9.28e-26 4.46e-22  47.8 NOT         Sign~
## 5 MCM6            -0.211    3.02 -12.1 1.02e-25 4.46e-22  47.7 NOT         Sign~
## 6 ADH1B            0.907    1.86  12.0 2.27e-25 7.04e-22  47.0 Up_regulat~ Both
## # i 2 more variables: IE <dbl>, `` <dbl>
```

```
sig_list <- signature_collection[c("TMEscoreB_CIR", "TMEscoreA_CIR", "DNA_replication",
                                   "Pan_F_TBRs", "TGFb.myCAF", "Ferroptosis", "TLS_Natur
sig_list
```

```
## $TMEscoreB_CIR
##   [1] "DCN"          "SEPP1"        "ACTA2"        "SPARCL1"      "BEX3"
##   [6] "MYLK"         "AKR1C1"       "TIMP2"        "MXRA7"        "C11orf96"
##  [11] "CAV1"         "PDGFRA"       "FHL1"         "MGP"          "EID1"
##  [16] "LOC101930400" "DST"          "GREM1"        "FERMT2"       "TNC"
##  [21] "CYBRD1"       "LTBP1"        "ACTG2"        "TMEM47"       "SERPINE2"
##  [26] "ANTXR2"       "GNG11"        "TAGLN"        "GSTA4"        "PKIG"
##  [31] "MAOA"         "PTRF"         "FAM3B"        "PBX1"         "WLS"
##  [36] "SELM"         "SVIL"         "MYH11"        "AGT"          "SPON1"
##  [41] "TGFB1I1"      "PDLIM3"       "PDK4"         "SYNPO2"       "MSRB3"
##  [46] "PROS1"        "EDNRA"        "AKAP12"       "PSD3"         "TNS1"
##  [51] "JAM3"         "PDZRN3"       "DDR2"         "HMGCS2"       "SGCE"
##  [56] "MRVI1"        "WFDC1"        "FBLN1"        "FMO5"         "MAOB"
##  [61] "AMOTL1"       "AKT3"         "CNRIP1"       "CPE"          "MAP1B"
##  [66] "RBP1"         "GNAI1"        "FOXF2"        "SORBS2"       "ZCCHC24"
##  [71] "ZNF704"       "ARMCX1"       "DIXDC1"       "SSTR1"        "THRB"
##  [76] "C3orf70"      "PKIB"         "CNN1"         "SYTL5"        "DACT1"
##  [81] "SYNPO"        "GAS1"         "DPYSL3"       "CCDC80"       "TSPYL5"
##  [86] "DCHS1"        "SOBP"         "AOC3"         "NDN"          "FGF7P3"
##  [91] "SMAD9"        "MCC"          "CLMP"         "MYL9"         "RBP4"
##  [96] "PLN"          "SPOCK1"       "COL14A1"      "CRYAB"        "SRPX"
```

```
## [101] "EML1"          "RERG"         "PPP1R3C"      "LOC100506718" "CH25H"
## [106] "HSPB8"         "PID1"         "TTC28"        "STON1"        "ABCG2"
## [111] "ZSCAN18"       "SCIN"         "C14orf132"    "TMEM55A"      "WASF3"
## [116] "PAPLN"         "COLEC12"      "ACKR1"        "TMEM150C"     "RAI2"
## [121] "TSPAN7"        "MRGPRF"       "ABCA8"        "CHIC1"        "NBEA"
## [126] "FAM13C"        "SETBP1"       "LDOC1"        "TMEM100"      "LOC101930349"
## [131] "PRICKLE2"      "TSPAN18"      "FABP4"        "ARHGEF26"     "ERICH5"
## [136] "MYOCD"         "BEX2"         "PPP1R14A"     "FGF13"        "RUNX1T1"
## [141] "MAGI2-AS3"     "LINC01279"    "REEP1"        "PLAC9"        "MYEF2"
## [146] "PRKD1"         "RGN"          "CLDN11"       "ANK2"         "ESRRG"
## [151] "SYNC"          "ZNF667-AS1"   "FGF7"         "SFRP1"        "HMCN1"
## [156] "TCEAL7"        "OGN"          "MAGI2"        "MIR100HG"     "FILIP1"
## [161] "LOC100507334"  "ANKRD6"       "PLEKHH2"      "ZNF542P"      "ARMCX4"
## [166] "NOV"           "DCLK1"        "ARHGAP28"     "C2orf40"      "TRHDE"
## [171] "EPHA7"         "SCRG1"        "ZNF677"       "ZFPM2"        "PEG3"
## [176] "SERP2"         "ZNF415"       "MAMDC2"       "RBM24"        "MEOX2"
##
## $TMEscoreA_CIR
##  [1] "HLA-DPB1"       "UBD"            "LOC100509457"   "WARS"
##  [5] "TAP1"           "HLA-DMA"        "TRIM22"         "PSAT1"
##  [9] "CXCL10"         "SOCS3"          "CXCL9"          "PBK"
## [13] "CCL4"           "CCL5"           "BCL2A1"         "TRBC1"
## [17] "IDO1"           "NFE2L3"         "CCL3L3"         "DTL"
## [21] "MMP9"           "SLC2A3"         "ZNF367"         "RCC1"
## [25] "STIL"           "TRAC"           "HELLS"          "GZMB"
## [29] "RTEL1-TNFRSF6B" "CXCL11"         "GBP5"           "CD2"
## [33] "CDCA2"          "CDT1"           "TNFAIP2"        "TYMP"
## [37] "MICB"           "SLC2A14"        "GZMK"           "CD8A"
## [41] "CENPH"          "MND1"           "BATF2"          "BRIP1"
## [45] "E2F7"           "KIF18A"         "AIM2"           "ETV7"
## [49] "ITK"            "GNLY"           "GPR171"         "WDHD1"
## [53] "GBP4"           "MB21D1"         "NLRP3"          "MCEMP1"
## [57] "POLR3G"         "NLRC3"          "KLRC2"          "CLEC5A"
## [61] "ARHGAP11A"      "GPR84"          "IFNG"           "ZBED2"
##
## $DNA_replication
##  [1] "RNASEH2A" "POLD3"    "DNA2"     "FEN1"     "POLA2"    "RNASEH1"
```

```
##  [7] "RPA4"     "LIG1"     "MCM2"     "MCM3"     "MCM4"     "MCM5"
## [13] "MCM6"     "MCM7"     "PCNA"     "POLE3"    "POLA1"    "POLD1"
## [19] "POLD2"    "POLE"     "POLE2"    "PRIM1"    "PRIM2"    "POLE4"
## [25] "POLD4"    "RFC1"     "RFC2"     "RFC3"     "RFC4"     "RFC5"
## [31] "RPA1"     "RPA2"     "RPA3"     "SSBP1"    "RNASEH2B" "RNASEH2C"
##
## $Base_excision_repair
##  [1] "PARP2" "PARP3" "POLD3" "PARP1" "PARP4" "FEN1"  "SMUG1" "NEIL2" "APEX2"
## [10] "POLL"  "HMGB1" "APEX1" "LIG1"  "LIG3"  "MPG"   "MUTYH" "NTHL1" "OGG1"
## [19] "PCNA"  "POLE3" "POLB"  "POLD1" "POLD2" "POLE"  "POLE2" "NEIL3" "POLE4"
## [28] "POLD4" "UNG"   "XRCC1" "NEIL1" "MBD4"
##
## $Pan_F_TBRs
##  [1] "ACTA2"    "ACTG2"    "ADAM12"   "ADAM19"   "CNN1"     "COL4A1"
##  [7] "CTGF"     "CTPS1"    "FAM101B"  "FSTL3"    "HSPB1"    "IGFBP3"
## [13] "PXDC1"    "SEMA7A"   "SH3PXD2A" "TAGLN"    "TGFBI"    "TNS1"
## [19] "TPM1"
##
## $TGFb.myCAF
##  [1] "CST1"    "LAMP5"   "LOXL1"   "EDNRA"   "TGFB1"   "TGFB3"   "TNN"
##  [8] "CST2"    "HES4"    "COL10A1" "ELN"     "THBS4"   "NKD2"    "OLFM2"
## [15] "COL6A3"  "LRRC17"  "COL3A1"  "THY1"    "HTRA3"   "TMEM204" "11-Sep"
## [22] "COMP"    "TNFAIP6" "ID4"     "GGT5"    "INAFM1"  "CILP"    "OLFML2B"
##
## $Ferroptosis
##  [1] "ACSL4"      "AKR1C1-3"   "ALOXs"      "ATP5G3"     "CARS"
##  [6] "CBS"        "CD44v"      "CHAC1"      "CISD1"      "CS"
## [11] "DPP4"       "FANCD2"     "GCLC/GCLM"  "GLS2"       "GPX4"
## [16] "GSS"        "HMGCR"      "HSPB1/5"    "KOD"        "LPCAT3"
## [21] "MT1G"       "NCOA4"      "NFE2L2"     "PTGS2"      "RPL8"
## [26] "SAT1"       "SLC7A11"    "SQS"        "TFRC"       "TP53"
## [31] "TTC35/EMC2" "MESH1"
##
## $TLS_Nature
## [1] "CD79B"  "CD1D"   "CCR6"   "LAT"    "SKAP1"  "CETP"   "EIF1AY" "RBP5"
## [9] "PTGDS"
##
```

```
## $Glycolysis
##  [1] "ACSS1"   "ACSS2"   "ADH1A"   "ADH1B"   "ADH1C"   "ADH4"    "ADH5"
##  [8] "ADH6"    "ADH7"    "ADPGK"   "AKR1A1"  "ALDH1A3" "ALDH1B1" "ALDH2"
## [15] "ALDH3A1" "ALDH3A2" "ALDH3B1" "ALDH3B2" "ALDH7A1" "ALDH9A1" "ALDOA"
## [22] "ALDOB"   "ALDOC"   "BPGM"    "DLAT"    "DLD"     "ENO1"    "ENO2"
## [29] "ENO3"    "FBP1"    "FBP2"    "G6PC"    "G6PC2"   "GALM"    "GAPDH"
## [36] "GAPDHS"  "GCK"     "GPI"     "HK1"     "HK2"     "HK3"     "HKDC1"
## [43] "LDHA"    "LDHAL6A" "LDHAL6B" "LDHB"    "LDHC"    "PANK1"   "PCK1"
## [50] "PCK2"    "PDHA1"   "PDHA2"   "PDHB"    "PFKFB1"  "PFKFB2"  "PFKFB3"
## [57] "PFKFB4"  "PFKL"    "PFKM"    "PFKP"    "PGAM1"   "PGAM2"   "PGAM4"
## [64] "PGK1"    "PGK2"    "PGM1"    "PGM2"    "PKLR"    "PKM"     "SLC2A2"
## [71] "TPI1"
```

```
gsea<-     sig_gsea(deg,
                    genesets          = sig_list,
                    path              = "GSEA",
                    gene_symbol       = "symbol",
                    logfc             = "log2FoldChange",
                    org               = "hsa",
                    show_plot         = FALSE,
                    msigdb            = TRUE,
                    category          = "H",
                    subcategory       = NULL,
                    palette_bar       = "set2")
```

Hallmark gene signatures

```
gsea<-     sig_gsea(deg,
                    genesets          = NULL,
                    path              = "GSEA",
                    gene_symbol       = "symbol",
                    logfc             = "log2FoldChange",
                    org               = "hsa",
                    show_plot         = FALSE,
                    msigdb            = TRUE,
                    category          = "H",
                    subcategory       = NULL,
                    palette_bar       = "aaas",
                    show_bar          = 5,
```

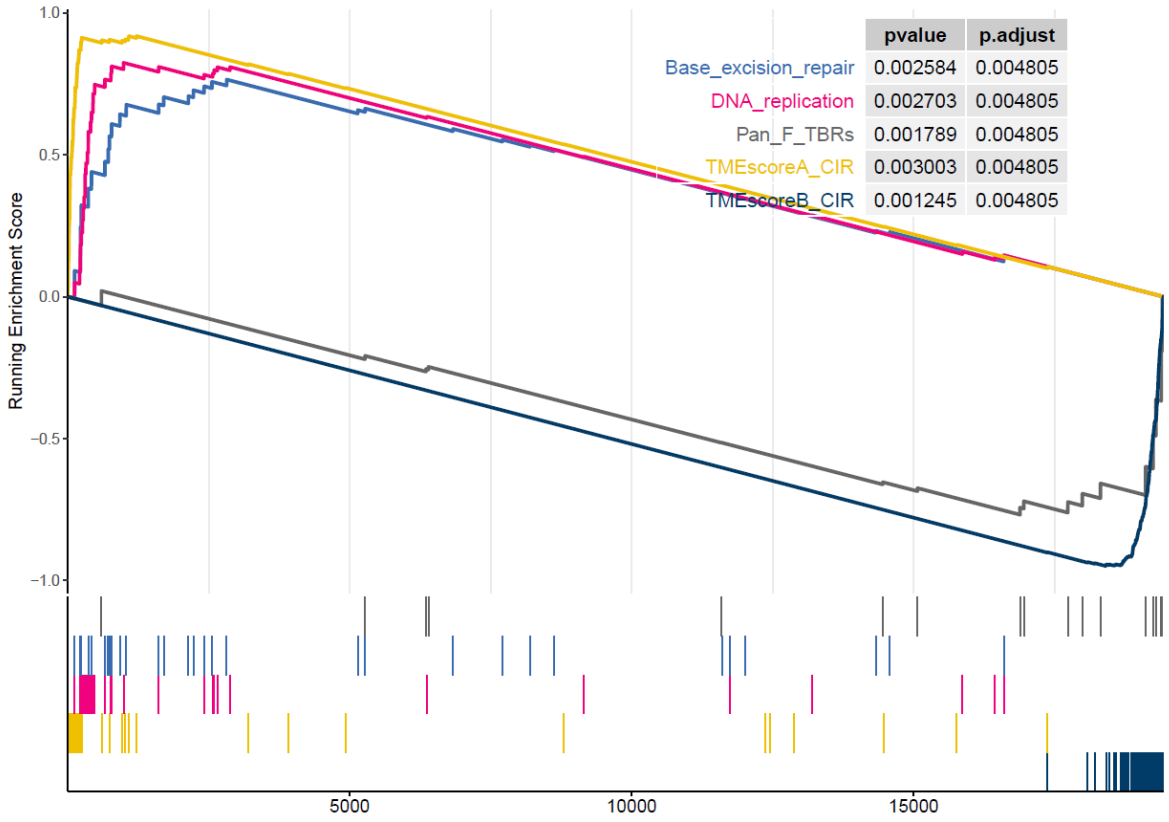


Figure 3.1: GSEA of TME gent sets

```
                    show_gsea          = 6)
```

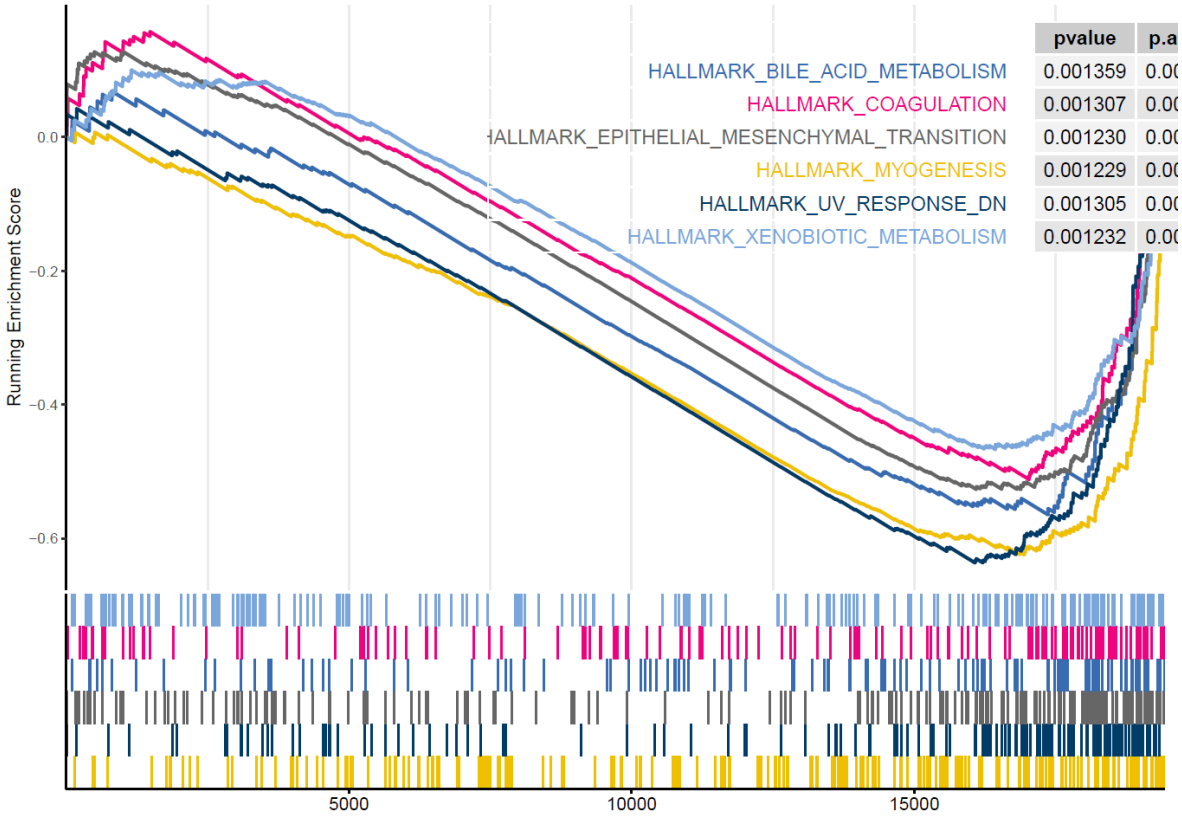


Figure 3.2: GSEA of Hallmark gent sets

3.7 DEG analysis: method2

Identifing TME subtype-related differential genes using `find_markers_in_bulk`

```
library(Seurat)
res <- find_markers_in_bulk(pdata      = tme,
                            eset       = eset,
                            group      = "TMEcluster",
                            nfeatures  = 2000,
                            top_n      = 20,
                            thresh.use = 0.15,
                            only.pos   = TRUE,
                            min.pct    = 0.10)
```

```
##
```

```
##  IA  IE  IS
## 107  96  97
## # A tibble: 56 x 7
## # Groups:   cluster [3]
##       p_val avg_log2FC pct.1 pct.2 p_val_adj cluster gene
##       <dbl>      <dbl> <dbl> <dbl>     <dbl> <fct>   <chr>
##  1 3.29e-20      0.218     1     1  7.15e-16 IA      IFNG
##  2 1.81e-18      0.172     1     1  3.93e-14 IA      CXCL10
##  3 1.01e-16      0.183     1     1  2.20e-12 IA      GZMB
##  4 2.82e-16      0.251     1     1  6.12e-12 IA      CXCL11
##  5 9.68e-16      0.170     1     1  2.10e-11 IA      CXCL9
##  6 3.11e-15      0.221     1     1  6.77e-11 IA      IDO1
##  7 9.90e-15      0.156     1     1  2.15e-10 IA      POLR3G
##  8 3.02e-14      0.184     1     1  6.57e-10 IA      GBP4
##  9 9.23e-14      0.152     1     1  2.01e- 9 IA      ZBED2
## 10 9.79e-12      0.155     1     1  2.13e- 7 IA      GNLY
## # i 46 more rows
```

```
top15 <-  res$top_markers %>% dplyr:: group_by(cluster) %>%  dplyr::top_n(15, avg_log2F
top15$gene
```

```
##  [1] "IFNG"          "CXCL10"        "GZMB"          "CXCL11"
##  [5] "CXCL9"         "IDO1"          "POLR3G"        "GBP4"
##  [9] "GNLY"          "PLEKHS1"       "KLRC2"         "VSNL1"
## [13] "AIM2"          "SLCO1B3"       "COL11A1"       "TMEM100"
## [17] "ADH1B"         "ABCA8"         "MAMDC2"        "C1QTNF7"
## [21] "SCN7A"         "C7"            "C2orf40"       "LIPF"
## [25] "PGA4"          "SCRG1"         "OGN"           "GKN1"
## [29] "GKN2"          "GIF"           "IL1A"          "EREG"
## [33] "PPBP"          "IL11"          "CXCL6"         "PI15"
## [37] "PROK2"         "HCAR3"         "CLEC5A"        "MAGEA10-MAGEA5"
## [41] "MAGEA4"        "MAGEA12"       "MAGEA6"        "MAGEA2B"
## [45] "REG1B"
```

Heatmap visualisation using `Seurat`'s `DoHeatmap`

```
#
cols <- c('#2692a4','#fc0d3a','#ffbe0b')
p1 <- DoHeatmap(res$sce, top15$gene, group.colors = cols )+
```

```
  scale_fill_gradientn(colours = rev(colorRampPalette(RColorBrewer::brewer.pal(11,"RdBu
```

Extracting variables from the expression matrix to merge with TME subtype

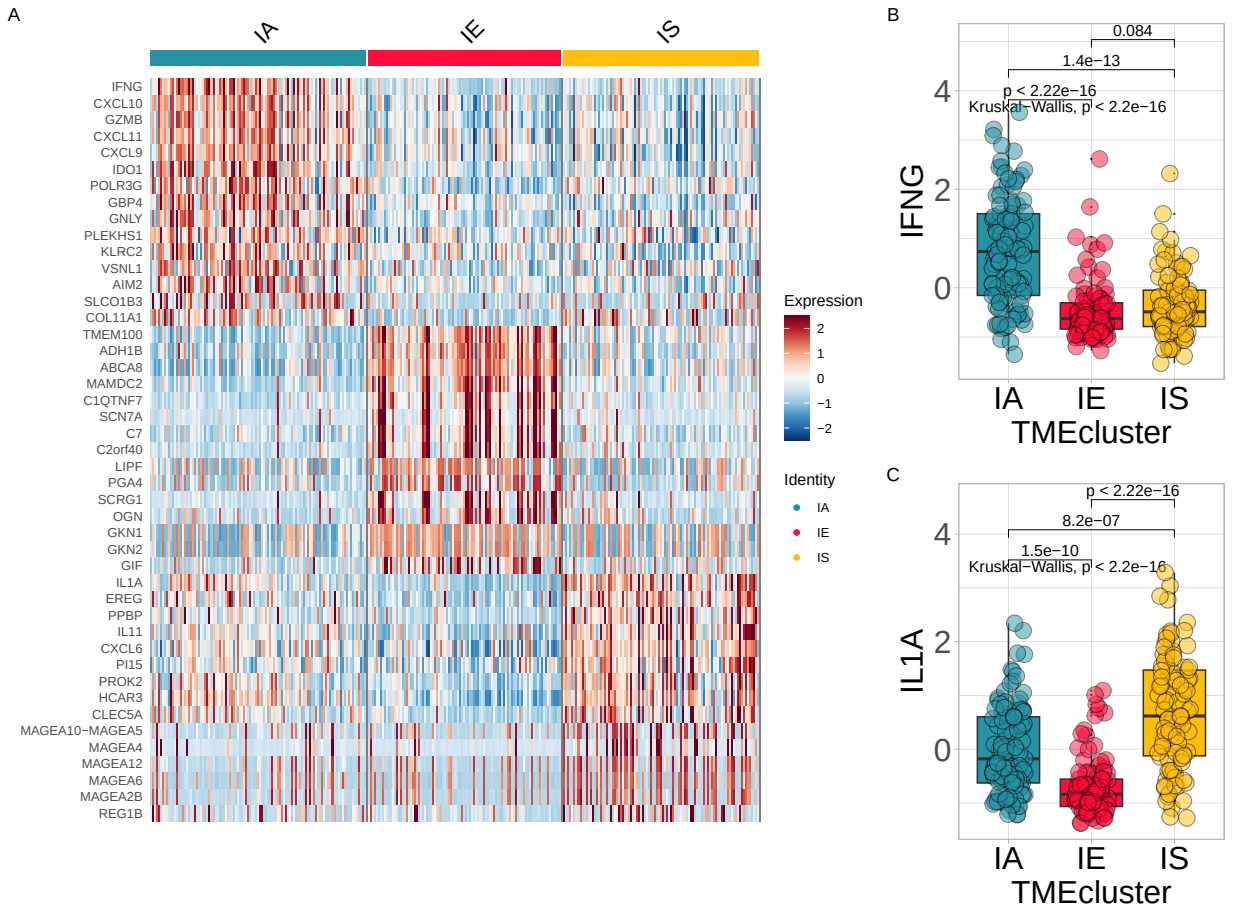
```
input <- combine_pd_eset(eset = eset, pdata = tme, feas = top15$gene, scale = T)
p2 <- sig_box(input, variable = "TMEcluster", signature = "IFNG", jitter = TRUE,
              cols =  cols, show_pvalue = TRUE, size_of_pvalue = 4)
```

```
## # A tibble: 3 x 8
##   .y.       group1 group2        p    p.adj p.format p.signif method
##   <chr>     <chr>  <chr>     <dbl>    <dbl> <chr>    <chr>    <chr>
## 1 signature IA     IE     4.09e-17 1.20e-16 < 2e-16  ****     Wilcoxon
## 2 signature IA     IS     1.44e-13 2.90e-13 1.4e-13  ****     Wilcoxon
## 3 signature IE     IS     8.35e- 2 8.4 e- 2 0.084    ns       Wilcoxon
```

```
p3 <- sig_box(input, variable = "TMEcluster", signature = "IL1A",
              jitter = TRUE, cols =  cols, show_pvalue = TRUE, size_of_pvalue = 4)
```

```
## # A tibble: 3 x 8
##   .y.       group1 group2        p    p.adj p.format p.signif method
##   <chr>     <chr>  <chr>     <dbl>    <dbl> <chr>    <chr>    <chr>
## 1 signature IA     IE     1.46e-10 2.90e-10 1.5e-10  ****     Wilcoxon
## 2 signature IA     IS     8.22e- 7 8.2 e- 7 8.2e-07  ****     Wilcoxon
## 3 signature IE     IS     4.90e-20 1.5 e-19 < 2e-16  ****     Wilcoxon
```

```
if (!requireNamespace("patchwork", quietly = TRUE))   install.packages("patchwork")
library(patchwork)
p <- (p1|p2/p3) + plot_layout(widths = c(2.3,1))
p + plot_annotation(tag_levels = 'A')
```



3.8 Identifying signatures associated with TME clusters

Calculate TME associated signatures-(through PCA method).

```
sig_tme<-calculate_sig_score(pdata           = NULL,
                             eset            = eset,
                             signature       = signature_collection,
                             method          = "pca",
                             mini_gene_count = 2)
sig_tme <- t(column_to_rownames(sig_tme, var = "ID"))
sig_tme[1:5, 1:3]
```

```
##                   GSM1523727 GSM1523728 GSM1523729
## CD_8_T_effector   -2.5513794  0.7789141 -2.1770675
## DDR               -0.8747614  0.7425162 -1.3272054
```

```
## APM                1.1098368  2.1988688 -0.9516419
## Immune_Checkpoint -2.3701787  0.9455120 -1.4844104
## CellCycle_Reg      0.1063358  0.7583302 -0.3649795
```

TMEcluster

```
res <- find_markers_in_bulk(pdata = tme, eset = sig_tme, group = "TMEcluster", nfeatures
```

```
##
##  IA  IE  IS
## 107  96  97
## # A tibble: 58 x 7
## # Groups:   cluster [3]
##       p_val avg_log2FC pct.1 pct.2 p_val_adj cluster gene
##       <dbl>      <dbl> <dbl> <dbl>     <dbl> <fct>   <chr>
##  1 6.21e-19       4.07 0.701 0.316  1.59e-16 IA      IFNG-signature-Ayers-et-al
##  2 1.82e-17       5.18 0.813 0.399  4.66e-15 IA      Th2-cells-Bindea-et-al
##  3 2.60e-16       4.65 0.813 0.383  6.65e-14 IA      Folate-One-Carbon-Metaboli~
##  4 1.52e-15       5.12 0.804 0.352  3.89e-13 IA      Homologous-recombination
##  5 4.95e-15       4.84 0.673 0.275  1.27e-12 IA      CD-8-T-effector
##  6 7.70e-15       3.16 0.71  0.332  1.97e-12 IA      Th1-cells-Bindea-et-al
##  7 9.52e-14       3.16 0.822 0.399  2.44e-11 IA      Purine-Biosynthesis
##  8 1.42e-13       2.83 0.664 0.352  3.63e-11 IA      ADP-Ribosylation
##  9 5.96e-13       2.96 0.785 0.42   1.53e-10 IA      TIP-Release-of-cancer-cell~
## 10 3.00e-12       2.71 0.776 0.409  7.68e-10 IA      Glycine--Serine-and-Threon~
## # i 48 more rows
```

```
top15 <-  res$top_markers %>% dplyr:: group_by(cluster) %>%  dplyr::top_n(15, avg_log2F

p1 <- DoHeatmap(res$sce, top15$gene, group.colors = cols)+
  scale_fill_gradientn(colours = rev(colorRampPalette(RColorBrewer::brewer.pal(11,"RdBu
```

```
top15$gene  <- gsub(top15$gene, pattern = "\\-", replacement = "\\_")
input <- combine_pd_eset(eset = sig_tme, pdata = tme, feas = top15$gene, scale = T)

p2 <- sig_box(input, variable = "TMEcluster", signature = "IFNG_signature_Ayers_et_al",
              cols =  cols, show_pvalue = TRUE, size_of_pvalue = 4, size_of_font = 6)
```

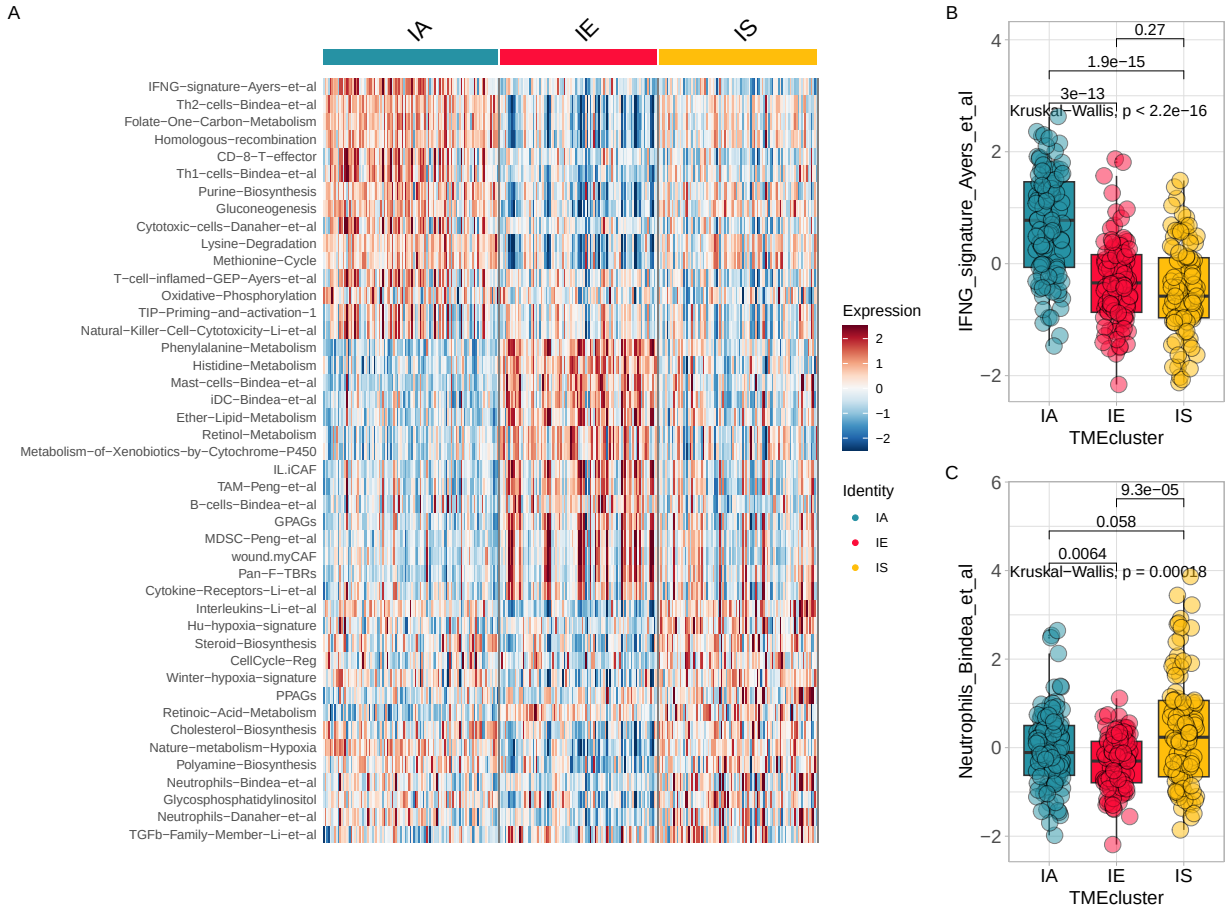
```
## # A tibble: 3 x 8
```

```
##   .y.       group1 group2        p   p.adj p.format p.signif method
##   <chr>     <chr>  <chr>     <dbl>   <dbl> <chr>    <chr>    <chr>
## 1 signature IA     IE     2.98e-13 6  e-13 3.0e-13  ****     Wilcoxon
## 2 signature IA     IS     1.85e-15 5.6e-15 1.9e-15  ****     Wilcoxon
## 3 signature IE     IS     2.68e- 1 2.7e- 1 0.27     ns       Wilcoxon
```

```
p3 <- sig_box(input, variable = "TMEcluster", signature = "Neutrophils_Bindea_et_al",
              jitter = TRUE, cols =  cols, show_pvalue = TRUE, size_of_pvalue = 4, size_
```

```
## # A tibble: 3 x 8
##   .y.       group1 group2         p   p.adj p.format p.signif method
##   <chr>     <chr>  <chr>      <dbl>   <dbl> <chr>    <chr>    <chr>
## 1 signature IA     IE     0.00639   0.013   0.0064   **       Wilcoxon
## 2 signature IA     IS     0.0584    0.058   0.0584   ns       Wilcoxon
## 3 signature IE     IS     0.0000929 0.00028 9.3e-05  ****     Wilcoxon
```

```
p <- (p1|p2/p3) + plot_layout(widths = c(2.3,1))
p + plot_annotation(tag_levels = 'A')
```



```
library(survminer)
data(pdata_acrg, package = "IOBR")
input <- merge(pdata_acrg, input, by = "ID")
p1<-surv_group(input_pdata       = input,
               target_group      = "TMEcluster",
               ID                = "ID",
               reference_group   = "High",
               project           = "ACRG",
               cols              = cols,
               time              = "OS_time",
               status            = "OS_status",
               time_type         = "month",
               save_path         = "result")
```

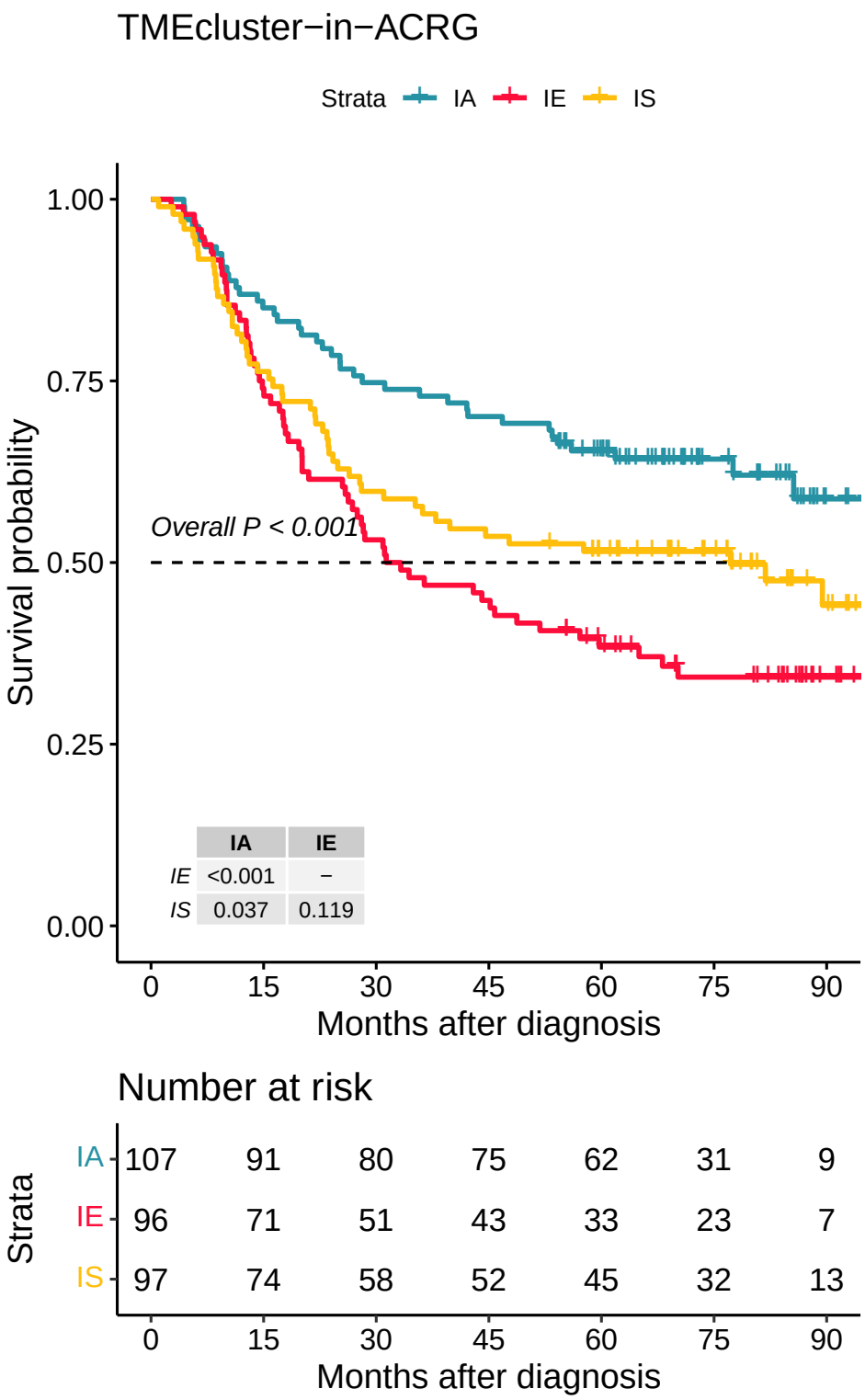
```
## >>> Dataset's survival follow up time is range between 1 to 105.7 months
```

```
##  IA  IE  IS
## 107  96  97
```

```
## 1079697
```

```
##   Maximum of follow up time is 105.7 months; and will be divided into 6 sections;
```

```
p1
```




```
p1<- percent_bar_plot(input, x = "TMEcluster" , y = "Subtype", palette = "jama")
```

```
## # A tibble: 12 x 5
## # Groups:   TMEcluster [3]
```

```
##    TMEcluster Subtype    Freq  Prop count
##    <chr>      <fct>     <dbl> <dbl> <dbl>
##  1 IA         EMT           7  0.07   107
##  2 IA         MSI          49  0.46   107
##  3 IA         MSS/TP53-    27  0.25   107
##  4 IA         MSS/TP53+    24  0.22   107
##  5 IE         EMT          24  0.25    96
##  6 IE         MSI           3  0.03    96
##  7 IE         MSS/TP53-    40  0.42    96
##  8 IE         MSS/TP53+    29  0.3     96
##  9 IS         EMT          15  0.15    97
## 10 IS         MSI          16  0.16    97
## 11 IS         MSS/TP53-    40  0.41    97
## 12 IS         MSS/TP53+    26  0.27    97
## [1] "'#374E55FF', '#DF8F44FF', '#00A1D5FF', '#B24745FF', '#79AF97FF', '#6A6599FF', '#
```

```
p2<- percent_bar_plot(input, x = "TMEcluster" , y = "Lauren", palette = "jama")
```

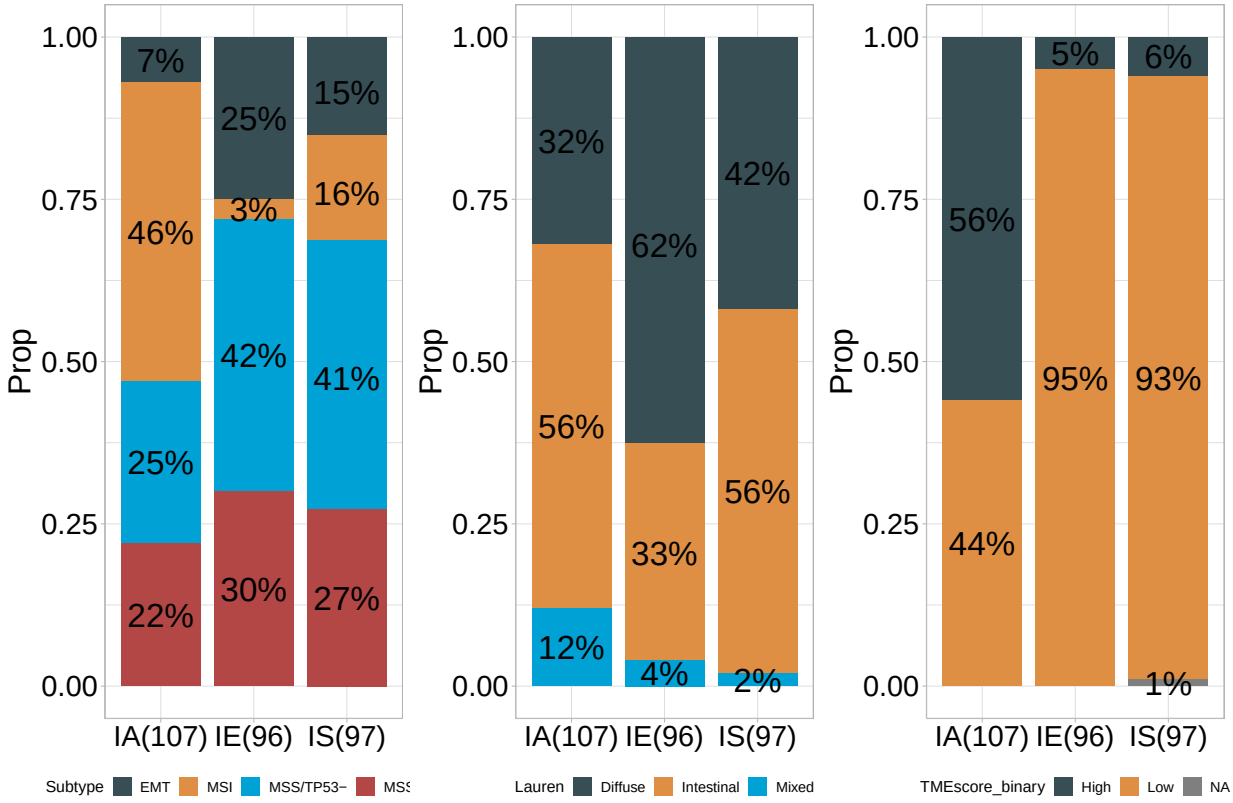
```
## # A tibble: 9 x 5
## # Groups:   TMEcluster [3]
##   TMEcluster Lauren      Freq  Prop count
##   <chr>      <fct>      <dbl> <dbl> <dbl>
## 1 IA         Diffuse       34  0.32   107
## 2 IA         Intestinal    60  0.56   107
## 3 IA         Mixed         13  0.12   107
## 4 IE         Diffuse       60  0.62    96
## 5 IE         Intestinal    32  0.33    96
## 6 IE         Mixed          4  0.04    96
## 7 IS         Diffuse       41  0.42    97
## 8 IS         Intestinal    54  0.56    97
## 9 IS         Mixed          2  0.02    97
## [1] "'#374E55FF', '#DF8F44FF', '#00A1D5FF', '#B24745FF', '#79AF97FF', '#6A6599FF', '#
```

```
p3<- percent_bar_plot(input, x = "TMEcluster" , y = "TMEscore_binary", palette = "jama")
```

```
## # A tibble: 7 x 5
## # Groups:   TMEcluster [3]
##   TMEcluster TMEscore_binary  Freq  Prop count
##   <chr>      <fct>           <dbl> <dbl> <dbl>
```

```
## 1 IA          High                60  0.56    107
## 2 IA          Low                 47  0.44    107
## 3 IE          High                 5  0.05     96
## 4 IE          Low                 91  0.95     96
## 5 IS          High                 6  0.06     97
## 6 IS          Low                 90  0.93     97
## 7 IS          <NA>                 1  0.01     97
## [1] "'#374E55FF', '#DF8F44FF', '#00A1D5FF', '#B24745FF', '#79AF97FF', '#6A6599FF', '#
```

```
p1|p2|p3
```



Chapter 4

Signature and relevant phenotypes

4.1 Loading packages

Load the IOBR package in your R session after the installation is complete:

```
library(IOBR)
library(survminer)
library(tidyverse)
```

4.2 Downloading data for example

Obtaining data set from GEO Gastric cancer: GSE62254 using `GEOquery` R package.

```
if (!requireNamespace("GEOquery", quietly = TRUE))  BiocManager::install("GEOquery")
library("GEOquery")
# NOTE: This process may take a few minutes which depends on the internet connection s
eset_geo <- getGEO(GEO = "GSE62254", getGPL  = F, destdir = "./")
eset    <-eset_geo[[1]]
eset    <-exprs(eset)
eset[1:5,1:5]
```

```
##           GSM1523727 GSM1523728 GSM1523729 GSM1523744 GSM1523745
## 1007_s_at  3.2176645  3.0624323  3.0279131   2.921683  2.8456013
## 1053_at    2.4050109  2.4394879  2.2442708   2.345916  2.4328582
## 117_at     1.4933412  1.8067380  1.5959665   1.839822  1.8326058
## 121_at     2.1965561  2.2812181  2.1865556   2.258599  2.1874363
```

```
## 1255_g_at  0.8698382  0.9502466  0.8125414   1.012860  0.9441993
```

Annotation of genes in the expression matrix and removal of duplicate genes.

```
# Load the annotation file `anno_hug133plus2` in IOBR.
head(anno_hug133plus2)
```

```
## # A tibble: 6 x 2
##   probe_id  symbol
##   <fct>     <fct>
## 1 1007_s_at MIR4640
## 2 1053_at   RFC2
## 3 117_at    HSPA6
## 4 121_at    PAX8
## 5 1255_g_at GUCA1A
## 6 1294_at   MIR5193
```

```
# Conduct gene annotation using `anno_hug133plus2` file; If identical gene symbols exi

eset<-anno_eset(eset       = eset,
                annotation = anno_hug133plus2,
                symbol     = "symbol",
                probe      = "probe_id",
                method     = "mean")
eset[1:5, 1:3]
```

```
##              GSM1523727 GSM1523728 GSM1523729
## SH3KBP1        4.327974   4.316195   4.351425
## RPL41          4.246149   4.246808   4.257940
## EEF1A1         4.293762   4.291038   4.262199
## COX2           4.250288   4.283714   4.270508
## LOC101928826   4.219303   4.219670   4.213252
```

4.3 Signature score estimation

4.3.1 Signature collection of IOBR

```
# Return available parameter options of signature estimation.
signature_score_calculation_methods
```

```
##           PCA        ssGSEA       z-score   Integration
##         "pca"      "ssgsea"      "zscore" "integration"
```

```
#TME associated signatures
names(signature_tme)[1:20]
```

```
##  [1] "CD_8_T_effector"            "DDR"
##  [3] "APM"                        "Immune_Checkpoint"
##  [5] "CellCycle_Reg"              "Pan_F_TBRs"
##  [7] "Histones"                   "EMT1"
##  [9] "EMT2"                       "EMT3"
## [11] "WNT_target"                 "FGFR3_related"
## [13] "Cell_cycle"                 "Mismatch_Repair"
## [15] "Homologous_recombination"   "Nucleotide_excision_repair"
## [17] "DNA_replication"            "Base_excision_repair"
## [19] "TMEscoreA_CIR"              "TMEscoreB_CIR"
```

```
#Metabolism related signatures
names(signature_metabolism)[1:20]
```

```
##  [1] "Cardiolipin_Metabolism"
##  [2] "Cardiolipin_Biosynthesis"
##  [3] "Cholesterol_Biosynthesis"
##  [4] "Citric_Acid_Cycle"
##  [5] "Cyclooxygenase_Arachidonic_Acid_Metabolism"
##  [6] "Prostaglandin_Biosynthesis"
##  [7] "Purine_Biosynthesis"
##  [8] "Pyrimidine_Biosynthesis"
##  [9] "Dopamine_Biosynthesis"
## [10] "Epinephrine_Biosynthesis"
## [11] "Norepinephrine_Biosynthesis"
## [12] "Fatty_Acid_Degradation"
## [13] "Fatty_Acid_Elongation"
## [14] "Fatty_Acid_Biosynthesis"
## [15] "Folate_One_Carbon_Metabolism"
## [16] "Folate_biosynthesis"
## [17] "Gluconeogenesis"
## [18] "Glycolysis"
## [19] "Glycogen_Biosynthesis"
```

```
## [20] "Glycogen_Degradation"
```

```
#Signatures associated with biomedical basic research: such as m6A and exosomes
names(signature_tumor)
```

```
##  [1] "Nature_metabolism_Hypoxia"
##  [2] "Winter_hypoxia_signature"
##  [3] "Hu_hypoxia_signature"
##  [4] "Molecular_Cancer_m6A"
##  [5] "MT_exosome"
##  [6] "SR_exosome"
##  [7] "Positive_regulation_of_exosomal_secretion"
##  [8] "Negative_regulation_of_exosomal_secretion"
##  [9] "Exosomal_secretion"
## [10] "Exosome_assembly"
## [11] "Extracellular_vesicle_biogenesis"
## [12] "MC_Review_Exosome1"
## [13] "MC_Review_Exosome2"
## [14] "CMLS_Review_Exosome"
## [15] "Ferroptosis"
## [16] "EV_Cell_2020"
```

```
#signature collection including all aforementioned signatures
names(signature_collection)[1:20]
```

```
##  [1] "CD_8_T_effector"            "DDR"
##  [3] "APM"                        "Immune_Checkpoint"
##  [5] "CellCycle_Reg"              "Pan_F_TBRs"
##  [7] "Histones"                   "EMT1"
##  [9] "EMT2"                       "EMT3"
## [11] "WNT_target"                 "FGFR3_related"
## [13] "Cell_cycle"                 "Mismatch_Repair"
## [15] "Homologous_recombination"   "Nucleotide_excision_repair"
## [17] "DNA_replication"            "Base_excision_repair"
## [19] "TMEscoreA_CIR"              "TMEscoreB_CIR"
```

```
#citation of signatures
signature_collection_citation[1:20, ]
```

```
## # A tibble: 20 x 6
```

```
##    Signatures                `Published year` Journal         Title PMID  DOI
##    <chr>                                <dbl> <chr>           <chr> <chr> <chr>
##  1 CD_8_T_effector                       2018 Nature          TGF ~ 2944~ 10.1~
##  2 DDR                                   2018 Nature          TGF ~ 2944~ 10.1~
##  3 APM                                   2018 Nature          TGF ~ 2944~ 10.1~
##  4 Immune_Checkpoint                     2018 Nature          TGF ~ 2944~ 10.1~
##  5 CellCycle_Reg                         2018 Nature          TGF ~ 2944~ 10.1~
##  6 Pan_F_TBRs                            2018 Nature          TGF ~ 2944~ 10.1~
##  7 Histones                              2018 Nature          TGF ~ 2944~ 10.1~
##  8 EMT1                                  2018 Nature          TGF ~ 2944~ 10.1~
##  9 EMT2                                  2018 Nature          TGF ~ 2944~ 10.1~
## 10 EMT3                                  2018 Nature          TGF ~ 2944~ 10.1~
## 11 WNT_target                            2018 Nature          TGF ~ 2944~ 10.1~
## 12 FGFR3_related                         2018 Nature          TGF ~ 2944~ 10.1~
## 13 Cell_cycle                            2018 Nature          TGF ~ 2944~ 10.1~
## 14 Mismatch_Repair                       2018 Nature          TGF ~ 2944~ 10.1~
## 15 Homologous_recombination              2018 Nature          TGF ~ 2944~ 10.1~
## 16 Nucleotide_excision_repair            2018 Nature          TGF ~ 2944~ 10.1~
## 17 DNA_replication                       2018 Nature          TGF ~ 2944~ 10.1~
## 18 Base_excision_repair                  2018 Nature          TGF ~ 2944~ 10.1~
## 19 TMEscoreA_CIR                         2019 Cancer Immunol~ Tumo~ 3084~ 10.1~
## 20 TMEscoreB_CIR                         2019 Cancer Immunol~ Tumo~ 3084~ 10.1~
```

Three methodologies were adopted in the process of signature score evaluation, comprising Single-sample Gene Set Enrichment Analysis (ssGSEA), Principal component analysis (PCA), and Z-score.

4.3.2 Estimated by PCA method

```
sig_tme<-calculate_sig_score(pdata           = NULL,
                             eset            = eset,
                             signature       = signature_collection,
                             method          = "pca",
                             mini_gene_count = 2)

sig_tme <- t(column_to_rownames(sig_tme, var = "ID"))
sig_tme[1:5, 1:3]
```

```
##                    GSM1523727 GSM1523728 GSM1523729
## CD_8_T_effector    -2.5513794  0.7789141 -2.1770675
## DDR                -0.8747614  0.7425162 -1.3272054
## APM                 1.1098368  2.1988688 -0.9516419
## Immune_Checkpoint -2.3701787  0.9455120 -1.4844104
## CellCycle_Reg      0.1063358  0.7583302 -0.3649795
```

4.3.3 Estimated by ssGSEA methodology

This method is suitable for gene sets with a large number of genes, such as those of GO, KEGG, REACTOME gene sets.

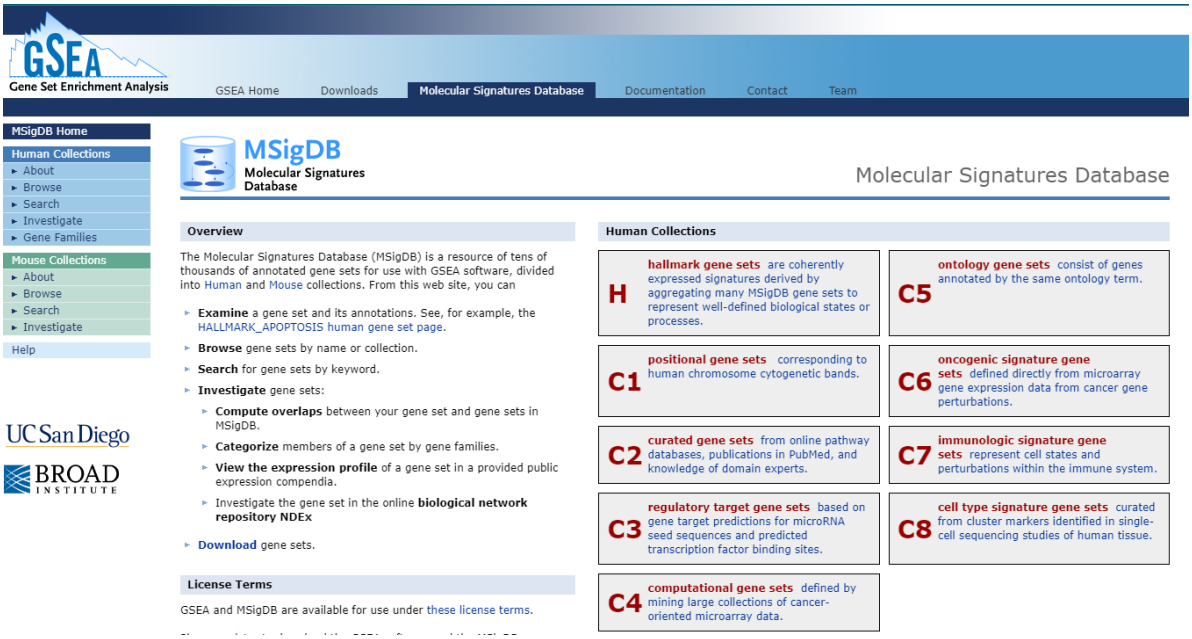


Figure 4.1: Gene sets of MSigDb

```
sig_tme<-calculate_sig_score(pdata           = NULL,
                             eset            = eset,
                             signature       = go_bp,
                             method          = "ssgsea",
                             mini_gene_count = 2)
```

4.3.4 Estimated by zscore function

```
sig_tme<-calculate_sig_score(pdata           = NULL,
                             eset            = eset,
```

```
                              signature       = signature_collection,
                              method          = "zscore",
                              mini_gene_count = 2)
```

4.3.5 Reference

ssgsea: Barbie, D.A. et al (2009). Systematic RNA interference reveals that oncogenic KRAS-driven cancers require TBK1. Nature, 462(5):108-112.

gsva: Hänzelmann, S., Castelo, R. and Guinney, J. (2013). GSVA: Gene set variation analysis for microarray and RNA-Seq data. BMC Bioinformatics, 14(1):7.

zscore: Lee, E. et al (2008). Inferring pathway activity toward precise disease classification. PLoS Comp Biol, 4(11):e1000217.

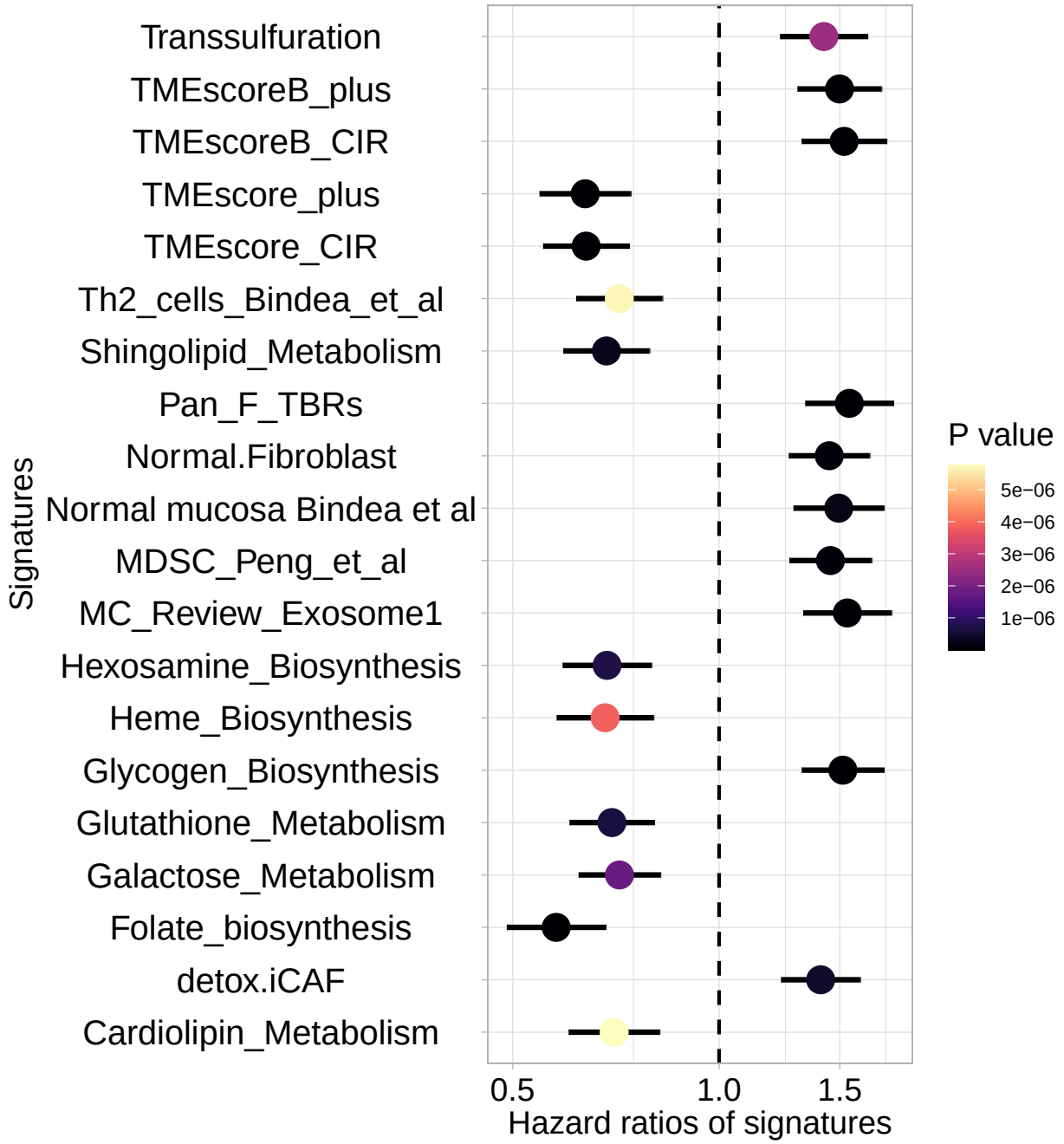
PCA method: Mariathasan S, Turley SJ, Nickles D, et al. TGF attenuates tumour response to PD-L1 blockade by contributing to exclusion of T cells. Nature. 2018 Feb 22;554(7693):544-548.

4.4 Identifying features associated with survival

```
data("pdata_acrg")
input <- combine_pd_eset(eset = sig_tme, pdata = pdata_acrg, scale = T)
res<- batch_surv(pdata    = input,
                 time     = "OS_time",
                 status   = "OS_status",
                 variable = colnames(input)[69:ncol(input)])
head(res)
```

```
## # A tibble: 6 x 5
##   ID                          P    HR CI_low_0.95 CI_up_0.95
##   <chr>                   <dbl> <dbl>       <dbl>      <dbl>
## 1 Folate_biosynthesis   1.00e-10 0.579       0.490      0.683
## 2 TMEscore_CIR          1.32e- 9 0.640       0.554      0.739
## 3 Glycogen_Biosynthesis 3.24e- 9 1.52        1.32       1.74
## 4 Pan_F_TBRs            6.33e- 9 1.55        1.34       1.80
## 5 TMEscoreB_CIR         7.17e- 9 1.52        1.32       1.75
## 6 TMEscore_plus         8.08e- 9 0.638       0.547      0.743
```

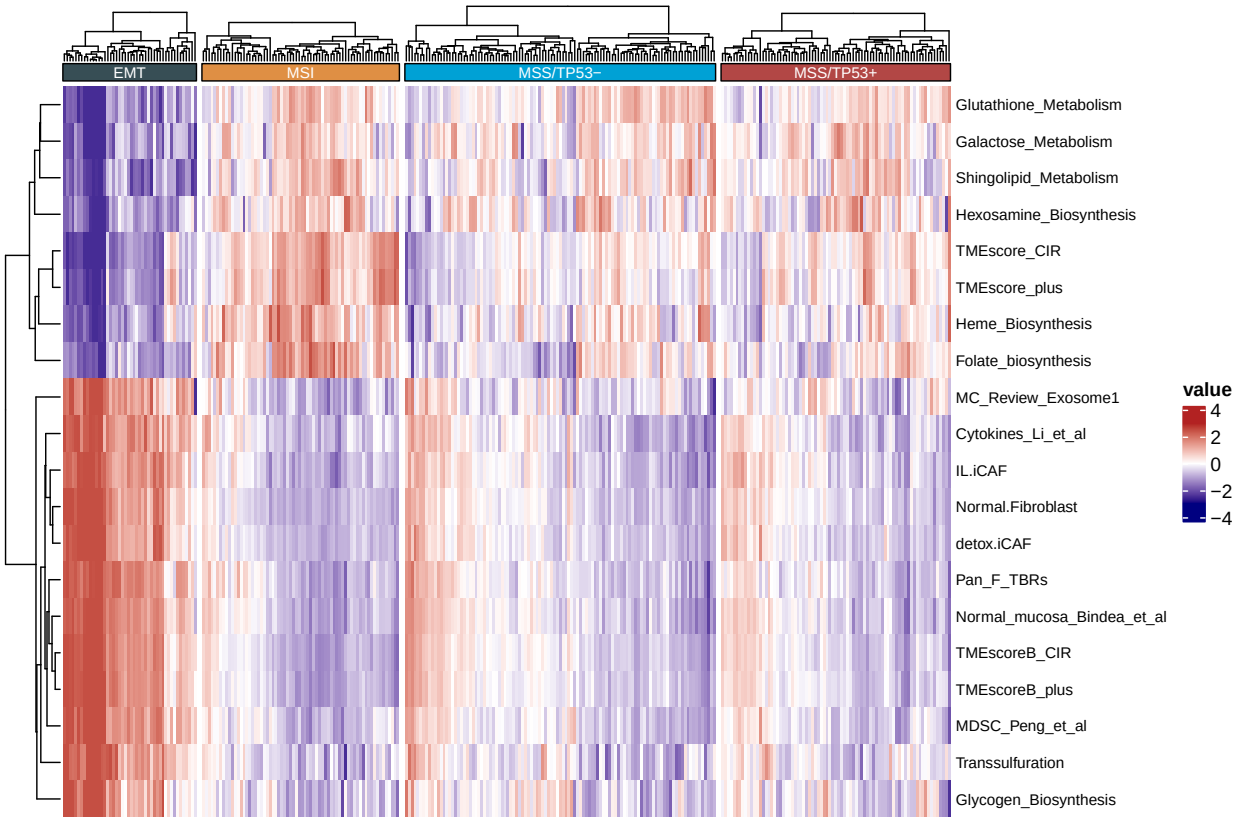
```
res<- res[nchar(res$ID)<=28, ]
p1<- sig_forest(res, signature = "ID", n = 20)
```



4.5 Visulization using heatmap

Signatures IOBR `sig_heatmap`

```
p2 <- sig_heatmap(input         = input,
                  features      = res$ID[1:20],
                  group         = "Subtype",
                  palette_group = "jama",
                  palette       = 6)
```



4.6 Focus on target signatures

```
p1 <- sig_box(data           = input,
              signature      = "Glycogen_Biosynthesis",
              variable       = "Subtype",
              jitter         = FALSE,
              cols           =  NULL,
              palette        = "jama",
              show_pvalue    = TRUE,
              size_of_pvalue = 5,
```

```
              hjust          = 1,
              angle_x_text   = 60,
              size_of_font   = 8)
```

```
## # A tibble: 6 x 8
##   .y.       group1    group2          p    p.adj p.format p.signif method
##   <chr>     <chr>     <chr>       <dbl>    <dbl> <chr>    <chr>    <chr>
## 1 signature EMT       MSI       5.39e-15 3.20e-14 5.4e-15  ****     Wilcoxon
## 2 signature EMT       MSS/TP53- 5.53e-13 2.8 e-12 5.5e-13  ****     Wilcoxon
## 3 signature EMT       MSS/TP53+ 1.90e-12 7.6 e-12 1.9e-12  ****     Wilcoxon
## 4 signature MSI       MSS/TP53- 1.14e- 3 3.4 e- 3 0.0011   **       Wilcoxon
## 5 signature MSI       MSS/TP53+ 7.05e- 3 1.4 e- 2 0.0071   **       Wilcoxon
## 6 signature MSS/TP53- MSS/TP53+ 7.16e- 1 7.2 e- 1 0.7161   ns       Wilcoxon
```

```
p2 <- sig_box(data           = input,
              signature      = "Pan_F_TBRs",
              variable       = "Subtype",
              jitter         = FALSE,
              cols           = NULL,
              palette        = "jama",
              show_pvalue    = TRUE,
              angle_x_text   = 60,
              hjust          = 1,
              size_of_pvalue = 5,
              size_of_font   = 8)
```

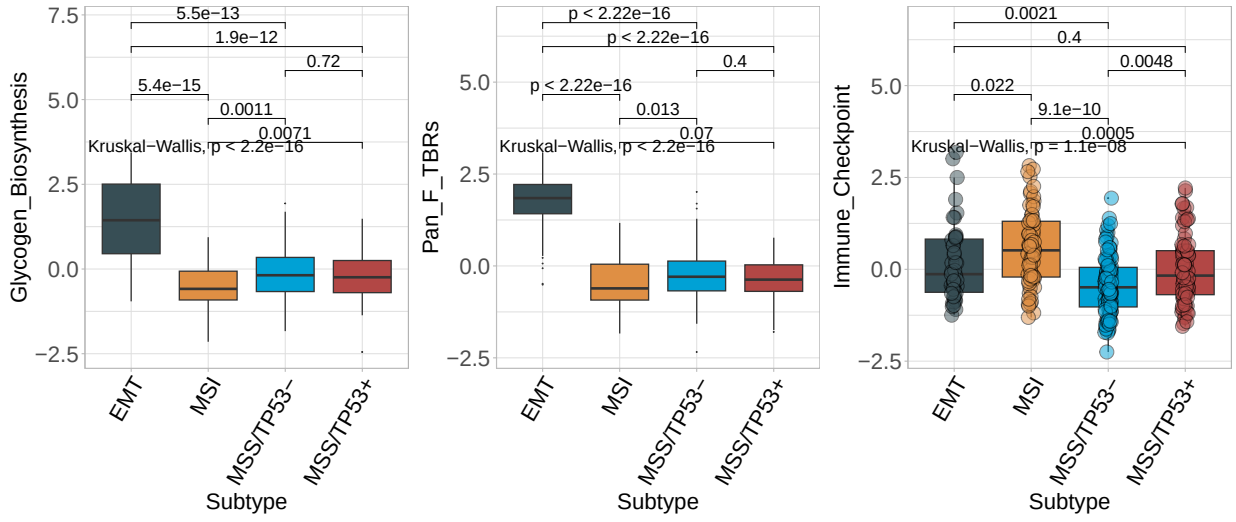
```
## # A tibble: 6 x 8
##   .y.       group1    group2          p    p.adj p.format p.signif method
##   <chr>     <chr>     <chr>       <dbl>    <dbl> <chr>    <chr>    <chr>
## 1 signature EMT       MSI       7.98e-17 3.20e-16 <2e-16   ****     Wilcoxon
## 2 signature EMT       MSS/TP53- 1.70e-17 1   e-16 <2e-16   ****     Wilcoxon
## 3 signature EMT       MSS/TP53+ 2.57e-17 1.3 e-16 <2e-16   ****     Wilcoxon
## 4 signature MSI       MSS/TP53- 1.32e- 2 4   e- 2 0.013    *        Wilcoxon
## 5 signature MSI       MSS/TP53+ 6.99e- 2 1.4 e- 1 0.070    ns       Wilcoxon
## 6 signature MSS/TP53- MSS/TP53+ 4.02e- 1 4   e- 1 0.402    ns       Wilcoxon
```

```
p3 <- sig_box(data           = input,
              signature      = "Immune_Checkpoint",
              variable       = "Subtype",
```

```
              jitter          = TRUE,
              cols            = NULL,
              palette         = "jama",
              show_pvalue     = TRUE,
              angle_x_text    = 60,
              hjust           = 1,
              size_of_pvalue = 5,
              size_of_font    = 8)
```

```
## # A tibble: 6 x 8
##   .y.       group1    group2           p        p.adj p.format p.signif method
##   <chr>     <chr>     <chr>        <dbl>        <dbl> <chr>    <chr>    <chr>
## 1 signature EMT       MSI       2.20e- 2 0.044        0.0220   *        Wilcoxon
## 2 signature EMT       MSS/TP53- 2.11e- 3 0.0085       0.0021   **       Wilcoxon
## 3 signature EMT       MSS/TP53+ 4.03e- 1 0.4          0.4026   ns       Wilcoxon
## 4 signature MSI       MSS/TP53- 9.13e-10 0.0000000055 9.1e-10  ****     Wilcoxon
## 5 signature MSI       MSS/TP53+ 5.03e- 4 0.0025       0.0005   ***      Wilcoxon
## 6 signature MSS/TP53- MSS/TP53+ 4.82e- 3 0.014        0.0048   **       Wilcoxon
```

```
p1|p2|p3
```



4.7 Survival analysis

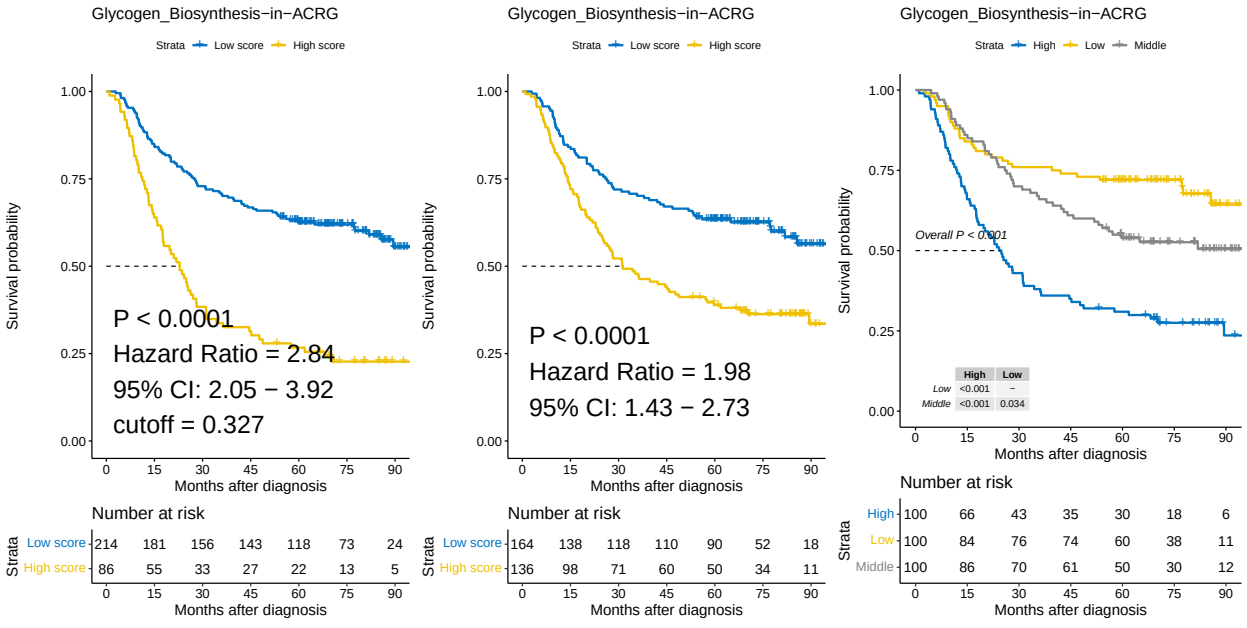
Signature

```
res <-       sig_surv_plot(input_pdata       = input,
                           signature         = "Glycogen_Biosynthesis",
                           cols              = NULL,
                           palette           = "jco",
                           project           = "ACRG",
                           time              = "OS_time",
                           status            = "OS_status",
                           time_type         = "month",
                           save_path         = "result")
```

```
##           ID   time status Glycogen_Biosynthesis group3 group2 bestcutoff
## 1 GSM1523727  88.73      0             -0.3612213 Middle    Low        Low
## 2 GSM1523728  88.23      0             -0.6926726    Low    Low        Low
## 3 GSM1523729  88.23      0             -0.9388531    Low    Low        Low
## 4 GSM1523744 105.70      0             -1.1825136    Low    Low        Low
## 5 GSM1523745 105.53      0             -0.3034304 Middle    Low        Low
## 6 GSM1523746  25.50      1              0.7517934   High   High       High
```

```
## [1] ">>>>>>>>>>"
```

```
res$plots
```



Signature ROC

```
p1<- roc_time(input      = input,
              vars       = "Glycogen_Biosynthesis",
              time       = "OS_time",
              status     = "OS_status",
              time_point = c(12, 24, 36),
              time_type  = "month",
              palette    = "jama",
              cols       = "normal",
              seed       = 1234,
              show_col   = FALSE,
              path       = "result",
              main       = "OS",
              index      = 1,
              fig.type   = "pdf",
              width      = 5,
              height     = 5.2)
```

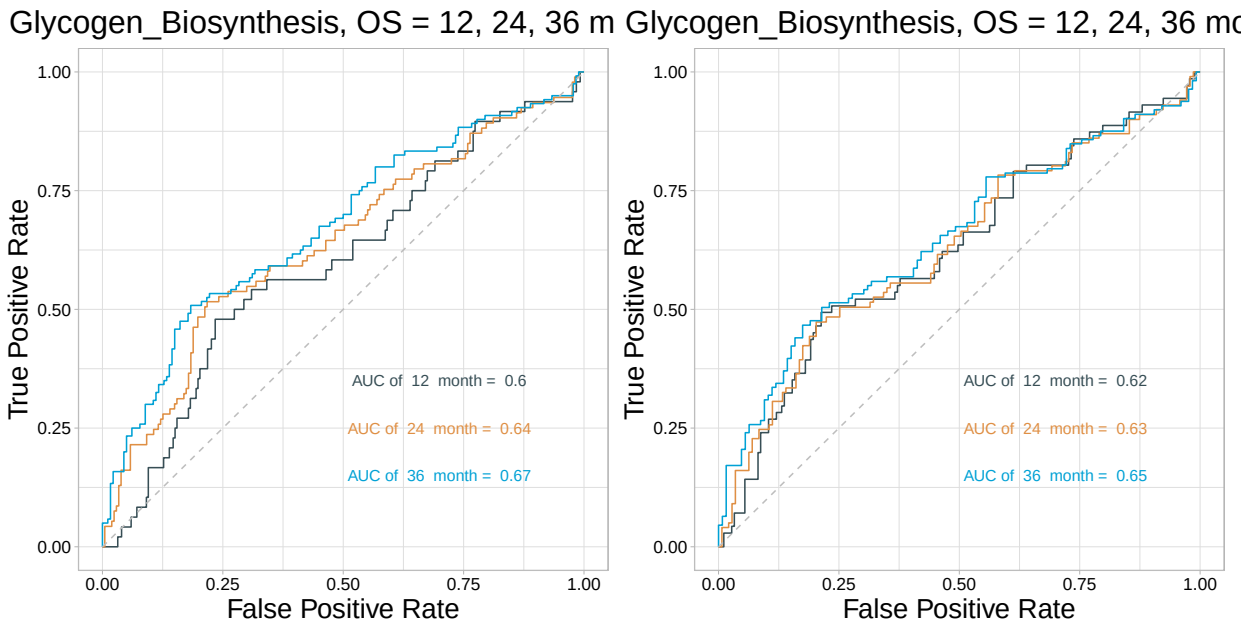
```
## [1] ">>>-- Range of Time: "
## [1]   1.0 105.7
```

```
p2<- roc_time(input      = input,
              vars       = "Glycogen_Biosynthesis",
              time       = "RFS_time",
              status     = "RFS_status",
              time_point = c(12, 24, 36),
              time_type  = "month",
              palette    = "jama",
              cols       = "normal",
              seed       = 1234,
              show_col   = FALSE,
              path       = "result",
              main       = "OS",
              index      = 1,
              fig.type   = "pdf",
              width      = 5,
              height     = 5.2)
```

```
## [1] ">>>-- Range of Time: "
```

```
## [1]   0.10 100.87
```

```
p1|p2
```



4.8 Batch correlation analysis

signature signatures

```
res <- batch_cor(data = input, target = "Glycogen_Biosynthesis", feature = colnames(inpu
head(res)
```

```
## # A tibble: 6 x 6
##   sig_names                          p.value statistic    p.adj log10pvalue stars
##   <chr>                                <dbl>     <dbl>    <dbl>       <dbl> <fct>
## 1 TMEscoreB_CIR                     8.89e-42     0.678 2.27e-39        41.1 ****
## 2 Glycine__Serine_and_Threonine_M~ 7.49e-40    -0.666 9.54e-38        39.1 ****
## 3 Ether_Lipid_Metabolism           3.84e-39     0.662 3.27e-37        38.4 ****
## 4 MDSC_Peng_et_al                  1.13e-38     0.659 7.21e-37        37.9 ****
## 5 Glycerophospholipid_Metabolism   8.72e-38    -0.653 4.44e-36        37.1 ****
## 6 TIP_Release_of_cancer_cell_anti~ 2.32e-37    -0.650 9.86e-36        36.6 ****
```

```
p1<- get_cor(eset = sig_tme, pdata = pdata_acrg, var1 = "Glycogen_Biosynthesis", var2 =
```

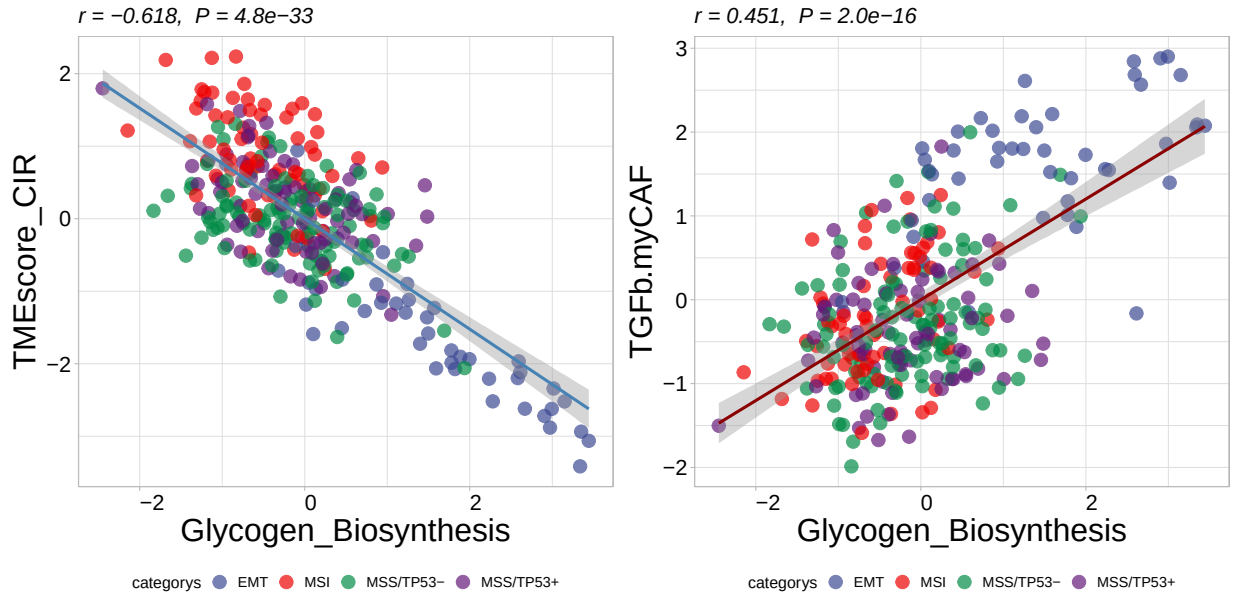
```
##
##  Spearman's rank correlation rho
```

```
##
## data:  data[, var1] and data[, var2]
## S = 7282858, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## -0.6184309
##
## [1] ">>>--- The exact p value is: 4.78971420439895e-33"
##       EMT       MSI MSS/TP53- MSS/TP53+
##        46        68       107        79
```

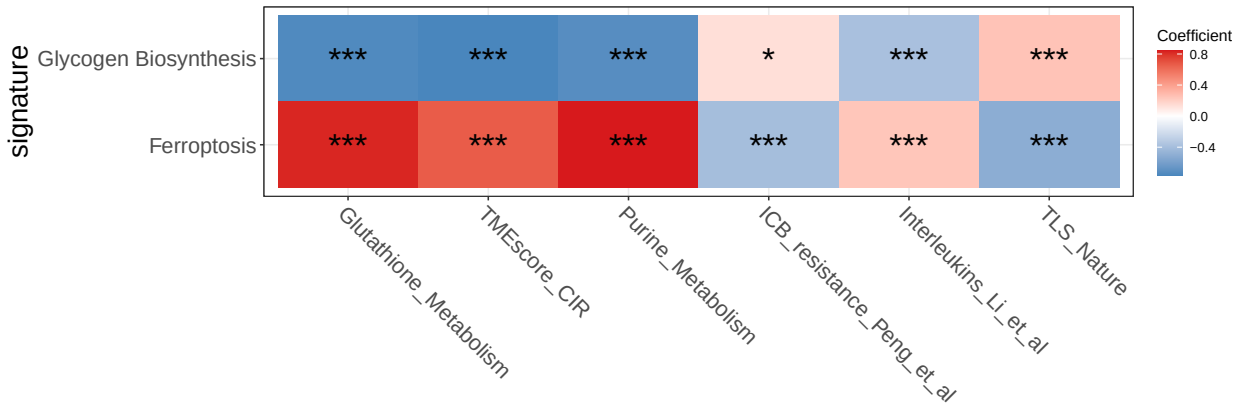
```
p2<- get_cor(eset = sig_tme, pdata = pdata_acrg, var1 = "Glycogen_Biosynthesis", var2 =
```

```
##
##  Spearman's rank correlation rho
##
## data:  data[, var1] and data[, var2]
## S = 2471758, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.4507143
##
## [1] ">>>--- The exact p value is: 2.04505761057615e-16"
##       EMT       MSI MSS/TP53- MSS/TP53+
##        46        68       107        79
```

```
p1|p2
```




```r
feas1 <- c("Glycogen_Biosynthesis", "Ferroptosis")
feas2 <- c("Glutathione_Metabolism", "TMEscore_CIR", "Purine_Metabolism", "ICB_resistanc
p <- get_cor_matrix(data          = input,
                    feas1         = feas2,
                    feas2         = feas1,
                    method        = "pearson",
                    font.size.star = 8,
                    font.size     = 15,
                    fill_by_cor   = FALSE,
                    round.num     = 1)
```

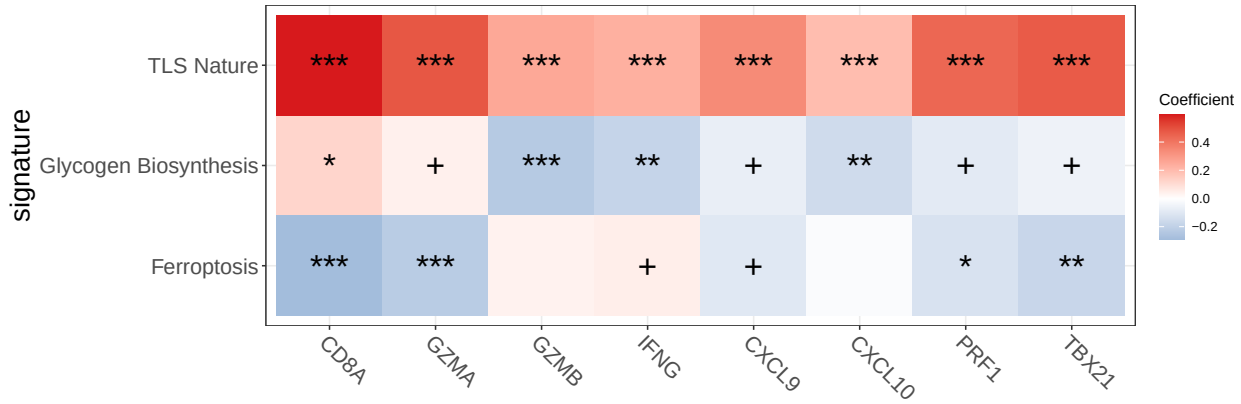


4.9 Visulization of correlations

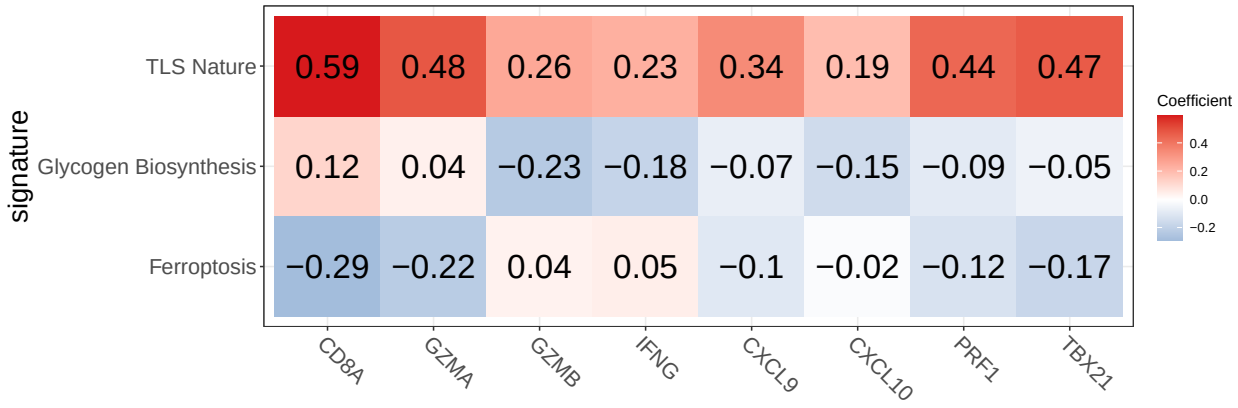
```
input2 <- combine_pd_eset(eset = eset, pdata =  input[, c("ID", "Glycogen_Biosynthesis",
feas1 <- c("Glycogen_Biosynthesis","TLS_Nature", "Ferroptosis")
feas2 <- signature_collection$CD_8_T_effector
feas2
```

```
## [1] "CD8A"   "GZMA"   "GZMB"   "IFNG"   "CXCL9"  "CXCL10" "PRF1"   "TBX21"
```

```
p <- get_cor_matrix(data           = input2,
                    feas1          = feas2,
                    feas2          = feas1,
                    method         = "pearson",
                    scale          = T,
                    font.size.star = 8,
                    font.size      = 15,
                    fill_by_cor    = FALSE,
                    round.num      = 1)
```




```
p <- get_cor_matrix(data           = input2,
                    feas1          = feas2,
                    feas2          = feas1,
                    method         = "pearson",
                    scale          = T,
                    font.size.star = 8,
                    font.size      = 15,
                    fill_by_cor    = TRUE,
                    round.num      = 2)
```


signature
TLS Nature
Glycogen Biosynthesis
Ferroptosis
0.59
0.48
0.26
0.23
0.34
0.19
0.44
0.47
0.12
0.04
−0.23
−0.18
−0.07
−0.15
−0.09
−0.05
−0.29
−0.22
0.04
0.05
−0.1
−0.02
−0.12
−0.17
CD8A
GZMA
GZMB
IFNG
CXCL9
CXCL10
PRF1
TBX21
Coefficient
0.4
0.2
0.0
−0.2

Chapter 5

TME deconvolution

5.1 Loading packages

Load the IOBR package in your R session after the installation is complete:

```
library(IOBR)
library(survminer)
library(tidyverse)
```

5.2 Downloading data for example

Obtaining data set from GEO Gastric cancer: GSE62254 using `GEOquery` R package.

```
if (!requireNamespace("GEOquery", quietly = TRUE))  BiocManager::install("GEOquery")
library("GEOquery")
# NOTE: This process may take a few minutes which depends on the internet connection s
eset_geo<-getGEO(GEO     = "GSE62254", getGPL  = F, destdir = "./")
eset    <-eset_geo[[1]]
eset    <-exprs(eset)
eset[1:5,1:5]
```

```
##           GSM1523727 GSM1523728 GSM1523729 GSM1523744 GSM1523745
## 1007_s_at  3.2176645  3.0624323  3.0279131   2.921683  2.8456013
## 1053_at    2.4050109  2.4394879  2.2442708   2.345916  2.4328582
## 117_at     1.4933412  1.8067380  1.5959665   1.839822  1.8326058
## 121_at     2.1965561  2.2812181  2.1865556   2.258599  2.1874363
```

```
## 1255_g_at  0.8698382  0.9502466  0.8125414   1.012860  0.9441993
```

Annotation of genes in the expression matrix and removal of duplicate genes.

```
library(IOBR)

# Load the annotation file `anno_hug133plus2` in IOBR.
head(anno_hug133plus2)
```

```
## # A tibble: 6 x 2
##   probe_id  symbol
##   <fct>     <fct>
## 1 1007_s_at MIR4640
## 2 1053_at   RFC2
## 3 117_at    HSPA6
## 4 121_at    PAX8
## 5 1255_g_at GUCA1A
## 6 1294_at   MIR5193
```

```
# Conduct gene annotation using `anno_hug133plus2` file; If identical gene symbols exi

eset<-anno_eset(eset       = eset,
                annotation = anno_hug133plus2,
                symbol     = "symbol",
                probe      = "probe_id",
                method     = "mean")
eset[1:5, 1:3]
```

```
##              GSM1523727 GSM1523728 GSM1523729
## SH3KBP1        4.327974   4.316195   4.351425
## RPL41          4.246149   4.246808   4.257940
## EEF1A1         4.293762   4.291038   4.262199
## COX2           4.250288   4.283714   4.270508
## LOC101928826   4.219303   4.219670   4.213252
```

5.3 Available Methods to Decode TME Contexture

```
tme_deconvolution_methods
```

```
##         MCPcounter               EPIC              xCell          CIBERSORT
```

```
##       "mcpcounter"             "epic"            "xcell"        "cibersort"
## CIBERSORT Absolute               IPS          ESTIMATE               SVR
##    "cibersort_abs"             "ips"         "estimate"             "svr"
##               lsei             TIMER          quanTIseq
##             "lsei"           "timer"        "quantiseq"
```

```
# Return available parameter options of deconvolution methods
```

The input data is a matrix subseted from ESET of ACRG cohort, with genes in rows and samples in columns. The row name must be HGNC symbols and the column name must be sample names.

```
eset_acrg <- eset[, 1:50]
eset_acrg[1:5, 1:3]
```

```
##              GSM1523727 GSM1523728 GSM1523729
## SH3KBP1        4.327974   4.316195   4.351425
## RPL41          4.246149   4.246808   4.257940
## EEF1A1         4.293762   4.291038   4.262199
## COX2           4.250288   4.283714   4.270508
## LOC101928826   4.219303   4.219670   4.213252
```

Check detail parameters of the function

```
# help(deconvo_tme)
```

5.4 Method 1: CIBERSORT

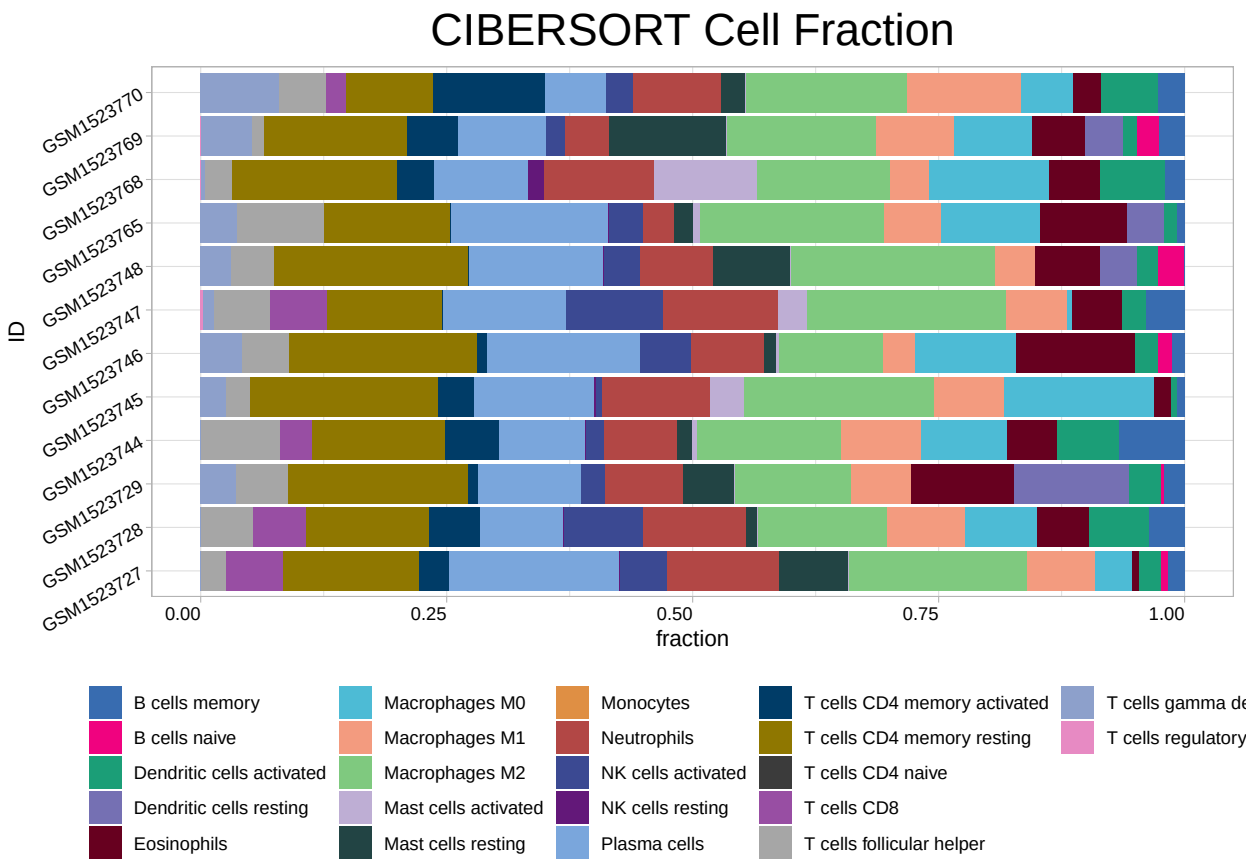
```
cibersort<-deconvo_tme(eset = eset_acrg, method = "cibersort", arrays = TRUE, perm = 100
```

```
##
## >>> Running CIBERSORT
```

```
# head(cibersort)
res<-cell_bar_plot(input = cibersort[1:12,], title = "CIBERSORT Cell Fraction")
```

```
## There are seven categories you can choose: box, continue2, continue, random, heatmap,
```

```
## >>>>=== Palette option for random: 1: palette1; 2: palette2; 3: palette3;  4: palette
```



5.5 Method 2: EPIC

```
# help(deconvo_epic)
epic<-deconvo_tme(eset = eset_acrg, method = "epic", arrays = TRUE)
```

```
##
## >>> Running EPIC

## Warning in IOBR::EPIC(bulk = eset, reference = ref, mRNA_cell = NULL, scaleExprs = TR
## GSM1523744; GSM1523746; GSM1523781; GSM1523786
##  - check fit.gof for the convergeCode and convergeMessage

## Warning in IOBR::EPIC(bulk = eset, reference = ref, mRNA_cell = NULL, scaleExprs
## = TRUE): mRNA_cell value unknown for some cell types: CAFs, Endothelial - using
## the default value of 0.4 for these but this might bias the true cell proportions
## from all cell types.
```

```
head(epic)
```

```
## # A tibble: 6 x 9
##   ID      Bcells_EPIC CAFs_EPIC CD4_Tcells_EPIC CD8_Tcells_EPIC Endothelial_EPIC
##   <chr>         <dbl>     <dbl>           <dbl>           <dbl>            <dbl>
## 1 GSM152~      0.0292   0.00888           0.145          0.0756           0.0876
## 2 GSM152~      0.0293   0.0109            0.159          0.0745           0.0954
## 3 GSM152~      0.0308   0.0106            0.149          0.0732           0.0941
## 4 GSM152~      0.0273   0.0108            0.145          0.0704           0.0860
## 5 GSM152~      0.0280   0.0111            0.151          0.0707           0.0928
## 6 GSM152~      0.0320   0.00958           0.148          0.0716           0.0907
## # i 3 more variables: Macrophages_EPIC <dbl>, NKcells_EPIC <dbl>,
## #   otherCells_EPIC <dbl>
```

5.6 Method 3: MCPcounter

```
mcp<-deconvo_tme(eset = eset_acrg, method = "mcpcounter")
```

```
##
## >>> Running MCP-counter
```

```
head(mcp)
```

```
## # A tibble: 6 x 11
##   ID         T_cells_MCPcounter CD8_T_cells_MCPcounter Cytotoxic_lymphocytes_M~1
##   <chr>                   <dbl>                  <dbl>                      <dbl>
## 1 GSM1523727               1.47                   1.11                       1.33
## 2 GSM1523728               1.53                   1.05                       1.60
## 3 GSM1523729               1.47                   1.07                       1.37
## 4 GSM1523744               1.46                   1.02                       1.44
## 5 GSM1523745               1.51                   1.10                       1.49
## 6 GSM1523746               1.51                   0.992                      1.40
## # i abbreviated name: 1: Cytotoxic_lymphocytes_MCPcounter
## # i 7 more variables: B_lineage_MCPcounter <dbl>, NK_cells_MCPcounter <dbl>,
## #   Monocytic_lineage_MCPcounter <dbl>,
## #   Myeloid_dendritic_cells_MCPcounter <dbl>, Neutrophils_MCPcounter <dbl>,
## #   Endothelial_cells_MCPcounter <dbl>, Fibroblasts_MCPcounter <dbl>
```

5.7 Method 4: xCELL

```
xcell<-deconvo_tme(eset = eset_acrg, method = "xcell", arrays = TRUE)
```

```
head(xcell)
```

```
## # A tibble: 6 x 68
##   ID         aDC_xCell Adipocytes_xCell Astrocytes_xCell `B-cells_xCell`
##   <chr>          <dbl>            <dbl>            <dbl>           <dbl>
## 1 GSM1523727  4.78e-19          0.0250          0                0
## 2 GSM1523728  9.41e- 2          0.00433         7.70e- 3         0
## 3 GSM1523729  1.02e- 1          0.0789          2.04e- 2         0
## 4 GSM1523744  7.88e- 2          0.0538          4.82e-18         0.0126
## 5 GSM1523745  9.02e- 2          0.0136          1.93e- 2         0
## 6 GSM1523746  3.40e- 2          0.0331          9.22e- 2         0
## # i 63 more variables: Basophils_xCell <dbl>,
## #   `CD4+_memory_T-cells_xCell` <dbl>, `CD4+_naive_T-cells_xCell` <dbl>,
## #   `CD4+_T-cells_xCell` <dbl>, `CD4+_Tcm_xCell` <dbl>, `CD4+_Tem_xCell` <dbl>,
## #   `CD8+_naive_T-cells_xCell` <dbl>, `CD8+_T-cells_xCell` <dbl>,
## #   `CD8+_Tcm_xCell` <dbl>, `CD8+_Tem_xCell` <dbl>, cDC_xCell <dbl>,
## #   Chondrocytes_xCell <dbl>, `Class-switched_memory_B-cells_xCell` <dbl>,
## #   CLP_xCell <dbl>, CMP_xCell <dbl>, DC_xCell <dbl>, ...
```

5.8 Method 5: ESTIMATE

```
estimate<-deconvo_tme(eset = eset_acrg, method = "estimate")
```

```
## [1] "Merged dataset includes 9940 genes (472 mismatched)."
## [1] "1 gene set: StromalSignature  overlap= 136"
## [1] "2 gene set: ImmuneSignature  overlap= 138"
```

```
head(estimate)
```

```
## # A tibble: 6 x 5
##   ID         StromalScore_estimate ImmuneScore_estimate ESTIMATEScore_estimate
##   <chr>                      <dbl>                <dbl>                   <dbl>
## 1 GSM1523727                -1250.                 268.                   -982.
## 2 GSM1523728                  197.                1334.                   1531.
## 3 GSM1523729                 -111.                 822.                    711.
```

```
## 4 GSM1523744                -119.                662.                 544.
## 5 GSM1523745                 324.               1015.                1339.
## 6 GSM1523746                -594.                621.                  27.0
## # i 1 more variable: TumorPurity_estimate <dbl>
```

5.9 Method 6: TIMER

```
timer<-deconvo_tme(eset = eset_acrg, method = "timer", group_list = rep("stad",dim(eset_
```

```
## [1] "Outlier genes: AGR2 B2M COL1A2 COL3A1 COX2 CYAT1 EEF1A1 EIF1 FTH1 GKN1 HUWE1 IGK
```

```
head(timer)
```

```
## # A tibble: 6 x 7
##   ID         B_cell_TIMER T_cell_CD4_TIMER T_cell_CD8_TIMER Neutrophil_TIMER
##   <chr>             <dbl>            <dbl>            <dbl>             <dbl>
## 1 GSM1523727        0.104            0.128            0.183             0.108
## 2 GSM1523728        0.103            0.130            0.192             0.118
## 3 GSM1523729        0.106            0.130            0.190             0.110
## 4 GSM1523744        0.101            0.126            0.187             0.111
## 5 GSM1523745        0.104            0.127            0.191             0.116
## 6 GSM1523746        0.105            0.129            0.192             0.111
## # i 2 more variables: Macrophage_TIMER <dbl>, DC_TIMER <dbl>
```

5.10 Method 7: quanTIseq

```
quantiseq<-deconvo_tme(eset = eset_acrg, tumor = TRUE, arrays = TRUE, scale_mrna = TRUE,
```

```
##
## Running quanTIseq deconvolution module

## Gene expression normalization and re-annotation (arrays: TRUE)

## Removing 17 genes with high expression in tumors

## Signature genes found in data set: 152/153 (99.35%)

## Mixture deconvolution (method: lsei)

## Deconvolution sucessful!
```

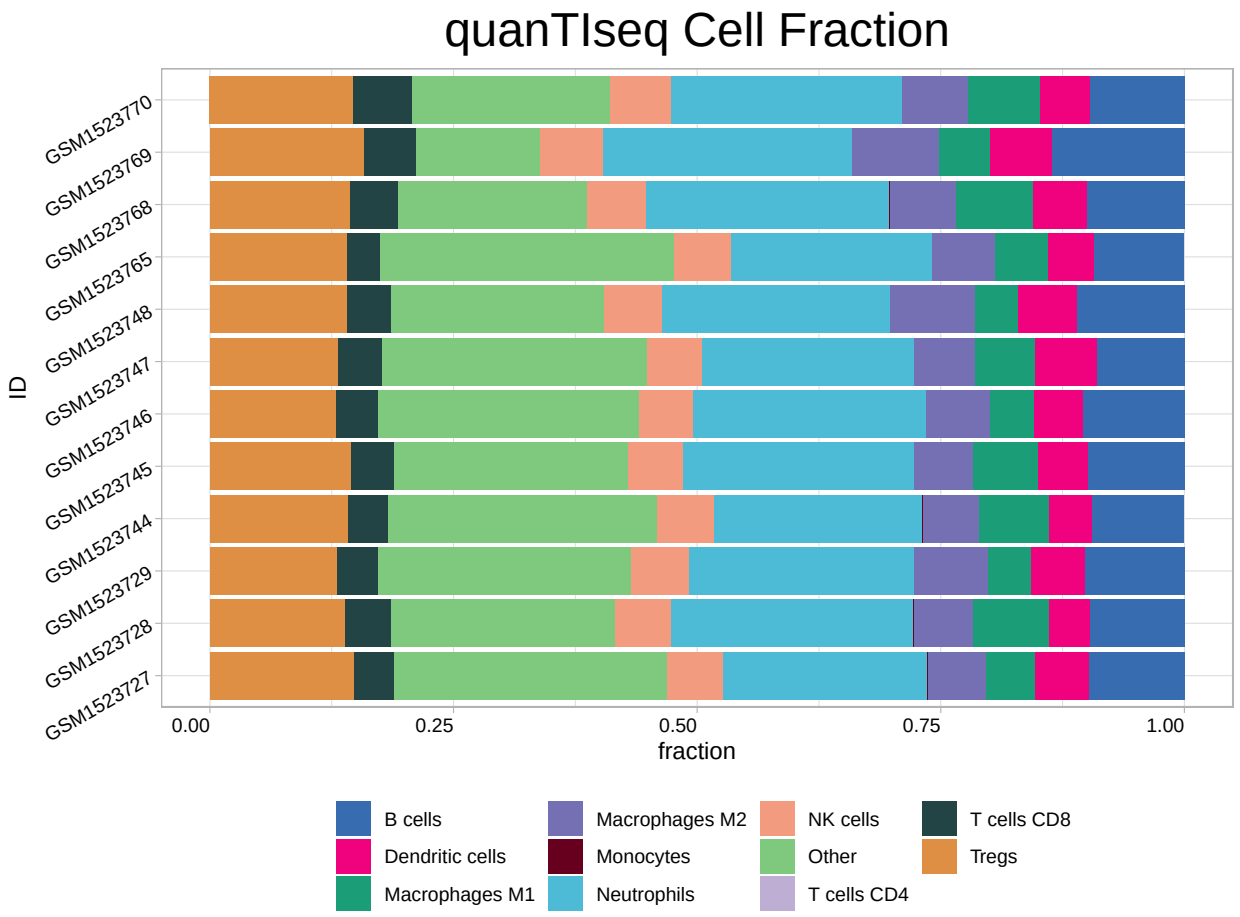
```
head(quantiseq)
```

```
## # A tibble: 6 x 12
##   ID         B_cells_quantiseq Macrophages_M1_quantiseq Macrophages_M2_quantiseq
##   <chr>                  <dbl>                    <dbl>                    <dbl>
## 1 GSM1523727            0.0983                   0.0510                   0.0598
## 2 GSM1523728            0.0967                   0.0795                   0.0607
## 3 GSM1523729            0.102                    0.0450                   0.0758
## 4 GSM1523744            0.0954                   0.0725                   0.0579
## 5 GSM1523745            0.0991                   0.0669                   0.0613
## 6 GSM1523746            0.105                    0.0453                   0.0662
## # i 8 more variables: Monocytes_quantiseq <dbl>, Neutrophils_quantiseq <dbl>,
## #   NK_cells_quantiseq <dbl>, T_cells_CD4_quantiseq <dbl>,
## #   T_cells_CD8_quantiseq <dbl>, Tregs_quantiseq <dbl>,
## #   Dendritic_cells_quantiseq <dbl>, Other_quantiseq <dbl>
```

```
res<-cell_bar_plot(input = quantiseq[1:12, ], title = "quanTIseq Cell Fraction")
```

```
## There are seven categories you can choose: box, continue2, continue, random, heatmap,
```

```
## >>>>=== Palette option for random: 1: palette1; 2: palette2; 3: palette3;  4: palette
```



5.11 Method 8: IPS

```
ips<-deconvo_tme(eset = eset_acrg, method = "ips", plot= FALSE)
head(ips)
```

```
## # A tibble: 6 x 7
##   ID         MHC_IPS EC_IPS SC_IPS  CP_IPS AZ_IPS IPS_IPS
##   <chr>        <dbl>  <dbl>  <dbl>   <dbl>  <dbl>   <dbl>
## 1 GSM1523727    2.25  0.404 -0.192  0.220    2.68       9
## 2 GSM1523728    2.37  0.608 -0.578 -0.234    2.17       7
## 3 GSM1523729    2.10  0.480 -0.322  0.0993   2.36       8
## 4 GSM1523744    2.12  0.535 -0.333  0.0132   2.34       8
## 5 GSM1523745    1.91  0.559 -0.479  0.0880   2.08       7
## 6 GSM1523746    1.94  0.458 -0.346  0.261    2.31       8
```

5.12 Combination of above deconvolution results

```
tme_combine<-cibersort %>%
  inner_join(.,mcp,by       = "ID") %>%
  inner_join(.,xcell,by     = "ID") %>%
  inner_join(.,epic,by      = "ID") %>%
  inner_join(.,estimate,by  = "ID") %>%
  inner_join(.,timer,by     = "ID") %>%
  inner_join(.,quantiseq,by = "ID") %>%
  inner_join(.,ips,by       = "ID")
dim(tme_combine)
```

```
## [1]  50 138
```

If you use this package in your work, please cite both our package and the method(s) you are using.

Citation and licenses of these deconvolution methods

CIBERSORT; free for non-commerical use only; Newman, A. M., Liu, C. L., Green, M. R., Gentles, A. J., Feng, W., Xu, Y., … Alizadeh, A. A. (2015). Robust enumeration of cell subsets from tissue expression profiles. Nature Methods, 12(5), 453–457. https://doi.org/10.1038/nmeth.3337;

ESTIMATE; free (GPL2.0); Vegesna R, Kim H, Torres-Garcia W, …, Verhaak R. (2013). Inferring tumour purity and stromal and immune cell admixture from expression data. Nature Communications 4, 2612. http://doi.org/10.1038/ncomms3612;

quanTIseq; free (BSD); Finotello, F., Mayer, C., Plattner, C., Laschober, G., Rieder, D., Hackl, H., …, Sopper, S. (2019). Molecular and pharmacological modulators of the tumor immune contexture revealed by deconvolution of RNA-seq data. Genome medicine, 11(1), 34. https://doi.org/10.1186/s13073-019-0638-6;

TIMER; free (GPL 2.0); Li, B., Severson, E., Pignon, J.-C., Zhao, H., Li, T., Novak, J., … Liu, X. S. (2016). Comprehensive analyses of tumor immunity: implications for cancer immunotherapy. Genome Biology, 17(1), 174. https://doi.org/10.1186/s13059-016-1028-7;

IPS; free (BSD); P. Charoentong et al., Pan-cancer Immunogenomic Analyses Reveal Genotype-Immunophenotype Relationships and Predictors of Response to Checkpoint Blockade. Cell Reports 18, 248-262 (2017). https://doi.org/10.1016/j.celrep.2016.12.019;

MCPCounter; free (GPL 3.0); Becht, E., Giraldo, N. A., Lacroix, L., Buttard, B., Elarouci,

N., Petitprez, F., … de Reyniès, A. (2016). Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression. Genome Biology, 17(1), 218. https://doi.org/10.1186/s13059-016-1070-5;

xCell; free (GPL 3.0); Aran, D., Hu, Z., & Butte, A. J. (2017). xCell: digitally portraying the tissue cellular heterogeneity landscape. Genome Biology, 18(1), 220. https://doi.org/10.1186/s13059-017-1349-1;

EPIC; free for non-commercial use only (Academic License); Racle, J., de Jonge, K., Baumgaertner, P., Speiser, D. E., & Gfeller, D. (2017). Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data. ELife, 6, e26476. https://doi.org/10.7554/eLife.26476;

GSVA free (GPL (>= 2)) Hänzelmann S, Castelo R, Guinney J (2013). "GSVA: gene set variation analysis for microarray and RNA-Seq data." BMC Bioinformatics, 14, 7. doi: 10.1186/1471-2105-14-7, http://www.biomedcentral.com/1471-2105/14/7

Chapter 6

References

If IOBR R package is utilized in your published research, please cite:

Zeng D, Ye Z, Shen R, Yu G, Wu J, Xiong Y,..., Liao W (2021) **IOBR**: Multi-Omics Immuno-Oncology Biological Research to Decode Tumor Microenvironment and Signatures. *Frontiers in Immunology*. 12:687975. doi: 10.3389/fimmu.2021.687975

6.1 TME deconvolution

Please cite the following papers appropriately for TME deconvolution algorithm if used:

CIBERSORT: Newman, A. M., Liu, C. L., Green, M. R., Gentles, A. J., Feng, W., Xu, Y., … Alizadeh, A. A. (2015). Robust enumeration of cell subsets from tissue expression profiles. Nature Methods, 12(5), 453–457. https://doi.org/10.1038/nmeth.3337

ESTIMATE: Vegesna R, Kim H, Torres-Garcia W, ..., Verhaak R.*(2013). Inferring tumour purity and stromal and immune cell admixture from expression data. Nature Communications 4, 2612. http://doi.org/10.1038/ncomms3612

quanTIseq: Finotello, F., Mayer, C., Plattner, C., Laschober, G., Rieder, D., Hackl, H., ..., Sopper, S.* (2019). Molecular and pharmacological modulators of the tumor immune contexture revealed by deconvolution of RNA-seq data. Genome medicine, 11(1), 34. https://doi.org/10.1186/s13073-019-0638-6

TIMER: Li, B., Severson, E., Pignon, J.-C., Zhao, H., Li, T., Novak, J., … Liu, X. S.* (2016). Comprehensive analyses of tumor immunity: implications for cancer immunotherapy. Genome Biology, 17(1), 174.

IPS: P. Charoentong et al.*, Pan-cancer Immunogenomic Analyses Reveal Genotype-Immunophenotype Relationships and Predictors of Response to Checkpoint Blockade. Cell Reports 18, 248-262 (2017). https://doi.org/10.1016/j.celrep.2016.12.019

MCPCounter: Becht, E., Giraldo, N. A., Lacroix, L., Buttard, B., Elarouci, N., Petitprez, F., … de Reyniès, A*. (2016). Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression. Genome Biology, 17(1), 218. https://doi.org/10.1186/s13059-016-1070-5

xCell: Aran, D., Hu, Z., & Butte, A. J.* (2017). xCell: digitally portraying the tissue cellular heterogeneity landscape. Genome Biology, 18(1), 220. https://doi.org/10.1186/s13059-017-1349-1

EPIC: Racle, J., de Jonge, K., Baumgaertner, P., Speiser, D. E., & Gfeller, D*. (2017). Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data. ELife, 6, e26476. https://doi.org/10.7554/eLife.26476

6.2 TME Signatures

For signature score estimation, please cite corresponding literature below:

ssgsea: Barbie, D.A. et al (2009). Systematic RNA interference reveals that oncogenic KRAS-driven cancers require TBK1. Nature, 462(5):108-112.

gsva: Hänzelmann, S., Castelo, R. and Guinney, J. (2013). GSVA: Gene set variation analysis for microarray and RNA-Seq data. BMC Bioinformatics, 14(1):7.

zscore: Lee, E. et al (2008). Inferring pathway activity toward precise disease classification. PLoS Comp Biol, 4(11):e1000217.

6.3 Data sets

For the datasets enrolled in IOBR, please cite the data sources:

UCSCXena: Wang et al.,et al (2019). The UCSCXenaTools R package: a toolkit for accessing genomics data from UCSC Xena platform, from cancer multi-omics to single-cell RNA-seq. Journal of Open Source Software, 4(40), 1627

TLSscore: Helmink BA, Reddy SM, Gao J, et al. B cells and tertiary lymphoid structures promote immunotherapy response. Nature. 2020 Jan;577(7791):549-555.

IMvigor210 immuntherapy cohort: Mariathasan S, Turley SJ, Nickles D, et al. TGF attenuates tumour response to PD-L1 blockade by contributing to exclusion of T cells. Nature. 2018 Feb 22;554(7693):544-548. **HCP5**: Kulski, J.K. Long Noncoding RNA HCP5, a Hybrid HLA Class I Endogenous Retroviral Gene: Structure, Expression, and Disease Associations. Cells 2019, 8, 480.

HCP5: Li, Y., Jiang, T., Zhou, W. et al. Pan-cancer characterization of immune-related lncRNAs identifies potential oncogenic biomarkers. Nat Commun 11, 1000 (2020). HCP5: Sun J, Zhang Z, Bao S, et alIdentification of tumor immune infiltration-associated lncRNAs for improving prognosis and immunotherapy response of patients with non-small cell lung cancerJournal for ImmunoTherapy of Cancer 2020;8:e000110.

LINC00657: Feng Q, Zhang H, Yao D, Chen WD, Wang YD. Emerging Role of Non-Coding RNAs in Esophageal Squamous Cell Carcinoma. Int J Mol Sci. 2019 Dec 30;21(1):258. doi: 10.3390/ijms21010258.

LINC00657: Qin X, Zhou M, Lv H, Mao X, Li X, Guo H, Li L, Xing H. Long noncoding RNA LINC00657 inhibits cervical cancer development by sponging miR-20a-5p and targeting RUNX3. Cancer Lett. 2020 Oct 28:S0304-3835(20)30578-4. doi: 10.1016/j.canlet.2020.10.044. **LINC00657**: Zhang XM, Wang J, Liu ZL, Liu H, Cheng YF, Wang T. LINC00657/miR-26a-5p/CKS2 ceRNA network promotes the growth of esophageal cancer cells via the MDM2/p53/Bcl2/Bax pathway. Biosci Rep. 2020;40(6):BSR20200525.

TCGA-STAD: Cancer Genome Atlas Research Network. Comprehensive molecular characterization of gastric adenocarcinoma. Nature. 2014 Sep 11;513(7517):202-9. doi: 10.1038/nature13480. TCGA.STAD MAF data: https://api.gdc.cancer.gov/data/c06465a3-50e7-46f7-b2dd-7bd654ca206b

6.4 Others

1. Newman, A. M., Liu, C. L., Green, M. R., Gentles, A. J., Feng, W., Xu, Y., … Alizadeh, A. A. (2015). Robust enumeration of cell subsets from tissue expression profiles. Nature Methods, 12(5), 453–457.

2. Vegesna R, Kim H, Torres-Garcia W, …, Verhaak R.*(2013). Inferring tumour purity and stromal and immune cell admixture from expression data. Nature Communications 4, 2612.

3. Rieder, D., Hackl, H., …, Sopper, S.* (2019). Molecular and pharmacological modulators of the tumor immune contexture revealed by deconvolution of RNA-seq data. Genome medicine, 11(1), 34.

4. Li, B., Severson, E., Pignon, J.-C., Zhao, H., Li, T., Novak, J., … Liu, X. S.* (2016). Comprehensive analyses of tumor immunity: implications for cancer immunotherapy. Genome Biology, 17(1), 174.

5. P. Charoentong et al.*, Pan-cancer Immunogenomic Analyses Reveal Genotype-Immunophenotype Relationships and Predictors of Response to Checkpoint Blockade. Cell Reports 18, 248-262 (2017).

6. Becht, E., Giraldo, N. A., Lacroix, L., Buttard, B., Elarouci, N., Petitprez, F., … de Reyniès, A*. (2016). Estimating the population abundance of tissue-infiltrating immune and stromal cell populations using gene expression. Genome Biology, 17(1), 218.

7. Aran, D., Hu, Z., & Butte, A. J.* (2017). xCell: digitally portraying the tissue cellular heterogeneity landscape. Genome Biology, 18(1), 220.

8. Racle, J., de Jonge, K., Baumgaertner, P., Speiser, D. E., & Gfeller, D*. (2017). Simultaneous enumeration of cancer and immune cell types from bulk tumor gene expression data. ELife, 6, e26476.

9. Barbie, D.A. et al (2009). Systematic RNA interference reveals that oncogenic KRAS-driven cancers require TBK1. Nature, 462(5):108-112.

10. Hänzelmann, S., Castelo, R. and Guinney, J. (2013). GSVA: Gene set variation analysis for microarray and RNA-Seq data. BMC Bioinformatics, 14(1):7.

11. Lee, E. et al (2008). Inferring pathway activity toward precise disease classification. PLoS Comp Biol, 4(11):e1000217.

12. Wang et al.,et al (2019). The UCSCXenaTools R package: a toolkit for accessing genomics data from UCSC Xena platform, from cancer multi-omics to single-cell RNA-seq. Journal of Open Source Software, 4(40), 1627

13. Helmink BA, Reddy SM, Gao J, et al. B cells and tertiary lymphoid structures promote immunotherapy response. Nature. 2020 Jan;577(7791):549-555.

14. Mariathasan S, Turley SJ, Nickles D, et al. TGF attenuates tumour response to PD-L1 blockade by contributing to exclusion of T cells. Nature. 2018 Feb 22;554(7693):544-548.

15. Kulski, J.K. Long Noncoding RNA HCP5, a Hybrid HLA Class I Endogenous Retroviral Gene: Structure, Expression, and Disease Associations. Cells 2019, 8, 480.

16. Li, Y., Jiang, T., Zhou, W. et al. Pan-cancer characterization of immune-related lncR-

NAs identifies potential oncogenic biomarkers. Nat Commun 11, 1000 (2020).

17. Sun J, Zhang Z, Bao S, et alIdentification of tumor immune infiltration-associated lncRNAs for improving prognosis and immunotherapy response of patients with non-small cell lung cancerJournal for ImmunoTherapy of Cancer 2020;8:e000110.

18. Feng Q, Zhang H, Yao D, Chen WD, Wang YD. Emerging Role of Non-Coding RNAs in Esophageal Squamous Cell Carcinoma. Int J Mol Sci. 2019 Dec 30;21(1):258. doi: 10.3390/ijms21010258.

19. Qin X, Zhou M, Lv H, Mao X, Li X, Guo H, Li L, Xing H. Long noncoding RNA LINC00657 inhibits cervical cancer development by sponging miR-20a-5p and targeting RUNX3. Cancer Lett. 2020 Oct

20. Zhang XM, Wang J, Liu ZL, Liu H, Cheng YF, Wang T. LINC00657/miR-26a-5p/CKS2 ceRNA network promotes the growth of esophageal cancer cells via the MDM2/p53/Bcl2/Bax pathway. Biosci Rep. 2020;40(6):BSR20200525.

21. Cancer Genome Atlas Research Network. Comprehensive molecular characterization of gastric adenocarcinoma. Nature. 2014 Sep 11;513(7517):202-9. doi: 10.1038/nature13480.