Analyze CRISPRi growth competition data after treatment with or without Lysine

Ute Hoffmann (Science For Life Laboratory (KTH), Stockholm, Sweden)

januari 18, 2024

Contents

1 Aim of the analysis **1**

2 Analysis **1**

2.1 Diagnostic plot to check if control sgRNAs look ok 1

2.2 Add annotation to results tables 2

2.3 Visualization 3

2.4 Gene set enrichment analysis 4

Session info **9**

1 Aim of the analysis

Basic visualization of CRISPRi data for cultivation with lysine. Data analysis was performed using nf-core-crispriscreen pipeline (https://github.com/MPUSP/nf-core-crispriscreen).

2 Analysis

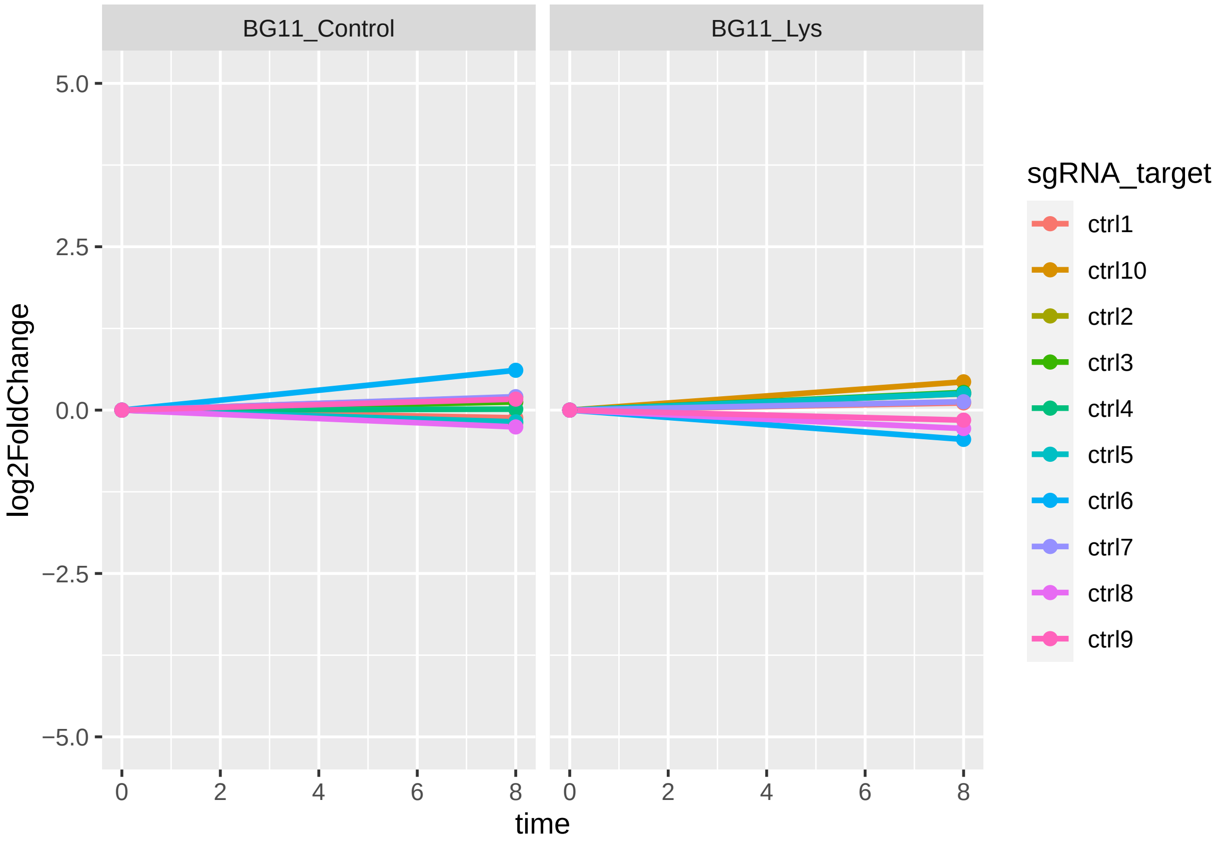
In a first step, the results given by the Nextflow pipeline are loaded.

```r
load("../results/fitness/result.Rdata")
```

2.1 Diagnostic plot to check if control sgRNAs look ok

Several control sgRNAs are included in the CRISPRi library. These control sgRNAs do not target any specific gene and serve as a control. Here, all of them behave neutrally.

```r
plot_controls_sgRNAs <- DESeq_result_table %>% filter(grepl("ctrl", sgRNA_target)) %>%
  ggplot(aes(x = time, y = log2FoldChange, color = sgRNA_target)) +
  geom_line(linewidth = 1) + geom_point(size = 2) + ylim(-5, 5) + facet_wrap(~ condition, ncol = 4)
print(plot_controls_sgRNAs)
```



```
ggsave("../R_results/plot_control_sgRNAs.pdf", plot=plot_controls_sgRNAs, width=12, height=12, units="cr
```

2.2 Add annotation to results tables

In the following, annotation is added to the results table provided by the Nextflow pipeline. Mapping of the sgRNA targets to slr-locus tags is given in this file, downloaded on 24/02/23: https://github.com/m-jahn/R-notebook-crispri-lib/blob/master/sgRNA_library_V2/data/input/mapping_trivial_names.tsv The appended annotation is based on Uniprot and Cyanobase, partially edited manually. The table used for annotation was created beginning of 2021. Therefore, it does not include several genes which were only recently characterized. For a detailed description of all the columns given in the results tables, consult https://mpusp.github.io/nf-core-crispriscreen/output or https://www.biorxiv.org/content/10.1101/2023.02.13.528328v1.full.pdf+htmls

```
mapping_gene_locus <- read_tsv("../input/2023-02-24_mapping_trivial_names.tsv", show_col_types=FALSE)
names(mapping_gene_locus) <- c("sgRNA_target", "locus")
DESeq_result_table <- DESeq_result_table %>% left_join(mapping_gene_locus)
```

```
annotation <- read_tsv("../input/annotation_locusTags_stand13012021.csv", show_col_types = FALSE)
annotation_2 <- annotation[,c(1,2,3)]
names(annotation_2) <- c("locus", "Gene name","Product")
DESeq_result_table <- DESeq_result_table %>% left_join(annotation_2)
```

```
write_tsv(DESeq_result_table, file="../R_results/annotated_DESeq_result_table.tsv")
df_reduced_info <- unique(subset(DESeq_result_table, DESeq_result_table$time==8 | DESeq_result_table$tin
write_tsv(df_reduced_info, file="../R_results/Reduced_annotated_DESeq_result_table.tsv")
```

```
df_red_wide <- pivot_wider(df_reduced_info, names_from=condition, values_from=c(wmean_fitness, sd_fitnes
write_tsv(df_red_wide, file="../R_results/Wide_DESeq_result_table.tsv")
```

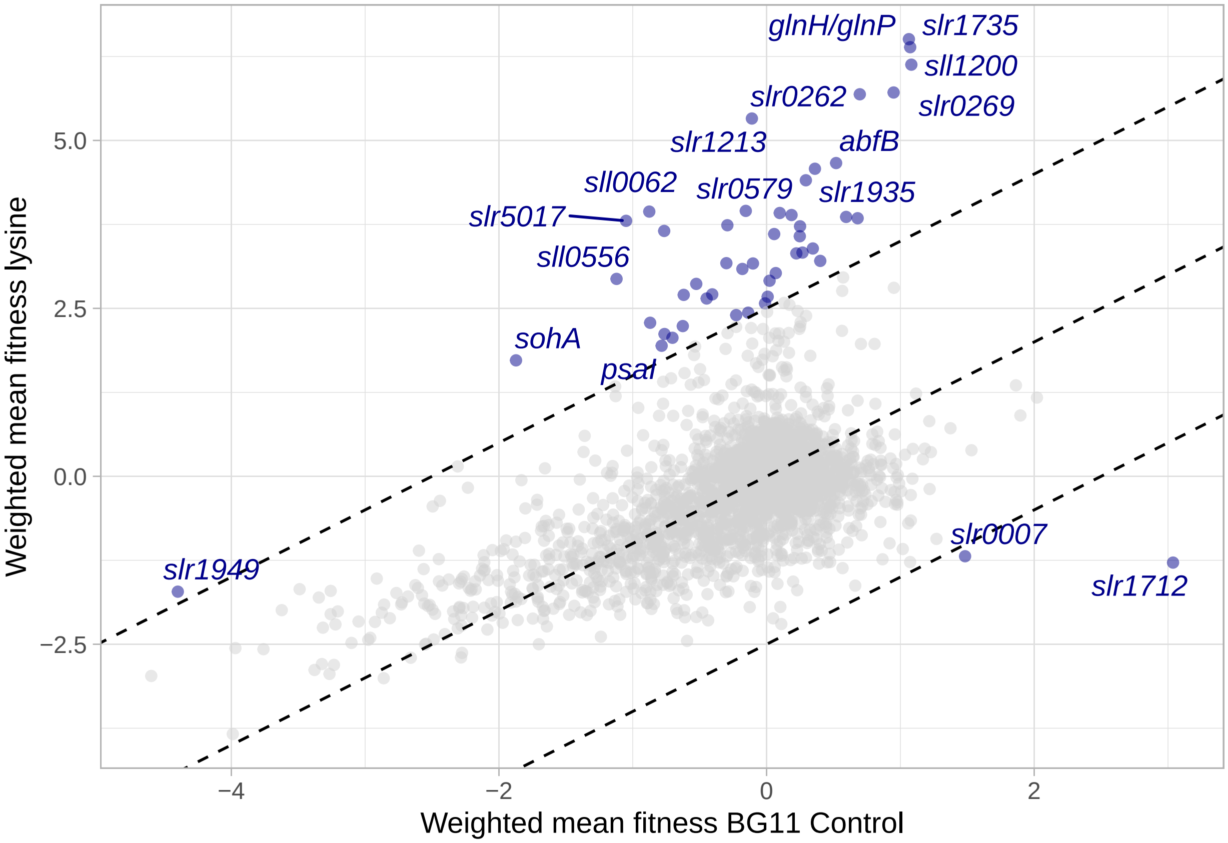
2.3 Visualization

The weighted mean fitness value combines the values of the different sgRNAs targeting the same gene. Fitness-fitness plots were created to identify genes which behave differently when lysine was added compared to the control cultivation. This was performed separately for ncRNAs and protein-coding genes.

2.3.1 Protein-coding genes

```
df_reduced <- unique(subset(DESeq_result_table, DESeq_result_table$time==8)[,c(2,4,20)])
df_red_ncRNAs <- subset(df_reduced, grepl("nc_", df_reduced$sgRNA_target))
df_red_no_ncRNAs <- subset(df_reduced, !grepl("nc_", df_reduced$sgRNA_target))
df_red_wide <- pivot_wider(df_red_no_ncRNAs,names_from="condition", values_from=c("wmean_fitness"))
```

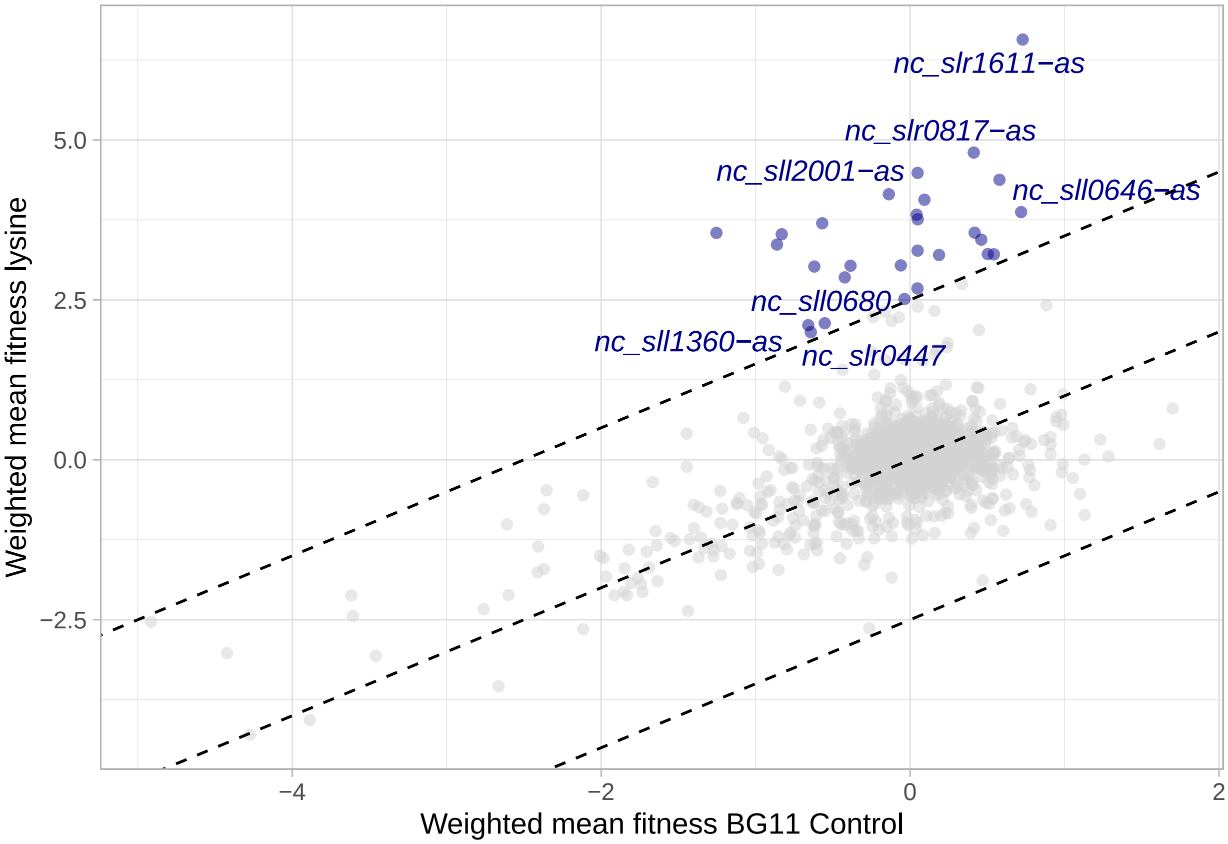
```
plot_fitness_fitness(df_red_wide, "BG11_Lys", y_axis_label="Weighted mean fitness lysine", filename_sav
```



2.3.2 ncRNAs

These include antisense RNAs, but also other ncRNAs.

```
df_red_wide_ncRNAs <- pivot_wider(df_red_ncRNAs,names_from="condition", values_from=c("wmean_fitness"))
plot_fitness_fitness(df_red_wide_ncRNAs, "BG11_Lys", y_axis_label="Weighted mean fitness lysine", filena
```



2.4 Gene set enrichment analysis

Functional enrichment analyses and gene set enrichment analyses help to check if a certain pathway or specific group of genes is especially affected by a treatment. Here, gene set enrichment analyses were performed for either Gene Ontology terms or KEGG pathways. To perform a gene set enrichment analysis, genes are sorted according to some measure, e.g. the log2FC after a certain time or the calculated fitness. Here, we used the weighted fitness of several sgRNAs as measure. The mapping of locus tags to Gene Ontology terms was downloaded from UniProt on the 18th Jan. 2024. There is the possibility to somehow weigh the adjusted p value in these calculation, e.g. by multiplying the weighted mean with the adjusted p value. Here, only the first few rows of each table is given. Full tables with all found terms/pathways are available.

In a first step, GSEAs were calculated for the control and the Lys-treated CRISPRi libraries separately. The depletion of essential pathways related to “Ribosomes” or “photosynthesis” is a first good quality measure for a CRISPRi screen.

2.4.1 BG11 Control, no amino acid added

```
DESeq_result_table_control <- unique(subset(DESeq_result_table, DESeq_result_table$condition=="BG11_Con
geneList <- DESeq_result_table_control$wmean_fitness
names(geneList) <- DESeq_result_table_control$locus
geneList = sort(geneList, decreasing = TRUE)
set.seed(513)
go_gsea_object <- GSEA(geneList, TERM2GENE = term_to_gene, TERM2NAME=term_to_name, seed=TRUE)
print(head(go_gsea_object)[,columns_to_show])
```

```
##                                                   Description setSize enrichmentScore
## GO:0031676 plasma membrane-derived thylakoid membrane     114      -0.7286053
## GO:0003735         structural constituent of ribosome      56      -0.7822414
## GO:0006412                                translation      63      -0.7593804
## GO:0048038                            quinone binding      16      -0.9159494
## GO:0008137   NADH dehydrogenase (ubiquinone) activity      15      -0.9057417
## GO:0019843                               rRNA binding      36      -0.7773143
##                  NES     p.adjust       qvalue
## GO:0031676 -1.848210 1.210000e-08 9.157895e-09
## GO:0003735 -1.918585 1.240726e-06 9.390448e-07
## GO:0006412 -1.869089 1.400688e-06 1.060112e-06
## GO:0048038 -1.866769 4.594211e-06 3.477133e-06
## GO:0008137 -1.835173 4.986341e-05 3.773916e-05
## GO:0019843 -1.826317 1.983321e-04 1.501078e-04
```

```
write.csv(go_gsea_object, "../R_results/GSEA_output/GO_GSEA_CRISPRi_control.csv")

set.seed(914)
kegg_gsea_object <- gseKEGG(geneList, organism="syn", minGSSize=10, pvalueCutoff = 0.05, seed=TRUE)
print(head(kegg_gsea_object)[,columns_to_show_KEGG])
```

```
##                ID
## syn03010 syn03010
## syn00190 syn00190
## syn01110 syn01110
## syn00195 syn00195
## syn01232 syn01232
## syn01230 syn01230
##                                                                  Description
## syn03010                              Ribosome - Synechocystis sp. PCC 6803
## syn00190             Oxidative phosphorylation - Synechocystis sp. PCC 6803
## syn01110 Biosynthesis of secondary metabolites - Synechocystis sp. PCC 6803
## syn00195                        Photosynthesis - Synechocystis sp. PCC 6803
## syn01232                 Nucleotide metabolism - Synechocystis sp. PCC 6803
## syn01230           Biosynthesis of amino acids - Synechocystis sp. PCC 6803
##          setSize enrichmentScore     p.adjust
## syn03010      54      -0.7853864 6.519030e-07
## syn00190      49      -0.7771898 6.570400e-06
## syn01110     287      -0.5778890 6.570400e-06
## syn00195      63      -0.6915508 1.175414e-03
## syn01232      27      -0.7788486 1.634279e-03
## syn01230      93      -0.6190411 3.846009e-03
```

```
write.csv(kegg_gsea_object, "../R_results/GSEA_output/KEGG_GSEA_CRISPRi_control.csv")
```

2.4.2 Lys added

```
DESeq_result_table_Lys <- unique(subset(DESeq_result_table, DESeq_result_table$condition=="BG11_Lys" & 
geneList <- DESeq_result_table_Lys$wmean_fitness
names(geneList) <- DESeq_result_table_Lys$locus
geneList = sort(geneList, decreasing = TRUE)
set.seed(513)
go_gsea_object <- GSEA(geneList, TERM2GENE = term_to_gene, TERM2NAME=term_to_name, seed=TRUE)
print(head(go_gsea_object)[,columns_to_show])
```

```
##                                            Description setSize enrichmentScore
## GO:0003735         structural constituent of ribosome      56      -0.8398701
## GO:0006412                               translation      63      -0.8153030
## GO:0019843                              rRNA binding      36      -0.8565056
## GO:0031676 plasma membrane-derived thylakoid membrane     114      -0.6487402
## GO:0022625          cytosolic large ribosomal subunit      20      -0.8842480
## GO:0005737                                 cytoplasm     304      -0.5100594
##                  NES     p.adjust       qvalue
## GO:0003735 -2.493330 3.569327e-09 2.390937e-09
## GO:0006412 -2.465058 3.569327e-09 2.390937e-09
## GO:0019843 -2.377284 3.569327e-09 2.390937e-09
## GO:0031676 -2.107224 3.569327e-09 2.390937e-09
## GO:0022625 -2.149936 2.411319e-08 1.615238e-08
## GO:0005737 -1.822332 3.537335e-08 2.369507e-08
```

```r
write.csv(go_gsea_object, "../R_results/GSEA_output/GO_GSEA_CRISPRi_Lys.csv")

set.seed(914)
kegg_gsea_object <- gseKEGG(geneList, organism="syn", minGSSize=10, pvalueCutoff = 0.05, seed=TRUE)
print(head(kegg_gsea_object)[,columns_to_show_KEGG])
```

```
##                ID
## syn03010 syn03010
## syn01110 syn01110
## syn00190 syn00190
## syn01230 syn01230
## syn01240 syn01240
## syn01232 syn01232
##                                                                  Description
## syn03010                              Ribosome - Synechocystis sp. PCC 6803
## syn01110 Biosynthesis of secondary metabolites - Synechocystis sp. PCC 6803
## syn00190             Oxidative phosphorylation - Synechocystis sp. PCC 6803
## syn01230           Biosynthesis of amino acids - Synechocystis sp. PCC 6803
## syn01240             Biosynthesis of cofactors - Synechocystis sp. PCC 6803
## syn01232                 Nucleotide metabolism - Synechocystis sp. PCC 6803
##          setSize enrichmentScore     p.adjust
## syn03010      54      -0.8689781 3.300000e-09
## syn01110     287      -0.5762084 3.300000e-09
## syn00190      49      -0.7275040 6.190555e-07
## syn01230      93      -0.6333268 6.190555e-07
## syn01240     138      -0.5501202 3.151174e-05
## syn01232      27      -0.7675151 5.506984e-05
```

```r
write.csv(kegg_gsea_object, "../R_results/GSEA_output/KEGG_GSEA_CRISPRi_Lys.csv")
```

2.4.3 Difference between control and Lys treatment

In a next step, we tried to check which GO terms or KEGG pathways show a divergent enrichment or depletion in the two libraries. For this, weighted fitness means belonging to the two conditions were subtracted from each other. These differences were used as input for the GSEA.

```r
df_difference <- unique(subset(DESeq_result_table, DESeq_result_table$time==8 & !is.na(DESeq_result_tabl
df_difference_wide <- pivot_wider(df_difference, names_from=condition, values_from=wmean_fitness)
df_difference_wide$difference <- df_difference_wide$BG11_Control - df_difference_wide$BG11_Lys
write_tsv(df_difference_wide, file="../R_results/fitness_difference_table.tsv")

geneList <- df_difference_wide$difference
names(geneList) <- df_difference_wide$locus
```

```
geneList = sort(geneList, decreasing = TRUE)
set.seed(513)
go_gsea_object <- GSEA(geneList, TERM2GENE = term_to_gene, TERM2NAME=term_to_name, seed=TRUE)
print(head(go_gsea_object)[,columns_to_show])
```

```
##                                   Description setSize enrichmentScore       NES
## GO:0006412                        translation      63       0.6473716  2.329588
## GO:0003735 structural constituent of ribosome      56       0.6687213  2.321194
## GO:0016020                           membrane     455      -0.4338693 -1.712730
## GO:0019843                       rRNA binding      36       0.6937528  2.180634
## GO:0022625  cytosolic large ribosomal subunit      20       0.7002752  1.956251
## GO:0005737                          cytoplasm     304       0.3285142  1.452559
##                p.adjust       qvalue
## GO:0006412 9.983351e-07 8.684951e-07
## GO:0003735 9.983351e-07 8.684951e-07
## GO:0016020 5.989312e-06 5.210363e-06
## GO:0019843 4.424678e-05 3.849219e-05
## GO:0022625 1.051533e-02 9.147741e-03
## GO:0005737 1.051533e-02 9.147741e-03
```

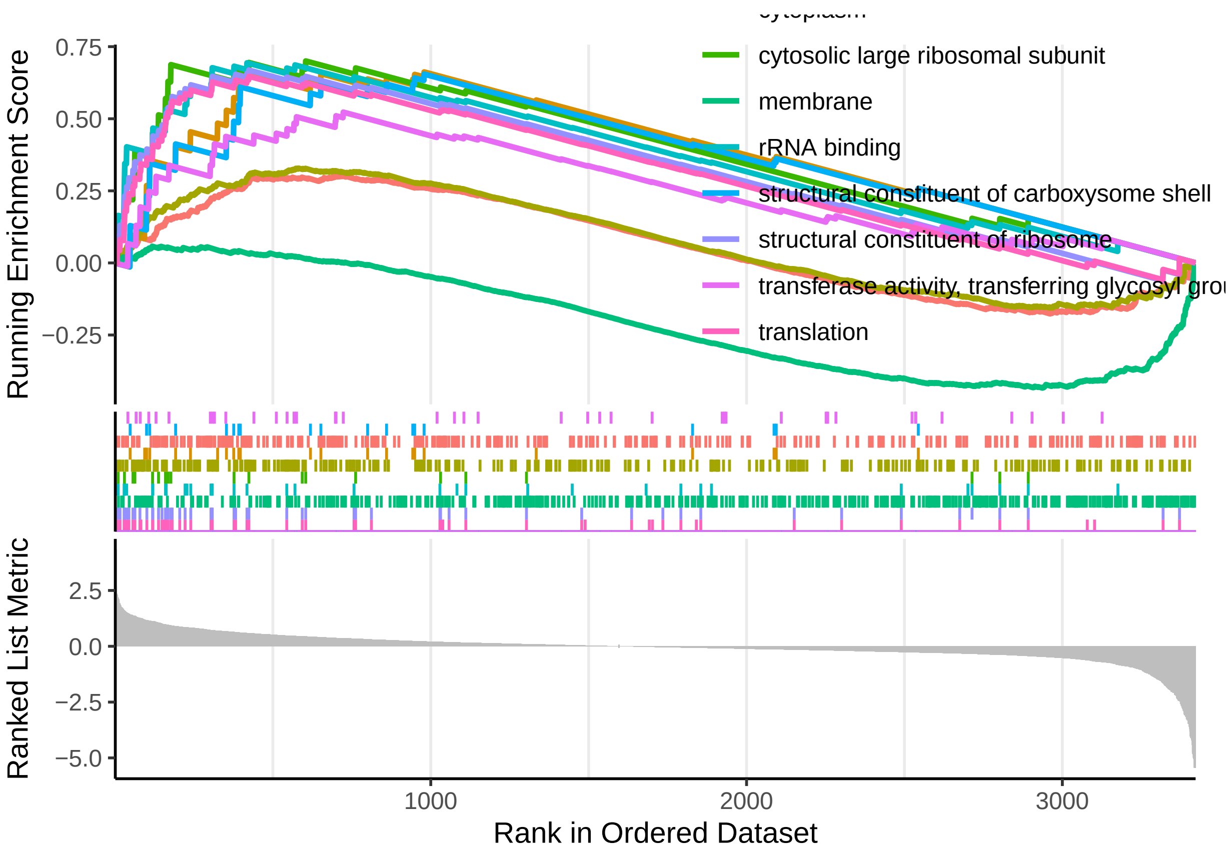
```
write.csv(go_gsea_object, "../R_results/GSEA_output/GO_GSEA_difference_control_Lys.csv")

set.seed(914)
kegg_gsea_object <- gseKEGG(geneList, organism="syn", minGSSize=10, pvalueCutoff = 0.05, seed=TRUE)
print(head(kegg_gsea_object)[,columns_to_show_KEGG])
```

```
##                ID
## syn03010 syn03010
## syn01240 syn01240
## syn01110 syn01110
## syn00970 syn00970
## syn01210 syn01210
## syn00290 syn00290
##                                                                          Description
## syn03010                                    Ribosome - Synechocystis sp. PCC 6803
## syn01240                   Biosynthesis of cofactors - Synechocystis sp. PCC 6803
## syn01110       Biosynthesis of secondary metabolites - Synechocystis sp. PCC 6803
## syn00970                 Aminoacyl-tRNA biosynthesis - Synechocystis sp. PCC 6803
## syn01210             2-Oxocarboxylic acid metabolism - Synechocystis sp. PCC 6803
## syn00290 Valine, leucine and isoleucine biosynthesis - Synechocystis sp. PCC 6803
##          setSize enrichmentScore     p.adjust
## syn03010      54       0.7205238 1.691723e-08
## syn01240     138       0.4674988 4.833635e-05
## syn01110     287       0.3404239 3.596327e-03
## syn00970      26       0.6244211 2.027942e-02
## syn01210      26       0.6033956 2.506559e-02
## syn00290      12       0.7519904 2.506559e-02
```

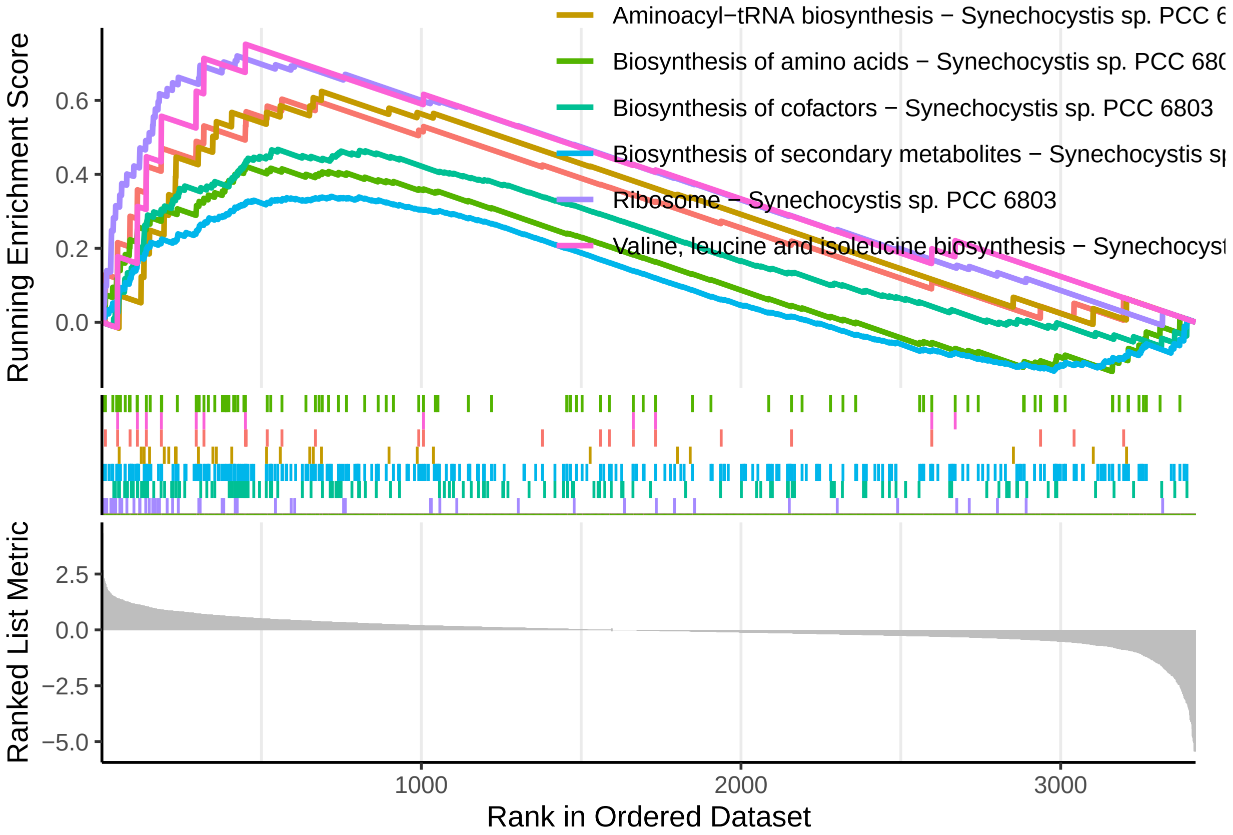
```
write.csv(kegg_gsea_object, "../R_results/GSEA_output/KEGG_GSEA_difference_control_Lys.csv")
```

Results for the comparison of the two libraries were visualized. These plots are separated in three panels. The lowest panel ("Ranked List Metric") shows the metric according to which the genes were sorted. In this case, this was the weighted fitness mean associated with the different genes. The upper panel shows the running enrichment score of terms/pathways which were enriched/depleted in a statistically significant manner. The middle panel shows where the genes associated with these terms are located within the ranked list of genes in the same color code as used in the upper panel. It is worth mentioning that some GO terms, such as "ATP binding", "cytoplasm" and "membrane" encompass gigantic gene sets. Same holds for some KEGG terms, e.g. "Biosynthesis of secondary metabolites" or "Biosynthesis of cofactors". I am personally always not quite sure how meaniningful it is if such huge sets are enriched or depleted.

```
p <- gseaplot2(go_gsea_object, geneSetID =1:10)
p
```



```
ggsave("../R_results/GSEA_output/GO_GSEA_differences.pdf", plot=p, width=20, height=30, units="cm")
p <- gseaplot2(kegg_gsea_object, geneSetID =1:7)
p
```



```
ggsave("../R_results/GSEA_output/KEGG_GSEA_differences.pdf", plot=p, width=15, height=18, units="cm")
# browseKEGGNew_3(kegg_gsea_object, "syn03010", 1)
# browseKEGGNew_3(kegg_gsea_object, "syn01240", 2)
# browseKEGGNew_3(kegg_gsea_object, "syn01110", 3)
# browseKEGGNew_3(kegg_gsea_object, "syn00970", 4)
# browseKEGGNew_3(kegg_gsea_object, "syn01210", 5)
# browseKEGGNew_3(kegg_gsea_object, "syn00290", 6)
```

Session info

```
## R version 4.3.2 (2023-10-31)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 22.04.3 LTS
##
## Matrix products: default
## BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
## LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.20.so;  LAPACK version 3.10.0
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=sv_SE.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=sv_SE.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=sv_SE.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=sv_SE.UTF-8 LC_IDENTIFICATION=C
##
## time zone: Europe/Stockholm
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] enrichplot_1.20.3     clusterProfiler_4.8.3 magrittr_2.0.3
##  [4] forcats_0.5.2         stringr_1.5.0         dplyr_1.0.10
##  [7] purrr_1.0.2           readr_2.1.4           tidyr_1.3.0
## [10] tibble_3.2.1          tidyverse_1.3.1       ggpubr_0.6.0
## [13] ggrepel_0.9.4         ggplot2_3.4.4         knitr_1.45
##
## loaded via a namespace (and not attached):
##   [1] RColorBrewer_1.1-3      rstudioapi_0.14         jsonlite_1.8.7
##   [4] farver_2.1.1            rmarkdown_2.25          ragg_1.2.6
##   [7] fs_1.6.3                zlibbioc_1.46.0         vctrs_0.6.4
##  [10] memoise_2.0.1           RCurl_1.98-1.13         ggtree_3.8.2
##  [13] rstatix_0.7.2           htmltools_0.5.7         haven_2.5.1
##  [16] broom_1.0.1             cellranger_1.1.0        gridGraphics_0.5-1
##  [19] plyr_1.8.9              lubridate_1.8.0         cachem_1.0.8
##  [22] igraph_1.5.1            lifecycle_1.0.4         pkgconfig_2.0.3
##  [25] gson_0.1.0              Matrix_1.6-3            R6_2.5.1
##  [28] fastmap_1.1.1           GenomeInfoDbData_1.2.10 digest_0.6.33
##  [31] aplot_0.2.2             colorspace_2.1-0        patchwork_1.1.3
##  [34] AnnotationDbi_1.62.2    S4Vectors_0.38.2        textshaping_0.3.7
##  [37] RSQLite_2.3.3           labeling_0.4.2          fansi_1.0.5
##  [40] httr_1.4.4              polyclip_1.10-6         abind_1.4-5
##  [43] compiler_4.3.2          bit64_4.0.5             withr_2.5.0
##  [46] downloader_0.4          backports_1.4.1         BiocParallel_1.34.2
##  [49] carData_3.0-5           viridis_0.6.4           DBI_1.1.3
##  [52] highr_0.10              ggforce_0.4.1           ggsignif_0.6.4
##  [55] MASS_7.3-60             HDO.db_0.99.1           tools_4.3.2
##  [58] scatterpie_0.2.1        ape_5.7-1               glue_1.6.2
##  [61] nlme_3.1-163            GOSemSim_2.26.1         shadowtext_0.1.2
##  [64] grid_4.3.2              reshape2_1.4.4          fgsea_1.26.0
##  [67] generics_0.1.3          gtable_0.3.4            tzdb_0.4.0
##  [70] data.table_1.14.8       hms_1.1.3               tidygraph_1.2.3
##  [73] xml2_1.3.5              car_3.1-2               utf8_1.2.4
##  [76] XVector_0.40.0          BiocGenerics_0.46.0     pillar_1.9.0
##  [79] vroom_1.6.4             yulab.utils_0.1.0       splines_4.3.2
##  [82] tweenr_2.0.2            treeio_1.24.3           lattice_0.22-5
##  [85] bit_4.0.5               tidyselect_1.2.0        GO.db_3.17.0
##  [88] Biostrings_2.68.1       gridExtra_2.3           IRanges_2.34.1
##  [91] stats4_4.3.2            xfun_0.41               graphlayouts_1.0.2
##  [94] Biobase_2.60.0          stringi_1.7.12          lazyeval_0.2.2
##  [97] ggfun_0.1.3             yaml_2.3.7              evaluate_0.23
## [100] codetools_0.2-19        ggraph_2.1.0            qvalue_2.32.0
## [103] ggplotify_0.1.2         cli_3.6.1               systemfonts_1.0.4
## [106] munsell_0.5.0           modelr_0.1.9            Rcpp_1.0.11
## [109] GenomeInfoDb_1.36.4     readxl_1.4.3            dbplyr_2.2.1
## [112] png_0.1-8               parallel_4.3.2          assertthat_0.2.1
## [115] blob_1.2.3              DOSE_3.26.2             reprex_2.0.2
## [118] bitops_1.0-7            viridisLite_0.4.2       tidytree_0.4.5
## [121] scales_1.2.1            crayon_1.5.2            rlang_1.1.2
## [124] cowplot_1.1.2           fastmatch_1.1-4         KEGGREST_1.40.1
## [127] rvest_1.0.3
```