Sample distribution analysis

Antonia Chroni achroni@stjude.org for St. Jude Children's Research Hospital BioHackathon Team 1

Contents

1 Information about this notebook **3**

2 Set up **3**

3 Directories and paths to file Inputs/Outputs **3**

4 Read metadata file **3**
4.1 Generate SJUID 4
4.2 Number of samples with assay information 4
4.3 Number of assays per brain cancer type 4

5 Number of samples per Assay **6**

6 Number of samples per brain cancer type and Assay **6**

7 Number of samples per brain cancer type, Assay, and SJUID **6**

8 Future directions **9**

9 Session Info **11**

```
## The following object is masked _by_ .GlobalEnv:
##
##     root_dir
```

Project: Comprehensive Omics Catalogue for Hartwell

St. Jude Children's Research Hospital BioHackathon Team 1

Date started: 09/04/2024 Date completed: 9/06/2024 Report generated: 13:45:48 CDT 09/05/2024

1 Information about this notebook

This is an exploratory analysis of the data availability in terms of assays in the Comprehensive Omics Catalogue for Hartwell. This is critical for mitigating duplicate sequencing requests and efforts through Hartwell. This notebook aims to showcase: (1) what data have already been sequenced by Hartwell and (2) what omics data are available per sample.

For demo purposes, we subset by human brain tumor samples. We investigate the number of samples per `cancer_type_brain` and `Assay`.

2 Set up

```
suppressPackageStartupMessages({
  library(tidyverse)
  library(flextable)
})
```

3 Directories and paths to file Inputs/Outputs

```
attach(params)
```

```
## The following object is masked _by_ .GlobalEnv:
##
##     root_dir
```

```
analysis_dir <- file.path(root_dir, "analyses", "sample-distribution-analysis")

# We will first read in metadata file as we need to define sample_name
metadata_file <- file.path(analysis_dir, "input", input_file) # metadata input file

# File path to `results` directory
results_dir <- file.path(analysis_dir, "results")
if (!dir.exists(results_dir)) {
  dir.create(results_dir)}
```

4 Read metadata file

We will subset by human brain tumor samples.

```
# Read metadata
project_df <- read.csv(metadata_file, stringsAsFactors=FALSE) %>%

  # Add cancer_type_brain: Ependymoma, HGG, LGG, Medulloblastoma
  add_column(cancer_type_brain = "other") %>%
  mutate(cancer_type_brain = case_when(grepl("Ependymoma", Disease) ~ "Ependymoma",
                                       grepl("HGG", Disease) ~ "HGG",
                                       grepl("LGG", Disease) ~ "LGG",
                                       grepl("MedulloBlastoma", Disease) ~ "Medulloblastoma"),
         Assay = Omics.Method) %>%
  mutate(across(where(is.character), ~ na_if(.,""))) %>% # Omics Method for NA
  filter(!cancer_type_brain == "other",
         !is.na(cancer_type_brain),
         !is.na(Omics.Method),
```

```
         Source == "Human") %>%
  select(Source, Disease, Assay, Omics.Method.Detail, Site, Sub.Group, cancer_type_brain)
```

4.1 Generate SJUID

We will generate random SJUID per brain cancer type as this information is not contained in the current data.

```
# Make this reproducible
set.seed(2024)

# create vector of data$Sample.SJUID with some duplicates
generate_string <- function(length) {
  chars <- c(LETTERS, LETTERS, 0:9)
  paste0(sample(chars, length, replace = TRUE), collapse = "")
}

# Create a smaller set of unique strings, each starting with "SJH"
unique_strings <- paste0("SJH", sapply(1:80, function(x) generate_string(8)))

# Sample from this set to create a vector of 100 strings, allowing duplicates
SJUID <- sample(unique_strings, 100, replace = TRUE)

# Generate vector for `cancer_type_brain`
cancer_type_brain_vec <- c("Ependymoma", "HGG", "LGG", "Medulloblastoma")
n <- 25 # random number
cancer_type_brain <- rep(cancer_type_brain_vec, each=n)

# Assign `SJUID` to `cancer_type_brain`
bind_df <- cbind(SJUID, cancer_type_brain)  %>%
  as.data.frame()

# Merge both df
df <- project_df %>%
  left_join(bind_df, by = "cancer_type_brain", relationship = "many-to-many") %>%
  unique() %>%
  mutate(match_id = paste(SJUID, Assay, sep = "_")) %>%
  distinct(match_id, .keep_all = TRUE) %>%
  write_tsv(file.path(results_dir, "cohort.tsv")) # save


# Number of samples per cancer_type_brain
assays_number <- length(df$SJUID)
samples_number <- length(unique(df$SJUID))
```

4.2 Number of samples with assay information

There are 60 brain tumor samples with 261 assays in total.

4.3 Number of assays per brain cancer type

Table 1: Summary of assays per brain cancer type

cancer_type_brain	n
Ependymoma	120
HGG	28
LGG	102
Medulloblastoma	11

5 Number of samples per Assay

Assay	n
Methylation	60
RNAseq	49
ChIPseq	37
WES	37
WGS	37
ATACseq	21
WGBS	20

6 Number of samples per brain cancer type and Assay

cancer_type_brain	ATACseq	ChIPseq	Methylation	RNAseq	WES	WGBS	WGS
LGG	21	17	17	13	17		17
Ependymoma		20	20	20	20	20	20
HGG			12	16			
Medulloblastoma			11				

7 Number of samples per brain cancer type, Assay, and SJUID

cancer_type_brain	SJUID	ATACseq	ChIPseq	Methylation	RNAseq	WES	WGBS	WGS
LGG	SJH2W47P7DG	1	1	1	1	1		1
LGG	SJH5HCKPC97	1						
LGG	SJHAY4GX4AN	1	1	1	1	1		1
LGG	SJHBN-JSZHW6	1	1	1	1	1		1
LGG	SJHBV3Q6UVR	1	1	1	1	1		1
LGG	SJHC70DZRJS	1						
LGG	SJH-CLGFJTIG	1	1	1	1	1		1
LGG	SJHCN03RCVD	1						

can-cer_type_brain	SJUID	ATACseq	ChIPseq	Methyla-tion	RNAseq	WES	WGBS	WGS
LGG	SJHD-DTE0SYL	1	1	1		1		1
LGG	SJHI52BLNWK	1	1	1	1	1		1
LGG	SJHIQT-NAYIF	1						
LGG	SJHN-PJTQHIT	1	1	1	1	1		1
LGG	SJHO-FORRR7C	1	1	1	1	1		1
LGG	SJHOY05OJJN	1	1	1	1	1		1
LGG	SJHQBOPS9FD	1	1	1		1		1
LGG	SJHUT-PISXFQ	1	1	1		1		1
LGG	SJHVIS5Q8HX	1	1	1		1		1
LGG	SJHWS0NRZVA	1	1	1	1	1		1
LGG	SJHXIKCWNKY	1	1	1	1	1		1
LGG	SJHYP-KTG3P5	1	1	1	1	1		1
LGG	SJHZW7GYEF9	1	1	1	1	1		1
Ependy-moma	SJH0H5WYREP		1	1	1	1	1	1
Ependy-moma	SJH2HPPEEKM		1	1	1	1	1	1
Ependy-moma	SJH51B396IW		1	1	1	1	1	1
Ependy-moma	SJH5HCKPC97		1	1	1	1	1	1
Ependy-moma	SJH98QNKIUU		1	1	1	1	1	1
Ependy-moma	SJH9YML-FOTI		1	1	1	1	1	1
Ependy-moma	SJHA6KC56J6		1	1	1	1	1	1
Ependy-moma	SJH-BKS7QSFO		1	1	1	1	1	1
Ependy-moma	SJHC70DZRJS		1	1	1	1	1	1
Ependy-moma	SJHCN03RCVD		1	1	1	1	1	1

can-cer_type_brain	SJUID	ATACseq	ChIPseq	Methyla-tion	RNAseq	WES	WGBS	WGS
Ependy-moma	SJHD1KI-UMM6		1	1	1	1	1	1
Ependy-moma	SJHFG-GJRHYO		1	1	1	1	1	1
Ependy-moma	SJH-HZH67WMF		1	1	1	1	1	1
Ependy-moma	SJHI8CC7QT8		1	1	1	1	1	1
Ependy-moma	SJHIQT-NAYIF		1	1	1	1	1	1
Ependy-moma	SJHKVQE-BOCP		1	1	1	1	1	1
Ependy-moma	SJHKYUIYAME		1	1	1	1	1	1
Ependy-moma	SJH-PVXLA-CLM		1	1	1	1	1	1
Ependy-moma	SJHUMKP2L6V		1	1	1	1	1	1
Ependy-moma	SJHXTBTQ5ZT		1	1	1	1	1	1
HGG	SJH5QHZM4US			1	1			
HGG	SJHADL5OJME			1	1			
HGG	SJH-BISK5KBU			1	1			
HGG	SJHBL7CDYRN			1	1			
HGG	SJHEOVURBMJ			1	1			
HGG	SJHEQG3P4FK			1	1			
HGG	SJHHPN-QERSQ			1	1			
HGG	SJHHW23UJ9P			1	1			
HGG	SJHJ59RLHSU			1	1			
HGG	SJHKKD-DXOYH			1	1			
HGG	SJHMI643DMD			1	1			
HGG	SJHRO-JUQZAP			1	1			
Medul-loblas-toma	SJH77NRD-WUX			1				

cancer_type_brain	SJUID	ATACseq	ChIPseq	Methylation	RNAseq	WES	WGBS	WGS
Medulloblastoma	SJH8HIE3P0P			1				
Medulloblastoma	SJHAF-TIOMPQ			1				
Medulloblastoma	SJHC1QST5GR			1				
Medulloblastoma	SJHF-CYGKSDY			1				
Medulloblastoma	SJHJAC-CGA3S			1				
Medulloblastoma	SJHQS0D51KH			1				
Medulloblastoma	SJHXVMEU21L			1				
Medulloblastoma	SJHY4W1ZWCY			1				
Medulloblastoma	SJHZ40PT-CYH			1				
Medulloblastoma	SJHZZLR-CGJ6			1				
HGG	SJHD-DTE0SYL				1			
HGG	SJHQBOPS9FD				1			
HGG	SJHUT-PISXFQ				1			
HGG	SJHVIS5Q8HX				1			

8 Future directions

The current exploratory data analysis can be expanded by investigating samples with paired assays. Moreover, if other metadata are available, e.g., disease_stage, treatment, this will lead to build large, longitudinal cohorts with multi-omic sequencing data. Such an analysis permits to consider samples according to the condition(s) of the experiment and research aims, accordingly. In addition, it can be used to refine research questions and/or generate new ones.

This will facilitate collaboration across departments at St Jude, expedite discoveries, and find cures for children with cancer and other catastrophic diseases.

9 Session Info

```
## R version 4.4.0 (2024-04-24)
## Platform: x86_64-pc-linux-gnu
## Running under: Red Hat Enterprise Linux 8.8 (Ootpa)
##
## Matrix products: default
## BLAS:   /research/rgs01/applications/hpcf/authorized_apps/rhel8_apps/lapack/3.10.1/install/lib64/libl
## LAPACK: /research/rgs01/applications/hpcf/authorized_apps/rhel8_apps/lapack/3.10.1/install/lib64/libl
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## time zone: America/Chicago
## tzcode source: system (glibc)
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] flextable_0.9.6 lubridate_1.9.3 forcats_1.0.0   stringr_1.5.1
##  [5] dplyr_1.1.4     purrr_1.0.2     readr_2.1.5     tidyr_1.3.1
##  [9] tibble_3.2.1    ggplot2_3.5.1   tidyverse_2.0.0
##
## loaded via a namespace (and not attached):
##  [1] gtable_0.3.5           xfun_0.47              bslib_0.8.0
##  [4] tzdb_0.4.0             vctrs_0.6.5            tools_4.4.0
##  [7] generics_0.1.3         parallel_4.4.0         curl_5.2.1
## [10] fansi_1.0.6            pkgconfig_2.0.3        data.table_1.15.4
## [13] uuid_1.2-1             lifecycle_1.0.4        compiler_4.4.0
## [16] textshaping_0.4.0      munsell_0.5.1          httpuv_1.6.15
## [19] fontquiver_0.2.1       fontLiberation_0.1.0   htmltools_0.5.8.1
## [22] sass_0.4.9             yaml_2.3.10            pillar_1.9.0
## [25] later_1.3.2            crayon_1.5.3           jquerylib_0.1.4
## [28] gfonts_0.2.0           openssl_2.2.1          cachem_1.1.0
## [31] mime_0.12              fontBitstreamVera_0.1.1 zip_2.3.1
## [34] tidyselect_1.2.1       digest_0.6.37          stringi_1.8.4
## [37] fastmap_1.2.0          grid_4.4.0             colorspace_2.1-1
## [40] cli_3.6.3              magrittr_2.0.3         crul_1.5.0
## [43] utf8_1.2.4             withr_3.0.1            gdtools_0.3.7
## [46] scales_1.3.0           promises_1.3.0         bit64_4.0.5
## [49] timechange_0.3.0       rmarkdown_2.28         officer_0.6.6
## [52] bit_4.0.5              askpass_1.2.0          ragg_1.3.2
## [55] hms_1.1.3              shiny_1.9.1            evaluate_0.24.0
## [58] knitr_1.48             rlang_1.1.4            Rcpp_1.0.13
## [61] xtable_1.8-4           glue_1.7.0             httpcode_0.3.0
## [64] xml2_1.3.6             vroom_1.6.5            jsonlite_1.8.8
## [67] R6_2.5.1               systemfonts_1.1.0
```