GBM_Spitzer2025_process

December 25, 2025

1 load data

[ ]:
```r
## read RDS file
file_list <- list.files(path = "/project/sex_cancer/data/GBM_Spitzer2025/
 ↪GSE274546/", pattern = "*.RDS", full.names = TRUE)
seurat_list <- lapply(file_list, function(file){
                        data <- readRDS(file)
                        seurat_object <- CreateSeuratObject(counts = data)
                        return(seurat_object)
                        })
obj.GBM <- Reduce(function(x, y) merge(x, y), seurat_list)
```

[ ]:
```r
md <- readRDS('/project/sex_cancer/data/GBM_Spitzer2025/
 ↪celltype_meta_data_2025_01_08.RDS')
meta <- readRDS('/project/sex_cancer/data/GBM_Spitzer2025/
 ↪care_full_gene_count_metadata.RDS')
rownames(meta) <- meta$CellID
meta <- meta[colnames(obj.GBM@assays$RNA),]
meta <- meta %>%
        merge(obj.GBM@meta.data, by = 'CellID', all = TRUE) %>%
        merge(md,by = 'CellID',all = TRUE)

info <- read_xlsx('/project/sex_cancer/data/GBM_Spitzer2025/GBM_Abishay2025.
 ↪xlsx',skip = 2)
colnames(info)[11]='Sex'
colnames(info)[12]='Age'

obj.GBM@meta.data <- merge(meta, info, by = 'Sample_ID_', all = TRUE)
rownames(obj.GBM@meta.data)=obj.GBM@meta.data$CellID
```

2 modify meta.data

[ ]:
```r
obj.GBM@meta.data <- obj.GBM@meta.data %>%
                     dplyr::rename(c('barcode' = 'CellID', 'SampleID' =␣
 ↪'Sample_ID', 'DonorID' = 'Patient_ID', 'TumorID' = 'Tumor_ID')) %>%
```

```
                    transform(SampleType = 'tumor', Cohort =␣
↪'GBM_Spitzer2025') %>%
                    transform(CellType = ext_list(CellType))
```

3 filter sample

[ ]:
```
obj.GBM <- obj.GBM %>%
           subset(Time_point == 'T1' & Primary_or_recurrent == 'Primary' &␣
↪Steroid_treatment_before_the_surgery == 'No' &
                  Radiation_before_the_surgery == 'No' &␣
↪Alkylate_agents_before_the_surgery == 'No' &
                 ␣
↪`Number_of_cycles_treated_by_adjuvant_TMZ_before_the_surgery_` == '-') %>%
           subset(is.na(CellType) == FALSE)
obj.GBM
```

4 cell type annotation

Reference: https://www.cell.com/trends/cancer/fulltext/S2405-8033(22)00233-3

4.1 assign oCT

[ ]:
```
obj.GBM@meta.data <- obj.GBM@meta.data %>%
                    transform(oCT = CellType)
```

4.2 check annotation

check cell type annotation provided in the original research via COSG

[ ]:
```
## check marker expression
marker_annotation <- readRDS("marker_annotation.rds")

obj <- obj.GBM
DefaultAssay(obj) <- "RNA"
obj <- obj %>% NormalizeData(normalization.method = "LogNormalize", scale.
↪factor = 10000, verbose = F)
Idents(obj) <- ext_list(obj$oCT)

marker_oCT <- obj %>%
              cosg(groups = "all", assay = "RNA", slot = "data",
              mu = 10, ## The penalty factor to penalize gene expression in␣
↪cells not belonging to the cluster of interest
              n_genes_user = 50, # Number of top ranked genes returned in the␣
↪result
```

```
            remove_lowly_expressed=T, # If TRUE, genes that express a␣
 ↪percentage of target cells smaller than a specific value (expressed_pct) are␣
 ↪not considered as marker genes for the target cells. The default value is␣
 ↪TRUE.
            expressed_pct=0.1) # If TRUE, genes that express a percentage of␣
 ↪target cells smaller than a specific value (expressed_pct) are not␣
 ↪considered as marker genes for the target cells.
marker_oCT <- cbind(marker_oCT[[1]] %>% melt(id.vars = NULL) %>% dplyr::
 ↪rename(c("oCT" = "variable", "marker" = "value")),
                    marker_oCT[[2]] %>% melt(id.vars = NULL) %>% dplyr::
 ↪select(-"variable") %>% dplyr::rename(c("COSGscore" = "value"))) %>%
            mutate(Cohort = unique(obj$Cohort)) %>% mutate(oCT =␣
 ↪ext_list(oCT))

oCT_marker <- marker_oCT
oCT_list <- unique(oCT_marker$oCT)
lapply(oCT_list, function(x){
        check <- oCT_marker %>% subset(oCT == x & marker %in%␣
 ↪marker_annotation[[x]])
        ifelse(nrow(check) == 0, print(x), return(check))
})
```

4.3 assign mCT

[ ]:
```
obj.GBM@meta.data  <-  obj.GBM@meta.data %>%
                       mutate(dCT = ext_list(CellType)) %>%
                       mutate(mCT = case_when(CellType %in% c('Malignant',␣
 ↪'Oligodendrocyte', 'Astrocyte', 'OPC') ~ 'Glia',
                                              CellType %in% c('Excitatory␣
 ↪neuron', 'Inhibitory neuron') ~ 'Neuron',
                                              CellType %in% c('TAM') ~ 'Mph',
                                              CellType %in% c('Lymphocyte') ~␣
 ↪'Lymphoid',
                                              CellType %in% c('Endothel') ~␣
 ↪'Endo',
                                              CellType %in% c('Pericyte') ~␣
 ↪'Pericyte',
                                              TRUE ~ 'Others'))
```

4.4 assign gCT

[ ]:
```
obj.GBM@meta.data <- obj.GBM@meta.data %>%
                     mutate(gCT = case_when(CellType %in% c('Malignant',␣
 ↪'Oligodendrocyte', 'Astrocyte', 'OPC') ~ 'Tumor',
```

```
                                                CellType %in% c('TAM',␣
↪'Lymphocyte') ~ 'Immune',
                                                CellType %in% c('Endothel',␣
↪'Pericyte', 'Excitatory neuron', 'Inhibitory neuron') ~ 'Stromal',
                                                TRUE ~ 'Others'))
```

4.5 lympho annotation

[ ]:
```
obj.GBM.lympho <- obj.GBM %>% subset(mCT %in% c('Lymphoid'))

DefaultAssay(obj.GBM.lympho) <- "RNA"
obj.GBM.lympho <- obj.GBM.lympho %>%
                  NormalizeData(normalization.method = "LogNormalize", scale.
↪factor = 10000, verbose = F) %>%
                  FindVariableFeatures(selection.method = "vst", nfeatures =␣
↪1000, verbose = F) %>%
                  ScaleData(vars.to.regress = c("nCount_RNA"), verbose = F) %>%
                  RunPCA(verbose = F) %>%
                  RunHarmony(group.by.vars = "SampleID", plot_convergence =␣
↪TRUE)

## cluster
nPC <- min(PC_selection_harmony(obj.GBM.lympho)$PCselect)
obj.GBM.lympho <- obj.GBM.lympho %>%
                  RunUMAP(reduction = "harmony", dims = 1:nPC, umap.method =␣
↪"uwot") %>%
                  RunTSNE(reduction = "harmony", dims = 1:nPC) %>%
                  FindNeighbors(reduction = "harmony", dims = 1:nPC) %>%
                  FindClusters(resolution=0.1) %>% FindClusters(resolution=0.2)␣
↪%>% FindClusters(resolution=0.3)
colnames(obj.GBM.lympho@meta.data) <- gsub("RNA_snn_res.0.","r0", colnames(obj.
↪GBM.lympho@meta.data))
```

[ ]:
```
marker_list <- c('PTPRC', 'LYN', 'HLA-DQB1',
                 'CD3D', 'CD3E', 'CD3G', 'CD4', 'CTLA4', 'FOXP3',
                 'CD8A', 'CD8B', 'GZMB', 'GNLY', 'PRF1', 'NKG7', 'KLRD1',␣
↪'NCAM1',
                 'CD79A', 'MS4A1')

options(repr.plot.height = 5, repr.plot.width = 4)
VlnPlot(obj.GBM.lympho, group.by = 'r02',
        features = marker_list,
        pt.size = 0, cols = pal_igv("default")(51),
        slot = 'data',assay = 'RNA', raster=FALSE, stack = TRUE, flip = TRUE)+
theme(legend.position = 'none')
```

[ ]:
```
## cluster DEG
Idents(obj.GBM.lympho) <- factor(obj.GBM.lympho$r02, levels = 3:0)
cluster_deg <- FindAllMarkers(obj.GBM.lympho, assay = "RNA", slot = "data",
                              logfc.threshold = 0.25, min.pct = 0.1, test.use =␣
 ↪"wilcox")
rownames(cluster_deg) <- NULL
```

[ ]:
```
## assign annotation
obj.GBM.lympho@meta.data <- obj.GBM.lympho@meta.data %>%
                            mutate(dCT = ext_list(CellType)) %>%
                            mutate(mCT = case_when(r02 == "0" ~ "CD4T",
                                                   r02 == "1" ~ "Microglia",
                                                   r02 == "2" ~ "NK",
                                                   r02 == "3" ~ "Treg",
                                                   TRUE ~ 'Others'))
table(obj.GBM.lympho$mCT, obj.GBM.lympho$CellType, obj.GBM.lympho$gCT, obj.GBM.
 ↪lympho$oCT)
```

[ ]:
```
obj.GBM.other <- subset(obj.GBM, mCT != 'Lymphoid')
obj.GBM.lympho@meta.data <- obj.GBM.lympho@meta.data[, names(obj.GBM.other@meta.
 ↪data)]
obj.GBM <- merge(obj.GBM.lympho, obj.GBM.other)
```

4.6 filter out unannotated cells

[ ]:
```
table(obj.GBM$mCT, useNA="ifany")
obj.GBM <- obj.GBM %>% subset(mCT %in% c("NA", "Others") == FALSE)
table(obj.GBM$mCT, useNA="ifany")
```

5 run UMAP

[ ]:
```
obj.GBM <- obj.GBM %>%
           NormalizeData(normalization.method = "LogNormalize", scale.factor =␣
 ↪10000, verbose = F) %>%
           FindVariableFeatures(selection.method = "vst", nfeatures = 3000,␣
 ↪verbose = F) %>%
           ScaleData(vars.to.regress = c("nCount_RNA"), verbose = F) %>%
           RunPCA(verbose = F)  %>%
           RunHarmony(group.by.vars = "SampleID", plot_convergence = TRUE)

## cluster
nPC <- min(PC_selection_harmony(obj.GBM)$PCselect)
obj.GBM <- obj.GBM %>%
           RunUMAP(reduction = "harmony", dims = 1:nPC, umap.method = "uwot")␣
 ↪%>%
           RunTSNE(reduction = "harmony", dims = 1:nPC)
```

[ ]:
```
options(repr.plot.height = 5, repr.plot.width = 25)
select <- 'umap'
DimPlot_scCustom(obj.GBM, pt.size = .1, group.by = "gCT", reduction = select,␣
 ↪label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.GBM, pt.size = .1, group.by = "mCT", reduction = select,␣
 ↪label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.GBM, pt.size = .1, group.by = "oCT", reduction = select,␣
 ↪label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))
```

[ ]:
```
obj.GBM@meta.data[,c('SampleType', 'Sex', 'SampleID')] %>% .[!duplicated(.
 ↪$SampleID),] %$% table(.$SampleType, .$Sex)
table(obj.GBM$SampleType)
```

6 save

[ ]:
```
saveRDS(obj.GBM, 'obj.GBM.use.rds')
```