TumorCell_integration

December 25, 2025

1 load data

[ ]:
```
objList <- list.files('/project/sex_cancer/data/data_zenodo', pattern = 'obj',␣
 ↪full.names = TRUE)
objList
length(objList)
```

[ ]:
```
seuratList <- lapply(objList, function(x){readRDS(x)})
names(seuratList) <- objList %>% gsub('/project/sex_cancer/data/data_zenodo/obj.
 ↪', '', .) %>% gsub('.rds', '', .)
```

2 extract intersect genes

[ ]:
```
geneList <- lapply(seuratList, function(x){rownames(x)})
geneList_all <- geneList %>% ext_list() %>% unique()
length(geneList_all) ## 65526 genes
geneList_freq13 <- geneList %>% unlist %>% table() %>% as.data.frame() %>%␣
 ↪subset(Freq == 13) %>% .[,1] %>% ext_list()
length(geneList_freq13) ## 13412 genes
```

3 filter tumor cells

[ ]:
```
seuratList_name <- names(seuratList)
seuratList_name
```

[ ]:
```
## filter cells & genes (retain only tumor cells)
seuratList <- lapply(seuratList, function(obj){
                        obj %>% subset(gCT == 'Tumor') %>% subset(SampleType ==␣
 ↪'tumor') %>% subset(feature = geneList_freq13)
                      })
names(seuratList) <- seuratList_name
seuratList

lapply(seuratList, function(x){ncol(x)}) %>% do.call(sum, .)
seurat_TumorCell <- merge(seuratList[[1]], seuratList[-1])
```

4 Diet Tumor component

downsample samples with > 1000 cells to 1000 cells

[ ]:
```
obj <- seurat_TumorCell
sampleList <- unique(obj$SampleID)
length(sampleList)
```

[ ]:
```
sampleDiet <- lapply(sampleList, function(x){
                    sampleMeta <- obj@meta.data %>% subset(SampleID == x)
                    Ncell <- nrow(sampleMeta)
                    if(Ncell > 1000){
                        sampleMeta <- sampleMeta[sample(Ncell, 1000),]
                    }
                    return(sampleMeta)
                }) %>% do.call(rbind, .)
dim(sampleDiet)
sampleDiet %>% head(n = 2)

## filter
obj.diet <- obj %>% subset(cells = rownames(sampleDiet))
```

5 sample integration

[ ]:
```
obj.harmony <-  obj.diet %>%
                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(npcs = 50, verbose = T) %>%
                RunHarmony(group.by.vars = c("Cohort", "mCT"),
                           theta = c(.1, 2), ## Diversity clustering penalty␣
 ↪parameter. Specify for each variable in group.by.vars. Default theta=2.␣
 ↪theta=0 does not encourage any diversity. Larger values of theta result in␣
 ↪more diverse clusters.
                           lambda = c(.1, 1), ## Ridge regression penalty␣
 ↪parameter. Specify for each variable in group.by.vars. Default lambda=1.␣
 ↪Lambda must be strictly positive. Smaller values result in more aggressive␣
 ↪correction.
                           sigma = .1,
                           max.iter.harmony = 50,
                           plot_convergence = F)
## cluster
nPC <- min(PC_selection_harmony(obj.harmony)$PCselect)
obj.harmony <- obj.harmony %>%
```

```
                RunUMAP(reduction = "harmony", dims = 1:nPC, umap.method =␣
 ↪"uwot")
colnames(obj.harmony@meta.data)
```

[ ]:
```
options(repr.plot.height = 5, repr.plot.width = 30)
select <- 'umap'
DimPlot_scCustom(obj.harmony, pt.size = .1, group.by = "gCT", reduction =␣
 ↪select, label = F, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.harmony, pt.size = .1, group.by = "mCT", reduction =␣
 ↪select, label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.harmony, pt.size = .1, group.by = "Cohort", reduction =␣
 ↪select, label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.harmony, pt.size = 1, group.by = "Sex", label = TRUE,␣
 ↪label.size = 4, colors_use = pal_igv("default")(51))
```

6 integration effect evaluation

[ ]:
```
embed <- obj.harmony@reductions$umap@cell.embeddings
meta <- obj.harmony@meta.data
res <- compute_lisi(embed, meta, c('Cohort'), perplexity = 100)
str(res)
data.frame(LISI_mean = mean(res$Cohort), LISI_median = median(res$Cohort))
```

[ ]:
```
options(repr.plot.height = 2, repr.plot.width = 4)
ggplot(res, aes(x = Cohort, y = 1))+
geom_density_ridges(scale = 2, alpha = 0.8, rel_min_height= 0, fill =␣
 ↪"#bba1cd", color = "#696969",
                    quantile_lines= TRUE, quantiles= 0.5, vline_size=0.3,␣
 ↪vline_linetype= "dashed")+
scale_x_continuous(breaks = c(1, 5, 9), labels = c(1, 5, 9))+
labs(x = "LISI", y = "", title = "Tumor component | integration quality")+
coord_cartesian(expand=TRUE, clip = "off")+
ridge_theme
```

7 save

[ ]:
```
DefaultAssay(obj.harmony) <- "RNA"
obj <- DietSeurat(obj.harmony, counts = TRUE, data = TRUE, scale.data = FALSE,␣
 ↪features = rownames(obj.harmony), assays = "RNA", dimreducs = c("pca",␣
 ↪"umap"), misc = FALSE)
saveRDS(obj, "obj.TumorCell.diet.rds")
```

8 function for use

[ ]:
```
ridge_theme <- theme(panel.background = element_rect(fill = NA),
                     panel.grid.major.y = element_blank(),
                     panel.grid.major.x = element_blank(),
                     plot.margin = margin(t=10,r=10,b=5,l=5,unit = "mm"),
                     legend.position = "none",
                     plot.title = element_text(size = 8, color = "#696969",␣
 ↪family = "Arial", face = "bold", vjust = 2, hjust = 0.5),
                     axis.ticks.length.x = unit(0.3, "mm"),
                     axis.ticks.y = element_blank(),
                     axis.line.x = element_line(colour = "grey40",size = 0.5),
                     axis.line.y = element_blank(),
                     axis.text.x = element_text(size = 6, family = "Arial",␣
 ↪color = "black"),
                     axis.text.y = element_blank(),
                     axis.title = element_text(size = 7, family = "Arial", face␣
 ↪= "bold", color = "black", hjust = 0.5))
```