FL_Han2022_process

December 25, 2025

1 load data

follicluar lymphoma - B-cell lymphoma
raw counts not provided in the original article

[ ]:
```
adata <- read_h5ad('/project/sex_cancer/data/FL_Han2022/
 ↪19dc3c7f-1318-401d-b885-e046fd96a13e.h5ad')
exp <- adata$X %>% t()
meta <- adata$obs

## create seurat object
obj.FL <- CreateSeuratObject(counts = exp,
                             meta.data = meta,
                             project = "FL_Han2022", assay = "RNA",
                             min.cells = 0, min.features = 0)

## add UMAP embedding
obj.FL[['umap']] <- CreateDimReducObject(embeddings = adata$obsm$X_UMAP  %>%␣
 ↪{colnames(.) <- c('UMAP_1', 'UMAP_2'); .},
                                          key = 'UMAP_', assay = 'RNA')
obj.FL@meta.data <- obj.FL@meta.data %>% mutate_if(~!is.numeric(.), ext_list)
```

[ ]:
```
## add sample info
info <- read_xlsx('/project/sex_cancer/data/FL_Han2022/HanPatientInfo.
 ↪xlsx',sheet = 2)
info$Sample_id <- unique(obj.FL@meta.data$sample_id)

meta <- obj.FL@meta.data %>% rownames_to_column('barcode') %>%
        merge(info, by = 'sample_id',all = TRUE)
names(meta)[26]='Sex'
meta=meta[,-c(39:43)]
```

2 modify meta.data

[ ]:
```
obj.FL@meta.data <- obj.FL@meta.data %>%
                    dplyr::rename(c('DonorID' = 'SampleID', 'Tissue' =␣
 ↪'tissue', 'Disease' = 'disease')) %>%
                    dplyr::rename(c('SampleID' = 'sample_id', 'percent.mt' =␣
 ↪'Ratio.MT')) %>%
                    transform(SampleType = ifelse(Disease == 'follicular␣
 ↪lymphoma', 'tumor', 'normal')) %>%
                    transform(Sex = ifelse(Sex == 'Female', 'F', 'M')) %>%
                    dplyr::rename(c('Chemistry' = 'assay'))  %>%
                    transform(Cohort = 'FL_Han2022')
```

3 filte sample

[ ]:
```
obj.FL <- obj.FL %>% subset(Prior.lines.of.therapy == 0 | Disease == 'Normal')␣
 ↪## discard treated samples
obj.FL
```

4 trans ENSG to symbol

[ ]:
```
hg19 <- get_map('/project/sex_cancer/data/Homo_sapiens.GRCh37.87.gtf') %>%
        dplyr::select(c('gene_id', 'gene_name'))

## map ENSEMBL ID to gene symbol
exp <- GetAssayData(obj.FL, assay = 'RNA', slot = 'counts')
gene_keep <- intersect(rownames(exp), hg19$gene_id)
exp <- exp[gene_keep,]
## merge symbol info
exp <- hg19 %>% column_to_rownames('gene_id') %>% .[gene_keep,] %>% cbind(.,␣
 ↪as_matrix(exp))
colnames(exp)[1] <- 'gene_symbol'
exp[1:6, 1:6]

## aggregate symbols with the same ENSG
exp <- exp %>% as.data.frame()
exp <- exp %>% mutate_at(vars(2:ncol(exp)), ~ as.numeric(.))
exp <- aggregate(. ~ gene_symbol, data = exp, FUN = max)
exp <- exp %>% column_to_rownames('gene_symbol') %>% .[,colnames(obj.FL)]
```

[ ]:
```
obj.FL <- CreateSeuratObject(counts = exp, meta.data = obj.FL@meta.
 ↪data[colnames(obj.FL),], min.cells = 0, min.features = 0, project =␣
 ↪'FL_Han2022')
obj.FL
```

5 cell type annotation

5.1 assign oCT

[ ]:
```
obj.FL@meta.data <- obj.FL@meta.data %>%
                    mutate(oCT = cell_type) %>%
                    mutate(dCT = case_when(oCT %in% c('Malignant') ~␣
 ↪'Malignant',
                                           oCT %in% c('CD4_CTL', 'CD4_Naive',␣
 ↪'CD4_Tfh', 'CD4Proliferating') ~ 'CD4T',
                                           oCT %in% c('CD4_Treg') ~ 'Treg',
                                           oCT %in% c('CD8_Naive', 'CD8_Exh',␣
 ↪'CD8_Eff', 'CD8Proliferating') ~ 'CD8T',
                                           oCT %in% c('Tcell_other') ~␣
 ↪'T_others',
                                           oCT %in% c('NKT') ~ 'NKT',
                                           oCT %in% c('NormalB') ~ 'B',
                                           oCT %in% c('NormalPlasma') ~␣
 ↪'Plasma',
                                           oCT %in% c('Myeloid') ~ 'Myeloid',
                                           oCT %in% c('pDC') ~ 'pDC',
                                           oCT %in% c('fDC') ~ 'fDC',
                                           oCT %in% c('Erythrocyte') ~␣
 ↪'Erythrocyte',
                                           TRUE ~ 'Others'))
```

5.2 check annotation

check cell type annotation provided in the original research via COSG

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

obj <- obj.FL
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))
})
```

5.3 assign mCT

[ ]:
```
obj.FL@meta.data <- obj.FL@meta.data %>%
                    mutate(mCT = case_when(dCT %in% c('CD4T', 'CD8T',␣
 ↪'T_others', 'NKT', 'Treg', 'Malignant', 'B', 'Plasma') ~ 'Lymphoid',
                                           dCT %in% c('Myeloid', 'pDC', 'fDC')␣
 ↪~ 'Myeloid',
                                           dCT %in% c('Erythrocyte') ~␣
 ↪'Erythroid',
                                           TRUE ~ 'Others'))
```

5.4 assign gCT

[ ]:
```
obj.FL@meta.data <- obj.FL@meta.data %>%
                    mutate(gCT = case_when(dCT %in% c('B', 'Malignant',␣
 ↪'Plasma') ~ 'Tumor',
                                           dCT %in% c('CD4T', 'Treg',␣
 ↪'T_others', 'CD8T', 'NKT', 'Myeloid', 'pDC', 'fDC', 'Erythrocyte') ~␣
 ↪'Immune',
                                           TRUE ~ 'Others'))
```

[ ]:
```
options(repr.plot.height = 5, repr.plot.width = 30)
select <- 'umap'
```

```
DimPlot_scCustom(obj.FL, pt.size = .1, group.by = "gCT", reduction = select,␣
  ↪label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.FL, pt.size = .1, group.by = "mCT", reduction = select,␣
  ↪label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))|
DimPlot_scCustom(obj.FL, pt.size = .1, group.by = "oCT", reduction = select,␣
  ↪label = TRUE, label.size = 4, colors_use = pal_igv("default")(51))
```

5.5 save

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