COSG_annotation_curation

December 25, 2025

1 extract marker gene sets for cell type annotation

[ ]:
```r
marker_annotation <- openxlsx::read.xlsx("/project/sex_cancer/data/Table3.
↪xlsx", startRow = 2)[,-1] %>% lapply(na.omit)
names(marker_annotation) <- names(marker_annotation) %>% strsplit2(split = "\\.
↪") %>% .[,1]
marker_annotation
save(marker_annotation, "marker_annotation.rds")
```

2 check annotation

[ ]:
```r
obj_list <- list.files("/project/sex_cancer/data/", pattern = "^obj.+_", full.
↪names = T)
obj_list
length(obj_list)
```

[ ]:
```r
## top 50 cell type-specific marker (COSG based)
library(COSG)
marker_oCT_COSG50 <- lapply(obj_list, function(x){
                                print(x)
                                obj <- readRDS(x)
                                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))
                                    return(marker_oCT)
                                })
```

2.1 check expression of marker genes in each dataset

[ ]:
```
for (i in 1:13){
    print(i)
    oCT_marker <- marker_oCT_COSG50[[i]]
    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))
    })
}
```