TumorCell_merge

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 extract tumor cells

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

```
[ ]: 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])
```

[ ]:

```
seurat_TumorCell <- seurat_TumorCell %>%
                    NormalizeData(normalization.method = "LogNormalize", scale.
 ↪factor = 10000, verbose = F)
```

4 malignancy score calculation

[ ]:

```
obj <- seurat_TumorCell %>% SplitObject(split.by = "Cohort")
obj
```

[ ]:

```
# code source: https://github.com/czythu/scCancer/blob/master/vignettes/
 ↪malignantCellIden.Rmd
scCancer_malignancy <- function(object){
                            model.path <- paste0(system.file("txt", package =␣
 ↪"scCancer"), "/sc_xgboost.model")
                            genes.path <- paste0(system.file("txt", package =␣
 ↪"scCancer"), "/genes-scRNA-tcga-sorted.txt")
                            model.ref <- xgb.load(model.path)

                            features <- as.list(read.table(genes.path))[[1]]
                            testdata <- t(as.matrix(object@assays$RNA@scale.
 ↪data))

                            temp <- matrix(data = 0, nrow = nrow(testdata),␣
 ↪ncol = length(features), dimnames = list(rownames(testdata), features))
                            current.features <- colnames(testdata)
                            for(j in 1:length(features)){
                                if(features[j] %in% current.features){
                                    temp[,j] <- testdata[, features[j]]
                                }
                            }
                            testdata <- temp

                            # Prediction
                            testdata <- xgb.DMatrix(testdata)
                            predict.label <- predict(model.ref, testdata)
                            predict.score <- predict.label
                            predict.label[which(predict.label > 0.5)] <-␣
 ↪"Malignant"
                            predict.label[which(predict.label <= 0.5)] <-␣
 ↪"nonMalignant"
                            table(predict.label)

                            # Visualization
                            object$malignant.label <- predict.label
                            object$malignant.score <- predict.score
                            return(object)
                        }
```

[ ]:

```
## run malignancy calculation
obj <- lapply(obj, function(x){scCancer_malignancy(x)})
obj <- merge(obj[[1]], obj[-1])
obj@meta.data <- obj@meta.data %>% dplyr::rename(c('Malignant_label' =␣
 ↪'malignant.label', 'Malignant_score' = 'malignant.score'))
obj
obj@meta.data %>% head(n = 2)
```

5 save

[ ]:

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