## ##Mapping query dataset to the reference## ##

library(SeuratDisk)

library(anndata)

library(SummarizedExperiment)

library(TabulaMurisData)

library(patchwork)

library(scater)

library(ExperimentHub)

library(SingleCellExperiment)

library(iSEE)

library(scran)

library(GEOquery)

####Azimuth####

###Save Seurat objects in smaller size format

experiment.diet <- DietSeurat(experiment)

head(experiment.diet[[]])

SaveH5Seurat(experiment.diet, "experiment.diet", overwrite = TURE)

####Load Seurat objects####

#example from https://figshare.com/articles/dataset/Expression\_of\_97\_surface\_markers\_and\_462\_mRNAs\_in\_31586\_cells\_from\_15\_leukemic\_human\_bone\_marrow/14780127

bone.marrow.data <- readRDS("AMLs\_Scano\_projected.rds")

head(bone.marrow.data[[]])

DimPlot(bone.marrow.data, label = FALSE , cols=colbig, group.by = "ct", repel = FALSE) + ggtitle("RNA Clustering Reference")

####Transform AnnData files into Seurat objects - inter-compatibiltiy between scnapy (Python) and Seurat (R)

Convert("tabula-muris-senis-droplet-official-raw-obj.h5ad", dest = "h5Seurat", overwrite = TRUE)

reference1 <- LoadH5Seurat("tabula-muris-senis-droplet-official-raw-obj.h5seurat")

head(reference1[[]])

####TabulaMuris dataset####

eh <- ExperimentHub()

query(eh, "TabulaMurisData")

##Droplet data

droplet <- eh[["EH1617"]]

droplet <- TabulaMurisDroplet()

colnames(colData(droplet))

##Droplet data all

droplet <- scran::computeSumFactors(droplet)

droplet <- scater::logNormCounts(droplet)

droplet <- scater::runPCA(droplet)

droplet <- scater::runUMAP(droplet)

droplet <- scater::runTSNE(droplet)

if (require(iSEE)) {

iSEE(droplet)

}

droplet.seurat <- as.Seurat(droplet, count = "counts", data = "logcounts")

head(droplet.seurat[[]])

##Droplet data spleen

spleen <- subset(x = droplet.seurat, subset = tissue == "Spleen")

head(spleen[[]])

DimPlot(spleen, reduction = "PCA", group.by = "cell\_ontology\_class") + NoLegend()

Idents(object = spleen) <- "mouse\_id"

####Datasets via Gene Expression Omnibus

gse <- getGEO("GSE168158", GSEMatrix = TRUE)

show(gse)

gse1 <- as.data.frame(exprs(gse[[1]]))

meta.data <- read.delim(file.choose("GSE139833\_series\_matrix.txt"))

WT.1.GEX.raw <- read.delim(file.choose("GSM5130034\_WT\_1\_singlecell\_gex\_raw\_counts.txt"))

WT.1.GEX.norm <- read.delim(file.choose("GSM5130034\_WT\_1\_singlecell\_gex\_norm\_counts.txt"))

WT.1.seurat <- CreateSeuratObject(counts = WT.1.GEX.raw, meta.data = meta.data)

WT.1.seurat.norm <- CreateSeuratObject(counts = WT.1.GEX.norm, meta.data = meta.data)

head(WT.1.seurat.norm[[]])

### 1) Mouse\_BM reference dataset

load(file.choose("NicheData10x.rda"), verbose = TRUE)

head(NicheData10x[[]])

colnames(NicheData10x[[]])

load(file.choose("NicheMarkers10x.rda"), verbose = TRUE)

head(NicheMarkers10x)

####Transfer anchors function####

#Load query datasets

experiment <- LoadH5Seurat("SeuratProject.h5Seurat")

DefaultAssay(experiment) <- "RNA"

p1=DimPlot(experiment, label = TRUE,cols=colbig,reduction = "rna.umap", label.size = 2.5) + NoLegend()

p2=DimPlot(experiment, label = TRUE,cols=colbig,reduction = "adt.umap", label.size = 2.5) + NoLegend()

p3=DimPlot(experiment, label = TRUE,cols=colbig, reduction = "wnn.umap", label.size = 2.5) + NoLegend()

FeaturePlot(reference, features = "Cd19", reduction = "umap")

#Select reference dataset

reference <- NicheData10x

DimPlot(NicheData10x, label = TRUE)

Idents(reference) <- levels(NicheData10x)

levels(NicheData10x)

NicheData10x@meta.data

query <- experiment

head(reference[[]])

head(query[[]])

DefaultAssay(reference) <- "RNA"

DefaultAssay(query) <- "RNA"

#Perform same normalisation step as for query

reference = SCTransform(reference, verbose = TRUE)

reference[["SCT"]]

reference <- RunPCA(reference, verbose = FALSE, features = VariableFeatures(object = reference))

pca\_variance <- reference@reductions$pca@stdev^2

plot(pca\_variance/sum(pca\_variance),

ylab="Proportion of variance explained",

xlab="Principal component")

abline(h = 0.01) #23

reference <- FindNeighbors(reference, dims = 1:23)

reference <- FindClusters(reference, resolution = 2.0, verbose = FALSE) #2.0for the resolution

clustree(reference, prefix = "SCT\_snn\_res.") + theme(legend.position="bottom")

reference <- RunUMAP(reference, dims = 1:23)

DimPlot(reference, label = TRUE, cols=colbig) + ggtitle("RNA Clustering")

#Or

reference <- NormalizeData(reference)

reference <- FindVariableFeatures(reference)

reference <- ScaleData(reference)

query <- NormalizeData(query)

query <- FindVariableFeatures(query)

query <- ScaleData(query)

#Find anchors

anchors <- FindTransferAnchors(reference = reference, query = query, dims = 1:9)

?FindTransferAnchors

anchors <- FindTransferAnchors(reference = reference, query = query, normalization.method = "SCT", dims = 1:18)

#Transfer labels

predictions <- TransferData(anchorset = anchors,refdata = Idents(reference))

?TransferData()

query <- AddMetaData(object = query, metadata = predictions)

DimPlot(query, label = TRUE, group.by = "predicted.id", repel = TRUE, reduction = "wnn.umap") + ggtitle("RNA Clustering query")