setwd('/home/medie/Rfile/chenchen/CCLM/2.rda')

options(stringsAsFactors = F)

#加载R包

library(Seurat)

library(dplyr)

library(ggplot2)

library(magrittr)

library(gtools)

library(stringr)

library(Matrix)

library(tidyverse)

library(patchwork)

library(data.table)

library(RColorBrewer)

library(ggpubr)

library(data.table)

library(RColorBrewer)

library(ggpubr)

library(SingleR)

library(celldex)

library(BiocParallel)

library(viridis)

library(pheatmap)

library(viridisLite)

load('CCLM.RData')

CCLM<-Seurat\_object

setwd('/home/medie/Rfile/chenchen/CCLM/3.jiangju')

library(Seurat)

library(dplyr)

library(ggplot2)

library(magrittr)

library(gtools)

library(stringr)

library(Matrix)

library(tidyverse)

library(patchwork)

library(data.table)

library(RColorBrewer)

library(ggpubr)

CCLM <- NormalizeData(CCLM, normalization.method = "LogNormalize", scale.factor = 10000)

CCLM <- FindVariableFeatures(CCLM,

selection.method = "vst",

nfeatures = 2000,

mean.cutoff=c(0.0125,3),

dispersion.cutoff =c(1.5,Inf))

top20 <- head(VariableFeatures(CCLM), 20)

plot1 <- VariableFeaturePlot(CCLM)

plot2 <- LabelPoints(plot = plot1, points = top20, repel = TRUE, size=3.0)

feat\_20 <- CombinePlots(plots = list(plot1, plot2),legend="bottom")

feat\_20

ggsave(filename = 'feat\_20.pdf',plot = feat\_20,he=10,wi=15)

#ScaleData

scale.genes <- rownames(CCLM)

CCLM <- ScaleData(CCLM, features = scale.genes)

CCLM <- RunPCA(CCLM, features = VariableFeatures(CCLM))

dimplot1 <- DimPlot(CCLM, reduction = "pca")

dimplot1

ggsave(filename = 'dimplot2.pdf',plot = dimplot1,he=10,wi=15)

elbowplot1 <- ElbowPlot(CCLM, ndims=50, reduction="pca")

elbowplot1

sc\_pca <- dimplot1+elbowplot1

sc\_pca

ggsave(filename = 'sc\_pca.pdf',plot = sc\_pca,he=10,wi=15)

top20 <- VizDimLoadings(CCLM, dims = 1:2, nfeatures = 20, reduction = "pca")

top20

ggsave(filename = 'top20.pdf',plot = top20,he=10,wi=15)

DimHeatmap(CCLM, dims = 1:20, cells = 500, balanced = TRUE)

CCLM$orig.ident

Dims <- 30

Resolution <- 0.1

CCLM <- FindNeighbors(object = CCLM, dims = 1:Dims)

CCLM <- FindClusters(object = CCLM, resolution = Resolution)

allcolour=c("#FF34B3", "#BC8F8F", "#20B2AA", "#00F5FF", "#FFA500", "#ADFF2F", "#FF6A6A", "#7FFFD4","#AB82FF", "#90EE90", "#00CD00", "#008B8B", "#6495ED", "#FFC1C1", "#CD5C5C", "#8B008B", "#FF3030", "#7CFC00", "#000000", "#708090", "#DC143C","#0000FF","#20B2AA","#FFA500","#9370DB","#98FB98","#F08080","#1E90FF","#7CFC00","#FFFF00","#808000","#FF00FF","#CCCCFF","#000000","#7B68EE","#9400D3","#A0522D","#800080","#D2B48C","#D2691E","#87CEEB","#40E0D0","#5F9EA0","#FF1493","#0000CD","#008B8B","#FFE4B5","#8A2BE2","#228B22","#E9967A","#4682B4","#32CD32","#F0E68C","#FFFFE0","#EE82EE","#FF6347","#6A5ACD","#9932CC","#8B008B","#8B4513","#DEB887")

allcolour=c("#E64B35FF", "#4DBBD5FF", "#00A087FF", "#3C5488FF",

"#FFA500", "#8491B4FF", "#FA8072", "#DC0000FF",

"#7E6148FF", "#B09C85FF","#800000","#91D1C2FF")

length(table(sce@active.ident))

mycolor = allcolour[1:length(table(sce@active.ident))]

### UMAP

CCLM <- RunUMAP(CCLM, dims=1:Dims, reduction="pca")

CCLM <- RunTSNE(CCLM,

dims=1:Dims,

reduction="pca")

sc\_umap = DimPlot(CCLM,#cols=allcolour,

reduction="umap",

#reduction="tsne",

label = "F",

pt.size = 0.2,

label.size = 3) +

theme(axis.line = element\_line(size=0.1, colour = "black"),

axis.ticks = element\_blank()

)

sc\_umap

ggsave('sc\_umapcluster.pdf',sc\_umap,he=7,wi=7)

sc\_umap\_sample = DimPlot(CCLM,#cols=allcolour,

group.by='tissueunique',

#reduction="umap",

reduction="tsne",

label = "T",

pt.size = 1.0,

label.size = 0) +

theme(axis.line = element\_line(size=0.1, colour = "black"),

axis.ticks = element\_blank()

) +ggtitle('')

sc\_umap\_sample

ggsave('sc\_umap\_sample.pdf',sc\_umap\_sample,he=7,wi=7)

sc\_umap\_group = DimPlot(CCLM,group.by='Group1',

#reduction="umap",

reduction="tsne",

label = "T",

pt.size = 0.2,

label.size = 0) +

theme(axis.line = element\_line(size=0.1, colour = "black"),

axis.ticks = element\_blank()

) +ggtitle('')

sc\_umap\_group

ggsave('sc\_umap\_group.pdf',sc\_umap\_group,he=7,wi=7)

setwd('/home/medie/Rfile/chenchen/CCLM/4.zhushi')

Logfc = 0.5

Minpct = 0.35

DefaultAssay(CCLM) <- "RNA"

CCLM.markers <- FindAllMarkers(object = CCLM,logfc.threshold = Logfc,

min.pct = Minpct,only.pos = T)

CCLM.markers["pct.diff"]=CCLM.markers$pct.1-CCLM.markers$pct.2

CCLM.markers <- CCLM.markers[CCLM.markers$p\_val\_adj<0.05,]

length(unique(CCLM.markers$gene))

head(CCLM.markers)

diff.wilcox = FindAllMarkers(CCLM)

CCLM.markers = diff.wilcox %>% select(gene, everything()) %>% subset(p\_val<0.05)

top10 = CCLM.markers %>% group\_by(cluster) %>% top\_n(n = 10, wt = avg\_log2FC)

write.table(CCLM.markers,'scRNA\_marker\_gene1.txt',quote = F,row.names = F,sep='\t')

Top5 <- CCLM.markers %>%

group\_by(cluster) %>%

slice\_max(n =5, order\_by = avg\_log2FC)

write.table(Top5,'Top5.txt',quote = F,row.names = F,sep='\t')

Top5 <- unique(Top5$gene)

sc\_marker\_dotplot <- DotPlot(object = CCLM,

features = Top5,

cols=c("blue", "red"),

scale = T)+

RotatedAxis()+ ggtitle("Top 5 Marker Genes")+

theme(plot.title = element\_text(hjust = 0.5))

sc\_marker\_dotplot

ggsave(filename = 'sc\_marker\_dotplot.pdf',

plot = sc\_marker\_dotplot,

height = 9,width = 25)

library(viridisLite)

sc\_marker\_heatmap<- DoHeatmap(object = CCLM,

features = Top5,

#group.colors = mycolor,

label = F) +

ggtitle("Top 5 Marker Genes") +

theme(plot.title = element\_text(hjust = 0.5))

sc\_marker\_heatmap

ggsave(filename = 'sc\_marker\_heatmap5.pdf',

plot = sc\_marker\_heatmap,

width = 12,height = 24)

library(SingleR)

library(celldex)

library(BiocParallel)

library(Seurat)

library(viridis)

library(pheatmap)

testdata <- GetAssayData(CCLM, slot="data")

refdata <- HumanPrimaryCellAtlasData()

clusters <- CCLM@meta.data$seurat\_clusters

cellpred <- SingleR(test = testdata, ref = refdata, labels = refdata$label.fine,

method = "cluster", clusters = clusters,

assay.type.test = "logcounts", assay.type.ref = "logcounts")

celltype = data.frame(ClusterID=rownames(cellpred), celltype=cellpred$labels, stringsAsFactors = F)

write.table(celltype,'celltype.txt',quote = F,sep = '\t',row.names = F)

CCLM@meta.data$celltype = "NA"

for(i in 1:nrow(celltype)){

CCLM@meta.data[which(CCLM@meta.data$seurat\_clusters == celltype$ClusterID[i]),'celltype'] <- celltype$celltype[i]}

unique(CCLM$celltype)

library(paletteer)

paletteer\_d("ggsci::nrc\_npg")

allcolor<-paletteer\_d("ggsci::nrc\_npg")

allcolor

umap\_celltype1<-DimPlot(CCLM,group.by = 'celltype',

cols = allcolor,

reduction="umap",

#reduction="tsne",

label = "F",

pt.size = 0.5,

label.size = 4)

umap\_celltype1

ggsave('umap\_celltype1111.pdf',umap\_celltype1,he=7,wi=12)

colour=c("#7E0021FF","#F97B72FF")

umap\_group1<-DimPlot(sce,group.by = 'Group1',

cols = colour,

#reduction="umap",

reduction="tsne",

label = "F",

pt.size = 0.2,

label.size = 4)

umap\_group1

tsnem <- umap\_celltype1|umap\_group1

tsnem

table(CCLM$orig.ident)

table(CCLM$celltype)

table(CCLM$celltype, CCLM$orig.ident)

Cellratio <- prop.table(table(CCLM$celltype, CCLM$orig.ident), margin = 2)

Cellratio

write.table(Cellratio,'Cellratio.txt',quote = T,row.names = T,sep='\t')

Cellratio <- as.data.frame(Cellratio)

Cellratio

colourCount = length(unique(Cellratio$Var1))

library(ggplot2)

ratio <- ggplot(data=Cellratio,aes(x=Var2,y=Freq,fill=Var1)) +

geom\_bar(stat="identity",position="stack") +

labs(x="Sample",y="Ratio") +

theme()

ratio

ratio1<-ratio+scale\_fill\_manual(values = c("#E64B35FF", "#4DBBD5FF", "#00A087FF", "#3C5488FF",

"#F39B7FFF", "#8491B4FF", "#91D1C2FF", "#DC0000FF",

"#7E6148FF", "#B09C85FF"))

ratio1

ggsave('ratio.pdf',ratio,he=7,wi=10)

save(CCLM,file = 'CCLM1.RData')

#load('sce1.RData')

setwd('/home/medie/Rfile/chenchen/CCLM/5.subcell')

rm()

Cells.sub\_NK <- subset(CCLM@meta.data,

celltype == "NK\_cell")

Cells.sub\_NK <- subset(CCLM@meta.data,

celltype == "NK\_cell"|“T\_cell”)

CCLMnk\_sce <- subset(CCLM, cells=row.names(Cells.sub\_NK))

nk\_sce = sce[,sce@meta.data$seurat\_clusters %in% c(12)]`

save(CCLMnk\_sce,file = 'CCLMnk\_sce.RData')

load('CCLMnk\_sce.RData')

CCLMnk\_sce <- FindVariableFeatures(CCLMnk\_sce, selection.method = "vst", nfeatures = 2000)

scale.genes <- rownames(CCLMnk\_sce)

CCLMnk\_sce <- ScaleData(CCLMnk\_sce, features = scale.genes)

CCLMnk\_sce <- RunPCA(CCLMnk\_sce, features = VariableFeatures(CCLMnk\_sce))

ElbowPlot(CCLMnk\_sce, ndims=20, reduction="pca")

pc.num=1:15

CCLMnk\_sce <- FindNeighbors(CCLMnk\_sce, dims = pc.num)

CCLMnk\_sce <- FindClusters(CCLMnk\_sce, resolution = 0.1)

table(CCLMnk\_sce@meta.data$seurat\_clusters)

metadata <- CCLMnk\_sce@meta.data

cell\_cluster <- data.frame(cell\_ID=rownames(metadata), cluster\_ID=metadata$seurat\_clusters)

write.csv(cell\_cluster,'cell\_cluster.csv',row.names = F)

CCLMnk\_sce <- RunTSNE(CCLMnk\_sce,

dims = pc.num,

reduction="pca")

embed\_tsne <- Embeddings(CCLMnk\_sce, 'tsne')

write.csv(embed\_tsne,'embed\_tsne.csv')

CCLMnk\_sce <- RunUMAP(CCLMnk\_sce, dims=pc.num, reduction="pca")

library(paletteer)

paletteer\_d("ggthemes::gdoc")

allcolor<-paletteer\_d("ggthemes::gdoc")

allcolor<-paletteer\_d("ggsci::nrc\_npg")

allcolor

plot1 = DimPlot(CCLMnk\_sce, reduction = "tsne", cols = allcolor,

pt.size = 0.5,

label.size = 4)

plot1

diff.wilcox = FindAllMarkers(CCLMnk\_sce)

all.markers = diff.wilcox %>% select(gene, everything()) %>% subset(p\_val<0.05)

top50 = all.markers %>% group\_by(cluster) %>% top\_n(n = 50, wt = avg\_log2FC)

Top10 <- all.markers %>%

group\_by(cluster) %>%

slice\_max(n =10, order\_by = avg\_log2FC)

write.table(top50,'Top50.txt',quote = F,row.names = F,sep='\t')

Top10 <- unique(Top10$gene)

NK\_marker\_dotplot <- DotPlot(object = CCLMnk\_sce,

features = Top10,

cols=c("blue", "red"),

scale = T)+

RotatedAxis()+ ggtitle("Top 5 Marker Genes")+

theme(plot.title = element\_text(hjust = 0.5))

NK\_marker\_dotplot <- DotPlot(object = CCLMnk\_sce,

features = Top10,

group.by='seurat\_clusters')

NK\_marker\_dotplot <- NK\_marker\_dotplot+scale\_color\_gradientn(values= seq(0,1,0.2), colours = c("#134B85","#2264A3", "#337CB7", "#4B94C4",

"#6BACD0","#8FC2DD", "#B1D4E7", "#CDE3EE",

"#E3EDF2","#F2EFEE", "#F9E8DD", "#FBD7C4",

"#F8BFA4","#F1A385", "#E48268", "#D56050",

"#C33D3D","#AC202F", "#8C0D25", "#67001F"))+theme\_bw()

NK\_marker\_dotplot

ggsave(filename = 'NK\_marker\_dotplot.pdf',

plot = sc\_marker\_dotplot,

height = 9,width = 25)

p <- DotPlot(CCLMnk\_sce, features=c('CD2', "GZMK", "NME8", "TEP1", "TRBC1","TTC38",

'CD69', "GRAP2", "SOCS1", "CCL4", "CCND2","APOL6",

'PTGER2', "NCR3", "OSM", "TNFSF14", "APOBEC3G","CDK6"), group.by='seurat\_clusters')

p <- p + scale\_color\_gradientn(values= seq(0,1,0.2), colours = c("#134B85","#2264A3", "#337CB7", "#4B94C4",

"#6BACD0","#8FC2DD", "#B1D4E7", "#CDE3EE",

"#E3EDF2","#F2EFEE", "#F9E8DD", "#FBD7C4",

"#F8BFA4","#F1A385", "#E48268", "#D56050",

"#C33D3D","#AC202F", "#8C0D25", "#67001F"))+theme\_bw()

p

library(viridisLite)

NK\_marker\_heatmap<- DoHeatmap(object = CCLMnk\_sce,

features = Top10,

group.colors = allcolor,

label = F) +

ggtitle("Top 5 Marker Genes") +

theme(plot.title = element\_text(hjust = 0.5))+

scale\_fill\_gradientn(colors = c("#377EB8", "white", "#E41A1C"))

NK\_marker\_heatmap

ggsave(filename = 'NK\_marker\_heatmap5.pdf',

plot = sc\_marker\_heatmap,

width = 12,height = 24)

write.table(top50,'top50.txt',quote = F,row.names = F,sep='\t')

write.table(all.markers,'nksce\_marker2\_gene.txt',quote = F,row.names = F,sep='\t')

write.csv(all.markers,'nksce\_marker2\_gene.csv',quote = F,row.names = F,sep='\t')

mycolor<-paletteer\_d("nbapalettes::cavaliers\_retro")

bfreaname.NK <- CCLMnk\_sce

bfreaname.NK

new.cluster.id <- c("R", "R", "A", "R", "R", "A", "O", "O")

names(new.cluster.id) <- levels(CCLMnk\_sce)

CCLMnk\_sce <- RenameIdents(CCLMnk\_sce, new.cluster.id)

plot4<-DimPlot(CCLMnk\_sce,

group.by = 'Sample',

split.by = 'tissueunique'

#reduction="umap",

#cols = mycolor,

reduction="tsne",

label = "F",

pt.size = 2.0,

label.size = 4)

plot4

table(CCLMnk\_sce$orig.ident)

table(CCLMnk\_sce$seurat\_clusters)

table(CCLMnk\_sce$Group)

table(CCLMnk\_sce$seurat\_clusters, CCLMnk\_sce$orig.ident)

Cellratio <- prop.table(table(CCLMnk\_sce$seurat\_clusters, CCLMnk\_sce$orig.ident), margin = 2)

Cellratio

write.table(Cellratio,'nkratio.txt',quote = T,row.names = T,sep='\t')

Cellratio <- as.data.frame(Cellratio)

colourCount = length(unique(Cellratio$Var1))

setwd('/home/medie/Rfile/chenchen/6.nishixu')

library(monocle)

rm()

a<-paletteer\_d("palettesForR::Browns")

a

select\_genes <- c('CD8A')

p1 <- VlnPlot(CCLMnk\_sce, features = select\_genes, pt.size=0, ncol=2)

p1

ggsave("cell\_identify/selectgenes\_VlnPlot.png", p1, width=6 ,height=8)

mycolor <- c('WhiteSmoke', '#8B4513FF')

p2 <- FeaturePlot(CCLMnk\_sce, features = select\_genes, reduction = "umap", label=T, ncol=2,cols = mycolor)

p2

ggsave("cell\_identify/selectgenes\_FeaturePlot.png", p2, width=8 ,height=12)

p3=p1|p2

ggsave("cell\_identify/selectgenes.png", p3, width=10 ,height=8)

set.seed(321)

sample(1:59529,30000,replace=F)->index

CCLMnk\_sce2<-CCLMnk\_sce[,index]

library(monocle)

library(clusterProfiler)

library(org.Hs.eg.db)

rm()

Logfc = 0.5

Minpct = 0.35

DefaultAssay(CCLMnk\_sce2) <- "RNA"

CCLMnk\_sce2.markers <- FindAllMarkers(object = CCLMnk\_sce2,logfc.threshold = Logfc,

min.pct = Minpct,only.pos = T)

CCLMnk\_sce2.markers["pct.diff"]=CCLMnk\_sce2.markers$pct.1-CCLMnk\_sce2.markers$pct.2

CCLMnk\_sce2.markers <- CCLMnk\_sce2.markers[CCLMnk\_sce2.markers$p\_val\_adj<0.05,]

length(unique(CCLMnk\_sce2.markers$gene))

head(CCLMnk\_sce2.markers)

write.csv(CCLMnk\_sce2.markers,'CCLMnk\_sce2\_marker\_gene.csv',quote = F,row.names = F,sep='\t')

setwd('/home/medie/Rfile/chenchen/CCLM/6.nishixu')

load('CCLMnk\_sce.RData')

library(clusterProfiler)

library(org.Hs.eg.db)

rm()

Logfc = 0.5

Minpct = 0.35

DefaultAssay(CCLMnk\_sce) <- "RNA"

CCLMnk\_sce.markers <- FindAllMarkers(object = CCLMnk\_sce,logfc.threshold = Logfc,

min.pct = Minpct,only.pos = T)

CCLMnk\_sce.markers["pct.diff"]=CCLMnk\_sce.markers$pct.1-CCLMnk\_sce.markers$pct.2

CCLMnk\_sce.markers <- CCLMnk\_sce.markers[CCLMnk\_sce.markers$p\_val\_adj<0.05,]

length(unique(CCLMnk\_sce.markers$gene))

head(CCLMnk\_sce.markers)

write.csv(CCLMnk\_sce.markers,'CCLMnk\_sce\_marker2\_gene.csv',quote = F,row.names = F,sep='\t')

dir.create("pseudotime")

data <- as(as.matrix(CCLMnk\_sce2@assays$RNA@counts), 'sparseMatrix')

pd <- new('AnnotatedDataFrame', data = CCLMnk\_sce2@meta.data)

fData <- data.frame(gene\_short\_name = row.names(data), row.names = row.names(data))

fd <- new('AnnotatedDataFrame', data = fData)

mycds <- newCellDataSet(data,

phenoData = pd,

featureData = fd,

expressionFamily = negbinomial.size())

mycds <- estimateSizeFactors(mycds)

mycds <- estimateDispersions(mycds, cores=4, relative\_expr = TRUE)

var.genes <- VariableFeatures(CCLMnk\_sce)

mycds <- setOrderingFilter(mycds, var.genes)

p2 <- plot\_ordering\_genes(mycds)

p2

disp\_table <- dispersionTable(mycds)

disp.genes <- subset(disp\_table, mean\_expression >= 0.1 & dispersion\_empirical >= 1 \* dispersion\_fit)$gene\_id

mycds <- setOrderingFilter(mycds, disp.genes)

p3 <- plot\_ordering\_genes(mycds)

p3

diff.genes <- read.csv('CCLMnk\_sce2\_marker\_gene.csv')

diff.genes

diff.genes <- subset(diff.genes,p\_val\_adj<0.001)$gene

mycds <- setOrderingFilter(mycds, diff.genes)

p1 <- plot\_ordering\_genes(mycds)

p1

ggsave("p1.png", plot = plot1, width = 8, height = 7)

p1|p2|p3

mycds <- reduceDimension(mycds, max\_components = 2, method = 'louvain')

mycds <- orderCells(mycds)

State <- plot\_cell\_trajectory(mycds, color\_by = "State")

State

ggsave("pseudotime/State.pdf", plot = State, width = 6, height = 5)

Cluster <- plot\_cell\_trajectory(mycds, color\_by = "Group")

Cluster

ggsave("pseudotime/Cluster.pdf", plot = Cluster, width = 6, height = 5)

Logfc = 0.5

Minpct = 0.35

DefaultAssay(CCLMnk\_sce2) <- "RNA"

CCLMnk\_sce2.markers <- FindAllMarkers(object = CCLMnk\_sce2,logfc.threshold = Logfc,

min.pct = Minpct,only.pos = T)

nk\_sce.markers["pct.diff"]=nk\_sce.markers$pct.1-sce.markers$pct.2

nk\_sce.markers <- nk\_sce.markers[nk\_sce.markers$p\_val\_adj<0.05,]

length(unique(nk\_sce.markers$gene))

head(nk\_sce.markers)

write.csv(nk\_sce.markers,'scRNA\_marker2\_gene.csv',quote = F,row.names = F,sep='\t')

dir.create("pseudotime")

data <- as(as.matrix(nk\_sce@assays$RNA@counts), 'sparseMatrix')

pd <- new('AnnotatedDataFrame', data = nk\_sce@meta.data)

fData <- data.frame(gene\_short\_name = row.names(data), row.names = row.names(data))

fd <- new('AnnotatedDataFrame', data = fData)

mycds <- newCellDataSet(data,

phenoData = pd,

featureData = fd,

expressionFamily = negbinomial.size())

mycds <- estimateSizeFactors(mycds)

mycds <- estimateDispersions(mycds, cores=4, relative\_expr = TRUE)

var.genes <- VariableFeatures(nk\_sce)

mycds <- setOrderingFilter(mycds, var.genes)

p2 <- plot\_ordering\_genes(mycds)

p2

disp\_table <- dispersionTable(mycds)

disp.genes <- subset(disp\_table, mean\_expression >= 0.1 & dispersion\_empirical >= 1 \* dispersion\_fit)$gene\_id

mycds <- setOrderingFilter(mycds, disp.genes)

p3 <- plot\_ordering\_genes(mycds)

p3

diff.genes <- read.csv('nksce\_marker2\_gene.csv')

diff.genes

diff.genes <- subset(diff.genes,p\_val\_adj<0.01)$gene

mycds <- setOrderingFilter(mycds, diff.genes)

p1 <- plot\_ordering\_genes(mycds)

p1

ggsave("p1.png", plot = plot1, width = 8, height = 7)

p1|p2|p3

mycds <- reduceDimension(mycds, max\_components = 2, method = 'DDRTree')

mycds <- orderCells(mycds)

State <- plot\_cell\_trajectory(mycds, color\_by = "State")

State

ggsave("pseudotime/State.pdf", plot = plot1, width = 6, height = 5)

Cluster <- plot\_cell\_trajectory(mycds, color\_by = "Group")

Cluster

ggsave("pseudotime/Cluster.pdf", plot = plot2, width = 6, height = 5)

Pseudotime <- plot\_cell\_trajectory(mycds, color\_by = "Pseudotime")

Pseudotime

ggsave("pseudotime/Pseudotime.pdf", plot = plot3, width = 6, height = 5)

plotc <- plot1|plot2|plot3

plotc

ggsave("pseudotime/Combination.pdf", plot = plotc, width = 10, height = 3.5)

ggsave("pseudotime/Combination.png", plot = plotc, width = 10, height = 3.5)

write.csv(pData(mycds), "pseudotime/pseudotime.csv")

p1 <- plot\_cell\_trajectory(mycds, color\_by = "State") + facet\_wrap(~State, nrow = 1)

p2 <- plot\_cell\_trajectory(mycds, color\_by = "seurat\_clusters") + facet\_wrap(~seurat\_clusters, nrow = 1)

plotc <- p1/p2

plotc

ggsave("pseudotime/trajectory\_facet.png", plot = plotc, width = 6, height = 5)

diff.genes <- read.csv('scRNA\_marker2\_gene.csv')

sig\_diff.genes <- subset(diff.genes,p\_val\_adj<0.0001&abs(avg\_log2FC)>0.75)$gene

sig\_diff.genes <- unique(as.character(sig\_diff.genes))

diff\_test <- differentialGeneTest(mycds[sig\_diff.genes,], cores = 1,

fullModelFormulaStr = "~sm.ns(Pseudotime)")

sig\_gene\_names <- row.names(subset(diff\_test, qval < 0.01))

p1 = plot\_pseudotime\_heatmap(mycds[sig\_gene\_names,], num\_clusters=3,

show\_rownames=T, return\_heatmap=T)

ggsave("pseudotime/pseudotime\_heatmap1.png", plot = p1, width = 5, height = 8)

disp\_table <- dispersionTable(mycds)

disp.genes <- subset(disp\_table, mean\_expression >= 0.5&dispersion\_empirical >= 1\*dispersion\_fit)

disp.genes <- as.character(disp.genes$gene\_id)

diff\_test <- differentialGeneTest(mycds[disp.genes,], cores = 4,

fullModelFormulaStr = "~sm.ns(Pseudotime)")

sig\_gene\_names <- row.names(subset(diff\_test, qval < 1e-04))

p2 = plot\_pseudotime\_heatmap(mycds[sig\_gene\_names,], num\_clusters=5,

show\_rownames=T, return\_heatmap=T)

ggsave("pseudotime/pseudotime\_heatmap2.png", plot = p2, width = 5, height = 10)

disp\_table <- dispersionTable(mycds)

disp.genes <- subset(disp\_table, mean\_expression >= 0.5&dispersion\_empirical >= 1\*dispersion\_fit)

disp.genes <- as.character(disp.genes$gene\_id)

mycds\_sub <- mycds[disp.genes,]

plot\_cell\_trajectory(mycds\_sub, color\_by = "State")

beam\_res <- BEAM(mycds\_sub, branch\_point = 1, cores = 8)

beam\_res <- beam\_res[order(beam\_res$qval),]

beam\_res <- beam\_res[,c("gene\_short\_name", "pval", "qval")]

mycds\_sub\_beam <- mycds\_sub[row.names(subset(beam\_res, qval < 1e-4)),]

plot\_genes\_branched\_heatmap(mycds\_sub\_beam, branch\_point = 1, num\_clusters = 3, show\_rownames = T)

rm()

library(Seurat)

library(ggplot2)

library(patchwork)

library(dplyr)

library(cowplot)

setwd('/home/medie/Rfile/chenchen/space/ST/ST-colon1')

name='colon1'

expr <- "/home/medie/Rfile/chenchen/space/ST/ST-colon1/filtered\_feature\_bc\_matrix.h5"

expr.mydata <- Seurat::Read10X\_h5(filename = expr )

mydataC1 <- Seurat::CreateSeuratObject(counts = expr.mydata, project = 'colon1', assay = 'Spatial')

mydataC1$slice <- 1

mydataC1$region <- 'colon1'

imgpath <- "/home/medie/Rfile/chenchen/space/ST/ST-colon1"

img <- Seurat::Read10X\_Image(image.dir = imgpath)

Seurat::DefaultAssay(object = img) <- 'Spatial'

img <- img[colnames(x = mydata)]

mydataC1[['image']] <- img

mydataC1

C1plot1 <- VlnPlot(mydataC1, features = "nCount\_Spatial", pt.size = 0.5) + NoLegend()

C1plot1

C1plot2 <- SpatialFeaturePlot(mydataC1, features = "nCount\_Spatial") + theme(legend.position = "right")

C1plot2

C1plot3 <- VlnPlot(mydataC1, features = "nFeature\_Spatial", pt.size = 0.5) + NoLegend()

C1plot3

C1plot4 <- SpatialFeaturePlot(mydataC1, features = "nFeature\_Spatial") + theme(legend.position = "right")

C1plot4

mydataC1[["percent.mt"]] <- PercentageFeatureSet(mydataC1, pattern = "^MT[-]")

C1plot5 <- VlnPlot(mydataC1, features = "percent.mt", pt.size = 0.5) + NoLegend()

C1plot5

C1plot6 <- SpatialFeaturePlot(mydataC1, features = "percent.mt") + theme(legend.position = "right")

C1plot6

mydataC1 <- subset(mydataC1, subset = nFeature\_Spatial > 200 & nFeature\_Spatial <7500 & nCount\_Spatial > 1000 & nCount\_Spatial < 60000 & percent.mt < 25)

mydataC1

mydataC1 <- SCTransform(mydataC1, assay = "Spatial", verbose = FALSE)

top10 <- head(VariableFeatures(mydataC1),10)

top10

mydataC1 <- RunPCA(mydataC1, assay = "SCT", verbose = FALSE)

mydataC1 <- FindNeighbors(mydataC1, reduction = "pca", dims = 1:30)

mydataC1 <- FindClusters(mydataC1, verbose = FALSE)

mydataC1 <- RunUMAP(mydataC1, reduction = "pca", dims = 1:30)

mydataC1 <- RunTSNE(mydataC1, reduction = "pca",dims = 1:30)

C1p9 <- DimPlot(mydataC1, reduction = "tsne", label = TRUE)

C1p9

C1p10 <- SpatialDimPlot(mydataC1, label = TRUE, label.size = 3)

C1p10

data.markersC1 <- FindAllMarkers(mydataC1, only.pos = FALSE, min.pct = 0.25, logfc.threshold = 0.25,test.use = "wilcox")

data.markersC1

write.table(data.markersC1,'data.markersC1.txt',quote = F,row.names = F,sep='\t')

topgeneC1<-data.markersC1 %>% group\_by(cluster) %>% top\_n(n = 5, wt = avg\_log2FC)

topgeneC1

DoHeatmap(mydataC1, features = topgeneC1$gene,size = 2) + NoLegend()

library(SingleR)

library(celldex)

library(BiocParallel)

library(Seurat)

library(viridis)

library(pheatmap)

ref.se <- HumanPrimaryCellAtlasData()

expdata =mydata2[["Spatial"]]@data

clusters <- mydata2@meta.data$seurat\_clusters

clusters

anno.cluster.main<- SingleR(test = expdata, ref = ref.se, labels = ref.se$label.fine, method= "cluster", clusters = clusters)

DimPlot(mydata2,reduction = "tsne" ,group.by="celltype")

SpatialDimPlot(mydata2, label = F, label.size = 3,group.by="celltype")

allcolour=c("#FF34B3", "#BC8F8F", "#20B2AA", "#00F5FF", "#FFA500", "#ADFF2F", "#FF6A6A", "#7FFFD4",

"#AB82FF", "#90EE90", "#00CD00", "#008B8B", "#6495ED", "#FFC1C1", "#CD5C5C", "#8B008B",

"#FF3030", "#7CFC00", "#000000", "#708090", "#DC143C","#0000FF","#20B2AA","#FFA500","#9370DB","#98FB98","#F08080","#1E90FF","#7CFC00","#FFFF00",

"#808000","#FF00FF","#CCCCFF","#000000","#7B68EE","#9400D3","#A0522D","#800080","#D2B48C","#D2691E",

"#87CEEB","#40E0D0","#5F9EA0","#FF1493","#0000CD","#008B8B","#FFE4B5","#8A2BE2","#228B22","#E9967A",

"#4682B4","#32CD32","#F0E68C","#FFFFE0","#EE82EE","#FF6347","#6A5ACD","#9932CC","#8B008B","#8B4513",

"#DEB887")

celltype = data.frame(ClusterID=rownames(anno.cluster.main), celltype=anno.cluster.main$labels, stringsAsFactors = F)

celltype

mydata2[['celltype']]<-

celltype$celltype[match(Idents(mydata2), celltype$ClusterID)]

DimPlot(mydata2,reduction = "tsne" ,group.by="SingleR.labels",cols = allcolour)

head(mydata2@meta.data)

SpatialDimPlot(mydata2, label = F, label.size = 3,group.by="SingleR.labels")

anno.cell.main=SingleR(test=expdata , ref = ref.se, labels = ref.se$label.main)

mydata2[["SingleR.labels"]]<- as.character(anno.cell.main$labels)

DimPlot(mydata2,reduction = "tsne" ,group.by="SingleR.labels")

NKA <- list(c("APOBEC3G","APOL6","CCND2","CCL4","CD69","CDK6","CSF2",

"DPP4","FASLG","GPR171","GPR18","GRAP2","IFNG","KIR2DL4",

"KIR2DS4","LTA","LTB","NCR3","OSM","PTGER2","SOCS1","TNFSF14"))

NKA <- list(c("TPSB2","TPSAB1","CPA3","LTC4S","CD69","CLU","HPGDS",

"CTSG","CD63","ANXA1"))

mydata3 <- AddModuleScore(mydata2,features = NKA, ctrl = 100, name = "NKA")

head(mydata3@meta.data)

colnames(mydata3@meta.data)

colnames(mydata3@meta.data)[7] <- "NKA"

VlnPlot(mydata3,features = "NKA")

plot1<-SpatialFeaturePlot(mydata3, features = "NKA")

plot1

NKR <- list(c("AZU1","BPI","CAMP","CD160","CDHR1","CEACAM8",

"DEFA4","ELANE","GFI1","GZMK","KLRC4","MGAM","MS4A3",

"NME8","PLEKHF1","TEP1","TTC38","ZNF135"))

NKR <- list(c("AZU1","CDHR1","DEFA4","ELANE","PLEKHF1","TEP1","TTC38","ZNF135"))

mydata4 <- AddModuleScore(mydata2,features =NKR, ctrl = 100, name = "NKR")

head(mydata4@meta.data)

colnames(mydata4@meta.data)

colnames(mydata4@meta.data)[13] <- "NKR"

VlnPlot(mydata4,features = "NKR")

plot2<-SpatialFeaturePlot(mydata4, features = "NKR")

wrap\_plots(plot1, plot2)