```
rm(list = ls())
suppressPackageStartupMessages(library(Seurat))
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(clustree))
suppressPackageStartupMessages(library(cowplot))
suppressPackageStartupMessages(library(dplyr))
suppressPackageStartupMessages(library(data.table))
C51 <- Read10X_h5("../data/GSM4475048_C51_filtered_feature_bc_matrix.h5",use.names = TRUE, unique.features = TRUE)
C52 <- Read10X_h5("../data/GSM4475049_C52_filtered_feature_bc_matrix.h5",use.names = TRUE, unique.features = TRUE)
C100 <- Read10X_h5("../data/GSM4475050_C100_filtered_feature_bc_matrix.h5",use.names = TRUE, unique.features = TRUE)
C01 <- Read10X("../data/01/")
C02 <- Read10X("../data/02/")
C03 <- Read10X("../data/03/")
seob.list <- list(C51,C52,C100,C01,C02,C03)
names(seob.list) <- c("C51","C52","C100","C01","C02","C03")
sce.list <- list()
for (i in 1:length(seob.list)) {
  print(i)
  pro <- names(seob.list)[i]
  print(pro)
  sce=CreateSeuratObject(counts = seob.list[[i]],project = pro)
  dim(sce)
  sce.list[[pro]] = sce
  print( Sys.time() )
}
dim(sce.list$C51)
dim(sce.list$C52)
dim(sce.list$C100)
dim(sce.list$C01)
dim(sce.list$C02)
dim(sce.list$C03)
save(sce.list,file = "../data/sce.list.Rdata")
load("../data/sce.list.Rdata")
sce.all <- merge(x = sce.list[[1]],
                 y = sce.list[-1],
                 add.cell.ids = names(sce.list))
sce.all$group <- ifelse(sce.all$orig.ident %in% c("C01","C02","C03"),"treat","normal")
save(sce.all,file = "../data/all_sample.Rdata")
mito_genes=rownames(sce.all)[grep("^MT-", rownames(sce.all))]
mito_genes
sce.all=PercentageFeatureSet(sce.all, "^MT-", col.name = "percent_mito")
fivenum(sce.all@meta.data$percent_mito)
ribo_genes=rownames(sce.all)[grep("^Rp[sl]", rownames(sce.all),ignore.case = T)]
ribo_genes
sce.all=PercentageFeatureSet(sce.all, "^RP[SL]", col.name = "percent_ribo")
fivenum(sce.all@meta.data$percent_ribo)
hb_genes <- rownames(sce.all)[grep("^Hb[^(p)]", rownames(sce.all),ignore.case = T)]
hb_genes
sce.all=PercentageFeatureSet(sce.all, "^HB[^(P)]", col.name = "percent_hb")
fivenum(sce.all@meta.data$percent_hb)
feats <- c("nFeature_RNA", "nCount_RNA", "percent_mito", "percent_ribo", "percent_hb")
feats <- c("nFeature_RNA", "nCount_RNA")
p1=VlnPlot(sce.all, group.by = "orig.ident", features = feats, pt.size = 0.01, ncol = 2) +
  NoLegend()
p1
ggsave(filename="../picture/1_QC/Vlnplot1.png",plot=p1)
feats <- c("percent_mito", "percent_ribo", "percent_hb")
p2=VlnPlot(sce.all, group.by = "orig.ident",
```

```
features = feats, pt.size = 0.01, ncol = 3, same.y.lims=T) +
  scale_y_continuous(breaks=seq(0, 100, 5)) +
  NoLegend()
p2
ggsave(filename="../picture/1_QC/Vlnplot2.png",plot=p2)
p3=FeatureScatter(sce.all, "nCount_RNA", "nFeature_RNA", group.by = "orig.ident", pt.size = 0.5)
p3
ggsave(filename="../picture/1_QC/Scatterplot.pdf",plot=p3)
selected_c <- WhichCells(sce.all, expression = nFeature_RNA > 300)
selected_f <- rownames(sce.all)[Matrix::rowSums(sce.all@assays$RNA@counts > 0 ) > 3]
sce.all.filt <- subset(sce.all, features = selected_f, cells = selected_c)
dim(sce.all)
dim(sce.all.filt)
table(sce.all$orig.ident)
table(sce.all.filt$orig.ident)
dim(sce.all.filt)
selected_mito <- WhichCells(sce.all.filt, expression = percent_mito < 15)
selected_ribo <- WhichCells(sce.all.filt, expression = percent_ribo > 1)
selected_hb <- WhichCells(sce.all.filt, expression = percent_hb < 5)
length(selected_hb)
length(selected_ribo)
length(selected_mito)
sce.all.filt <- subset(sce.all.filt, cells = selected_mito)
sce.all.filt <- subset(sce.all.filt, cells = selected_ribo)
sce.all.filt <- subset(sce.all.filt, cells = selected_hb)
dim(sce.all.filt)
table(sce.all$orig.ident)
table(sce.all.filt$orig.ident)
feats <- c("nFeature_RNA", "nCount_RNA")
p1_filtered=VlnPlot(sce.all.filt, group.by = "orig.ident", features = feats, pt.size = 0.1, ncol = 2) +
  NoLegend()
p1_filtered
ggsave(filename="../picture/1_QC/Vlnplot1_filtered.pdf",plot=p1_filtered)

feats <- c("percent_mito", "percent_ribo", "percent_hb")
p2_filtered=VlnPlot(sce.all.filt, group.by = "orig.ident", features = feats, pt.size = 0.1, ncol = 3) +
  NoLegend()
p2_filtered
ggsave(filename="../picture/1_QC/Vlnplot2_filtered.pdf",plot=p2_filtered)

p3_filtered=FeatureScatter(sce.all.filt, "nCount_RNA", "nFeature_RNA", group.by = "orig.ident", pt.size = 0.5)
p3_filtered
ggsave(filename="../picture/1_QC/Scatterplot2.pdf",plot=p3_filtered)

pp1 <- p1|p1_filtered
ggsave(filename="../picture/1_QC/Vlnplot1_before_after.pdf",plot=pp1,units = "cm",width = 35,height = 18)
pp2 <- p2|p2_filtered
ggsave(filename="../picture/1_QC/Vlnplot2_before_after.pdf",plot=pp2,units = "cm",width = 35,height = 18)
pp3 <- p3|p3_filtered
ggsave(filename="Scatterplot_before_after.pdf",plot=pp3,units = "cm",width = 20,height = 18)
save(sce.all.filt,file = "../subdata/sce.all.filt.Rdata")
rm(list=ls())
load("../subdata/sce.all.filt.Rdata")
dim(sce.all.filt)
table(sce.all.filt$orig.ident)
```

```
sce <- sce.all.filt
sce <- NormalizeData(sce)
sce = ScaleData(sce,
                features = rownames(sce),
              )
sce <- FindVariableFeatures(sce, nfeatures = 3000)
sce <- RunPCA(sce, npcs = 30)
sce <- RunTSNE(sce, dims = 1:30)
sce <- RunUMAP(sce, dims = 1:30)
colnames(sce@meta.data)
p1.compare=plot_grid(ncol = 3,
                     DimPlot(sce, reduction = "pca", group.by = "group")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce, reduction = "tsne", group.by = "group")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(sce, reduction = "umap", group.by = "group")+NoAxes()+ggtitle("UMAP"),

                     DimPlot(sce, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(sce, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p1.compare
ggsave(plot=p1.compare,filename="../picture/2_PCA_UMAP_TSNE/Before_inter_1.pdf")
DimPlot(sce,
        reduction = "umap",
        group.by = "orig.ident",
        label = F)
ggsave("../picture/2_PCA_UMAP_TSNE/Dimplot_before_1.pdf",units = "cm", width = 20,height = 18)
sce.all=sce
sce.all
sce.all.list <- SplitObject(sce.all, split.by = "orig.ident")
sce.all.list
for (i in 1:length(sce.all.list)) {
  print(i)
  sce.all.list[[i]] <- NormalizeData(sce.all.list[[i]], verbose = FALSE)
  sce.all.list[[i]] <- FindVariableFeatures(sce.all.list[[i]],
selection.method = "vst",

nfeatures = 2000, verbose = FALSE)
}
alldata.anchors <- FindIntegrationAnchors(object.list = sce.all.list, dims = 1:30,

reduction = "cca")
sce.all.int <- IntegrateData(anchorset = alldata.anchors, dims = 1:30, new.assay.name = "CCA")
names(sce.all.int@assays)
sce.all.int@active.assay
sce.all.int=ScaleData(sce.all.int)
sce.all.int=RunPCA(sce.all.int, npcs = 30)
sce.all.int=RunTSNE(sce.all.int, dims = 1:30)
sce.all.int=RunUMAP(sce.all.int, dims = 1:30)
names(sce.all.int@reductions)
p2.compare=plot_grid(ncol = 3,
                     DimPlot(sce.all.int, reduction = "pca", group.by = "group")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce.all.int, reduction = "tsne", group.by = "group")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(sce.all.int, reduction = "umap", group.by = "group")+NoAxes()+ggtitle("UMAP"),

                     DimPlot(sce.all.int, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce.all.int,
```

```
reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                    DimPlot(sce.all.int, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p2.compare
ggsave(plot=p2.compare,filename="../picture/2_PCA_UMAP_TSNE/Before_inter_CCA_2.pdf")

p5.compare=plot_grid(ncol = 3,
                    DimPlot(sce, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                    DimPlot(sce, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                    DimPlot(sce, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP"),

                    DimPlot(sce.all.int, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                    DimPlot(sce.all.int, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                    DimPlot(sce.all.int, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p5.compare
ggsave(plot=p5.compare,filename="../picture/2_PCA_UMAP_TSNE/umap_by_har_before_after.pdf",units = "cm", width = 40,height = 18)
save(sce.all.int,file = "../subdata/sce.all.int_by_cca.Rdata")
rm(list = ls())
load("../subdata/sce.all.int_by_cca.Rdata")
sce <- sce.all.int
sce <- FindNeighbors(sce, dims = 1:30)
for (res in c(0.01, 0.05, 0.1, 0.2, 0.3, 0.5,0.8,1))
{
  print(res)
  sce <- FindClusters(sce, graph.name = "CCA_snn", resolution = res, algorithm = 1)
}
ls <- apply(sce@meta.data[,grep("CCA_snn_res",colnames(sce@meta.data))],2,table)
ls
save(sce,file = '../subdata/first_sce_by_CCA.Rdata')

cluster_umap <- plot_grid(ncol = 4,
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.01", label = T) & NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.05", label = T)& NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.1", label = T) & NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.2", label = T)& NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.3", label = T)& NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.5", label = T) & NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.8", label = T) & NoAxes(),
                    DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.1", label = T) & NoAxes()
)
ggsave(cluster_umap, filename = "../picture/2_PCA_UMAP_TSNE/cluster_umap_all_res_by_CCA.pdf", width = 16, height = 8)

saveRDS(sce, file = "../subdata/sce_all.Rds")
library(harmony)
```

```
seuratObj <- RunHarmony(sce, "orig.ident",project.dim = F)
names(seuratObj@reductions)
seuratObj <- RunUMAP(seuratObj, dims = 1:15,
                     reduction = "harmony")
DimPlot(seuratObj,reduction = "umap",label=T )
ggsave("../picture/2_PCA_UMAP_TSNE/umap.pdf")
p3.compare=plot_grid(ncol = 3,
                     DimPlot(seuratObj, reduction = "pca", group.by = "group")+NoAxes()+ggtitle("PCA"),
                     DimPlot(seuratObj, reduction = "tsne", group.by = "group")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(seuratObj, reduction = "umap", group.by = "group")+NoAxes()+ggtitle("UMAP"),

                     DimPlot(seuratObj, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                     DimPlot(seuratObj, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(seuratObj, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p3.compare
ggsave(plot=p3.compare,filename="../picture/2_PCA_UMAP_TSNE/umap_by_har_before_after.pdf",units = "cm", width = 40,height = 18)

sce_har=seuratObj
save(sce_har,file = "../subdata/sce_har_by_harmony.Rdata")
p4.compare=plot_grid(ncol = 3,
                     DimPlot(sce.all.int, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce.all.int, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(sce.all.int, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP"),

                     DimPlot(seuratObj, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                     DimPlot(seuratObj, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(seuratObj, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p4.compare
ggsave(plot=p4.compare,filename="../picture/2_PCA_UMAP_TSNE/umap_by_har_cca.pdf",units = "cm", width = 40,height = 18)
cluster_umap <- plot_grid(ncol = 4,
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.01", label = T) & NoAxes(),
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.05", label = T)& NoAxes(),
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.1", label = T) & NoAxes(),
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.2", label = T)& NoAxes(),
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.3", label = T)& NoAxes(),
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.5", label = T) & NoAxes(),
                          DimPlot(sce, reduction = "umap", group.by = "CCA_snn_res.0.8", label = T) & NoAxes(),
                          DimPlot(sce,
```

```
reduction = "umap", group.by = "CCA_snn_res.1", label = T) & NoAxes()
)

rm(list = ls())
suppressPackageStartupMessages(library(Seurat))
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(clustree))
suppressPackageStartupMessages(library(cowplot))
suppressPackageStartupMessages(library(dplyr))
suppressPackageStartupMessages(library(data.table))

load('../subdata/sce.all.int.cca.Rdata')
sce.all=sce.all.int
sce.all@active.assay
sce.all=FindNeighbors(sce.all, dims = 1:30, k.param = 60, prune.SNN = 1/15)
for (res in c(0.01, 0.05, 0.1, 0.2, 0.3, 0.5,0.8,1)) {
   sce.all=FindClusters(sce.all, graph.name = "CCA_snn", resolution = res, algorithm = 1)
}
apply(sce.all@meta.data[,grep("CCA_snn_res",colnames(sce.all@meta.data))],2,table)

p1_dim=plot_grid(ncol = 3, DimPlot(sce.all, reduction = "umap", group.by = "RNA_snn_res.0.01") +

ggtitle("louvain_0.01"), DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.0.1") +
                         ggtitle("louvain_0.1"), DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.0.2") +
                         ggtitle("louvain_0.2"))
ggsave(plot=p1_dim, filename="../picture/2_PCA_UMAP_TSNE/Dimplot_diff_resolution_low.pdf",units = "cm",width = 45,height =18 )

p1_dim=plot_grid(ncol = 2, DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.0.8") +
                         ggtitle("louvain_0.8"), DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.1") +
                         ggtitle("louvain_1"), DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.0.3") +
                         ggtitle("louvain_0.3"), DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.0.5") +
                         ggtitle("louvain_0.5"))
ggsave(plot=p1_dim, filename="../picture/2_PCA_UMAP_TSNE/Dimplot_diff_resolution_high.pdf",units = "cm",width = 40,height = 36)


p2_tree=clustree(sce.all@meta.data, prefix = "CCA_snn_res.")
ggsave(plot=p2_tree, filename="../picture/2_PCA_UMAP_TSNE/Tree_diff_resolution.pdf",units = "cm",width = 24,height =18)
sel.clust = "CCA_snn_res.0.8"
sce.all <- SetIdent(sce.all, value = sel.clust)
table(sce.all@active.ident)
saveRDS(sce.all, "../subdata/sce.all_int.rds")
rm(list=ls())
sce.all
DimPlot(sce.all, reduction = "umap", group.by = "seurat_clusters",label = T)
DimPlot(sce.all, reduction = "umap", group.by = "CCA_snn_res.0.8",label = T)
ggsave('../picture/3_celltype/umap_by_CCA_snn_res.0.8.pdf',units = "cm",width = 18,height = 18)
rm(list=ls())
library(Seurat)
library(ggplot2)
```

```
library(clustree)
library(cowplot)
library(dplyr)
load("../subdata/last.Rdata")
sce.all.list <- SplitObject(sce.all , split.by = "celltype")
sce.all.list
names(sce.all.list)
for (i in names(sce.all.list)) {
  print(i)
  epi_mat = sce.all.list[[i]]@assays$RNA@counts
  epi_phe = sce.all.list[[i]]@meta.data
  sce=CreateSeuratObject(counts = epi_mat, meta.data = epi_phe )
  sce
  table(sce@meta.data$orig.ident)
  save(sce,file = paste0("../subcluster/",i,'.Rdata'))
}
sce.all$group <- ifelse(sce.all$orig.ident %in% c("C51","C100","C52"),"control","treat")
colnames(sce.all@meta.data)
table(sce.all$seurat_clusters)
table(sce.all$orig.ident)
table(sce.all$celltype)
table(sce.all$group)
sce.all.list <- SplitObject(sce.all , split.by = "celltype")
sce.all.list
sce <- sce.all
result_DEG <- list()
for (i in 1:length(sce.all.list)) {
  # i=1
  print(i)
  name <- names(sce.all.list)[i]
  print(name)
  seob = sce.all.list[[i]]
  seob
  Idents(seob) <- seob@meta.data$group
  result_DEG[[name]]<- FindMarkers(seob,

ident.1 = "treat",
ident.2 = "control",

assay = "RNA"
  )
  VlnPlot(object = seob,
          features = head(rownames(result_DEG[[i]]),6),
          split.by = "group",
          pt.size = 0.2,
          fill.by = "feature")

ggsave(paste0("../picture/4_DEG/",name,"top12_gene_expression.pdf"),units = "cm",width = 40,height = 24)
}
for (name in names(result_DEG)) {
  result_DEG[[name]]$symbol <- rownames(result_DEG[[name]])
  result_DEG[[name]] <- result_DEG[[name]]
}
df <- plyr::ldply(result_DEG, data.frame)
write.csv(df,file = "../subdata/sample_DEG.csv")
VlnPlot(object = sce,
        features = head(rownames(result_DEG[[1]]),6),
        split.by = "orig.ident",
        pt.size = 0.2,
        fill.by = "feature")

save(sce.all,result_DEG,file = "../subdata/DEG_result.Rdata")
# load("../subdata/DEG_result.Rdata")
library(tidyr)
library(reshape2)
library(dplyr)
library(ggplot2)
sce <- sce.all
bar_data <- as.data.frame(table(sce$celltype,sce@meta.data$group))

bar_per <- bar_data %>%
```

```
  group_by(Var2) %>%
  mutate(sum(Freq)) %>%
  mutate(percent = Freq / `sum(Freq)`)
da <- bar_per %>%
  group_by(Var1) %>%
  mutate(sum(percent)) %>%
  mutate(percent/`sum(percent)`)

write.csv(da,file = "../picture/4_DEG/celltype_percent.csv")
ggplot(da, aes(x = Var1, y = percent/`sum(percent)`)) +
  geom_bar(aes(fill = Var2) , stat = "identity") + coord_flip() +theme(axis.ticks = element_line(linetype = "blank"),legend.position = "top",

panel.grid.minor = element_line(colour = NA,

linetype = "blank"), panel.background = element_rect(fill = NA),

plot.background = element_rect(colour = NA)) +labs(y = "% Relative cell source", fill = NULL)+labs(x = NULL)
ggsave("../picture/4_DEG/Relative_cell_source.pdf",units = "cm",width = 20,height = 12)
subcluster_list <- list()

colors <- c('#e41a1c','#377eb8','#4daf4a','#984ea3','#ff7f00','#ffff33','#a65628','#f781bf')
names(colors) <- unique(colnames(sce@meta.data)[grep("RNA_snn_res", colnames(sce@meta.data))])

for (name in names(sce.all.list)) {
  #name <- "DC"
  print(name)
  sce <- sce.all.list[[name]]
  seob <- CreateSeuratObject(
    sce@assays$RNA@counts,
    project = sce$orig.ident
  )
  sce <- seob
  sce <- NormalizeData(sce)
  sce = ScaleData(sce,
                  features = rownames(sce),
                  # vars.to.regress = c("nFeature_RNA", "percent_mito")
  )
  sce <- FindVariableFeatures(sce, nfeatures = 3000)
  sce <- RunPCA(sce, npcs = 30)
  sce <- RunTSNE(sce, dims = 1:30)
  sce <- RunUMAP(sce, dims = 1:30)
  sce <- FindNeighbors(sce, dims = 1:30)
  for (res in c(0.01, 0.05, 0.1, 0.2, 0.3, 0.5,0.8,1)) {
    #res=0.01
    sce <- FindClusters(sce, graph.name = "RNA_snn", resolution = res, algorithm = 1)
  }
  p2_tree <- clustree(sce@meta.data, prefix = "RNA_snn_res.")
  ggsave(p2_tree, filename = paste0("../picture/5_suncluster.tree/",name,"_cluster_tree.pdf"), width = 10, height = 8)
  subcluster_list[[name]] <- sce
}
colors <- c('#a6cee3','#1f78b4','#b2df8a','#33a02c','#fb9a99','#e31a1c','#fdbf6f','#ff7f00','#cab2d6','#6a3d9a','#ffff99','#b15928')

for (name in names(subcluster_list)) {
  #name <- "DC"
  print(name)
  sce <- subcluster_list[[name]]
  for (i in c("umap","tsne")) {
    #i="umap"
    print(i)
    p <- DimPlot(sce, reduction = "umap", group.by = "RNA_snn_res.0.2", label = T, cols = colors)+
      scale_y_continuous(expand = c(0, 0)) +
      scale_x_discrete(position = "bottom")
```

```
+
        scale_fill_manual( breaks = rev(names(colors)),values = colors)
      ggsave( p, filename = paste0("../picture/6_subcluster_by_umap/",name,"_cluster_",i,".pdf"), width = 10, height = 8)
   }
rm(list = ls())
library(Seurat)
library(ggplot2)
library(clustree)
library(cowplot)
library(dplyr)
load('../subdata/Macrophages.Rdata')
sce
sce$group <- ifelse(sce$orig.ident %in% c("C51","C100","C52"),"control","treat")
table(sce@meta.data$group)

sce <- NormalizeData(sce, normalization.method = "LogNormalize",
                     scale.factor = 1e4)
GetAssay(sce,assay = "RNA")
sce <- FindVariableFeatures(sce,

selection.method = "vst", nfeatures = 2000)
sce <- ScaleData(sce)
sce <- RunPCA(object = sce, pc.genes = VariableFeatures(sce))
DimHeatmap(sce, dims = 1:12, cells = 100, balanced = TRUE)
ElbowPlot(sce)
sce <- FindNeighbors(sce, dims = 1:15)
sce <- FindClusters(sce, resolution = 0.4)
table(sce@meta.data$RNA_snn_res.0.4)
set.seed(123)
sce <- RunTSNE(object = sce, dims = 1:15, do.fast = TRUE)
DimPlot(sce,reduction = "tsne",label=T)
sce <- RunUMAP(object = sce, dims = 1:15, do.fast = TRUE)
DimPlot(sce,reduction = "umap",label=T)
head(sce@meta.data)
DimPlot(sce,reduction = "umap",label=T,group.by = 'orig.ident')
sce <- FindClusters(sce, resolution = 0.1)
DimPlot(sce,reduction = "umap",label=T)
ggsave(filename = '../picture/Macrophage/umap_recluster_by_0.4.pdf')
DimPlot(sce,reduction = "umap",label=T,
        group.by = 'orig.ident')
ggsave(filename = '../picture/Macrophage/umap_sce_subcluster_res.0.4_by_patients.pdf')
table(sce@meta.data$orig.ident,sce@meta.data$seurat_clusters)
library(gplots)
colnames(sce@meta.data)
p1.compare=plot_grid(ncol = 3,
                     DimPlot(sce, reduction = "pca", group.by = "group")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce, reduction = "tsne", group.by = "group")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(sce, reduction = "umap", group.by = "group")+NoAxes()+ggtitle("UMAP"),

                     DimPlot(sce, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                     DimPlot(sce, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                     DimPlot(sce, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p1.compare
ggsave(plot=p1.compare,filename="../picture/Macrophage/Before_inter.pdf",units = "cm",width = 18,height = 12)
library(harmony)
seuratObj <- RunHarmony(sce, "orig.ident")
names(seuratObj@reductions)
seuratObj <- RunUMAP(seuratObj, dims =
```

```
1:15,
                    reduction = "harmony")
DimPlot(seuratObj,reduction = "umap",label=T )

p2.compare=plot_grid(ncol = 3,
                    DimPlot(seuratObj, reduction = "pca", group.by = "group")+NoAxes()+ggtitle("PCA"),
                    DimPlot(seuratObj, reduction = "tsne", group.by = "group")+NoAxes()+ggtitle("tSNE"),
                    DimPlot(seuratObj, reduction = "umap", group.by = "group")+NoAxes()+ggtitle("UMAP"),

                    DimPlot(seuratObj, reduction = "pca", group.by = "orig.ident")+NoAxes()+ggtitle("PCA"),
                    DimPlot(seuratObj, reduction = "tsne", group.by = "orig.ident")+NoAxes()+ggtitle("tSNE"),
                    DimPlot(seuratObj, reduction = "umap", group.by = "orig.ident")+NoAxes()+ggtitle("UMAP")
)
p2.compare
ggsave(plot=p2.compare,filename="../picture/Macrophage/after_inter.pdf",units = "cm",width = 18,height = 12)
sce=seuratObj
sce <- FindNeighbors(sce, reduction = "harmony",dims = 1:15)
sce <- FindClusters(sce, resolution = 0.1)
table(sce@meta.data$seurat_clusters)
DimPlot(sce,reduction = "umap",label=T)
ggsave(filename = 'harmony_umap_recluster_by_0.1.pdf')
p1 <- DimPlot(sce,reduction = "umap",label=T,
              group.by = 'orig.ident')
ggsave(p1,filename = '../picture/Macrophage/harmony_umap_sce_recluster_by_patients.pdf',units = "cm",width = 18,height = 12)
p2 <- DimPlot(sce,reduction = "umap",label=T,
              group.by = 'group')
p <- p1|p2
ggsave(p,filename = '../picture/Macrophage/harmony_umap_sce_recluster_by_patients_group.pdf',units = "cm",width = 28,height = 12)
rm(list = ls())
library(monocle)
library(Seurat)
library(tidyverse)
dir.create("monocle/")
dir.create("monocle_data/")
load(file = '../subdata/first_sce.Rdata')
colnames(sce@meta.data)
Idents(sce) <- "CCA_snn_res.0.1"
DimPlot(sce,reduction = "umap",label=T )
table(sce$seurat_clusters)
tmp = data.frame(cell = rownames(sce@meta.data),
                 cluster = sce$seurat_clusters)
tmp_list = split(tmp, tmp$cluster)
names(tmp_list)
cells = unlist(lapply(tmp_list, function(x){
  #x = tmp_list[[1]]
  set.seed(1)
  if(nrow(x)>1000){
    cells = x$cell[sample(1:nrow(x),1000)]
  } else {
    cells = x$cell
  }
  return(cells)
}))
tmp_sub = tmp[cells,]
table(tmp_sub$cluster)

sce_sampled = sce[,tmp_sub$cell]
dim(sce_sampled)
saveRDS(sce_sampled, file = "monocle/monocle2_sample.rds")
scRNAsub <- sce_sampled
data <-
```

```
as(as.matrix(scRNAsub@assays$RNA@counts), 'sparseMatrix')
pd <- new('AnnotatedDataFrame', data = scRNAsub@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)
diff.wilcox = FindAllMarkers(scRNAsub)
all.markers = diff.wilcox %>%dplyr::select(gene, everything()) %>% subset(p_val<0.05)
diff.genes <- subset(all.markers,p_val_adj<0.01)$gene
mycds <- setOrderingFilter(mycds, diff.genes)
p1 <- plot_ordering_genes(mycds)
p1
#FindVariableFeatures(scRNAsub, selection.method = "vst", nfeatures = 2000)
var.genes <- VariableFeatures(scRNAsub)
mycds <- setOrderingFilter(mycds, var.genes)
p2 <- plot_ordering_genes(mycds)
p <- p1|p2|p3
ggsave(p,filename = "monocle/FindVariable_gene.pdf")
mycds <- reduceDimension(mycds, max_components = 2, method = 'DDRTree')
mycds <- orderCells(mycds)
save(mycds,file = "monocle/mycds.monocle2.Rdata")
p1 = plot_cell_trajectory(mycds, color_by = "State")
ggsave(p1,filename = "monocle/new.macrophage.state.pdf")
p1 = plot_cell_trajectory(mycds, color_by = "seurat_clusters")
ggsave(p1,filename = "monocle/new.macrophage.seurat_clusters.pdf")
plot3 <- plot_cell_trajectory(mycds, color_by = "Pseudotime")
ggsave(plot3,filename = "monocle/new.macrophage.Pseudotime.pdf")
```