**安诺Seurat差异分析结果rds文件介绍**

[1. 单细胞技术简介 2](#_Toc28706824)

[2. Seurat介绍 3](#_Toc28706825)

[3. rds文件的来源 3](#_Toc28706826)

[4. rds文件读取 3](#_Toc28706827)

[5. rds文件结构说明 3](#_Toc28706828)

[5.1 assays的结构如下 4](#_Toc28706829)

[5.2 meta.data矩阵 5](#_Toc28706830)

[5.3 active.ident矩阵 5](#_Toc28706831)

[5.4 reductions矩阵 5](#_Toc28706832)

[6 rds文件说明 6](#_Toc28706833)

[6.1 assays 6](#_Toc28706834)

[6.1.1 counts矩阵 6](#_Toc28706835)

[6.1.2 data矩阵 7](#_Toc28706836)

[6.1.3 scale.data矩阵 8](#_Toc28706837)

[6.2 meta.data矩阵 8](#_Toc28706838)

[6.3 active.ident 9](#_Toc28706839)

[6.4 reductions 10](#_Toc28706840)

[7 总结： 11](#_Toc28706841)

[8 参考说明： 11](#_Toc28706842)

**安诺Seurat差异分析结果rds文件介绍**

by 姚盟成

# 1. 单细胞技术简介

随着2009年汤富酬实验室首次发表单细胞研究技术，经过十年的发展，单细胞技术得到了较大的发展，特别是10xGenomics单细胞转录技术的发展，使得单细胞转录组的成本有了极大的降低，极大的提高的单细胞转录组技术的发展，大大的促进了单细胞转录组技术在不同领域的应用。



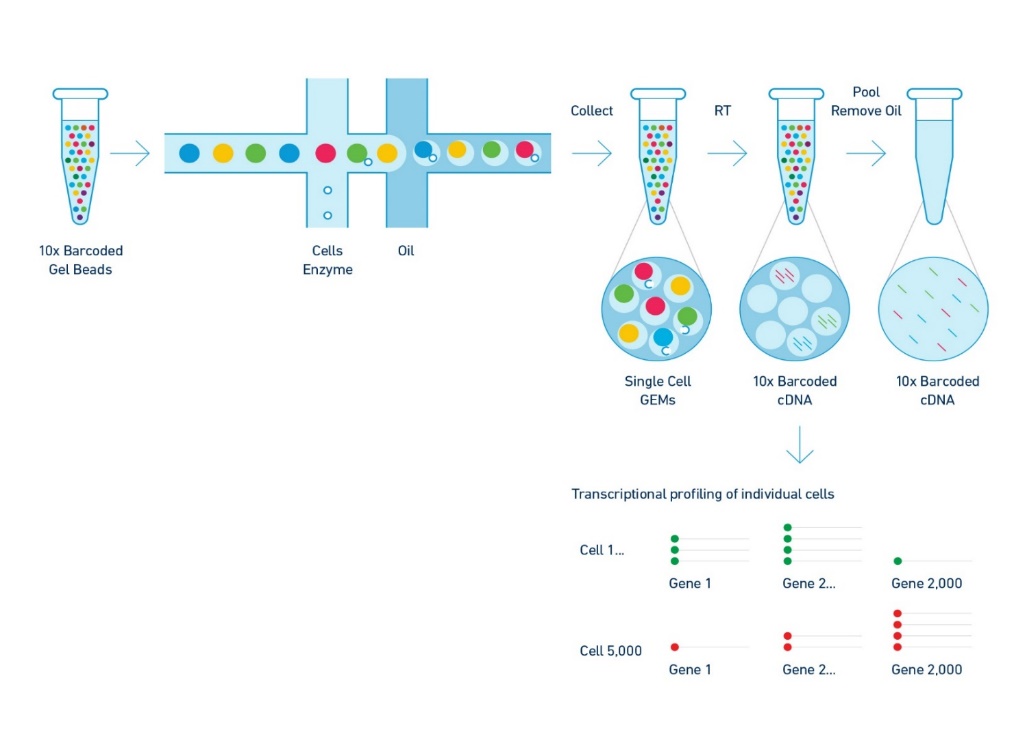


图1 左上为10xGenomics公司转录组产品；右上为单细胞转录组文库结果；下图为10xGenomics单细胞转录建库流程

随着单细胞技术的发展（成本和通量），产生大量的单细胞数据，这些数据对发现新的细胞亚群、细胞发育轨迹等研究产生了有益的作用，但是这些数据的比较综合分析，比如不同亚群之间和不同条件之间的比较分析，到目前为止仍然是一个难题。基于R语言的单细胞数据分析**Seurat**软件包的出现，成功的解决了此问题，特别是**Seurat3.0**版本出来以后，相比于3.0.0之前的版本，其大大的提高分析的效率、增加了多组学分析模块、增加了新的分析方法，这是一次巨大的升级。

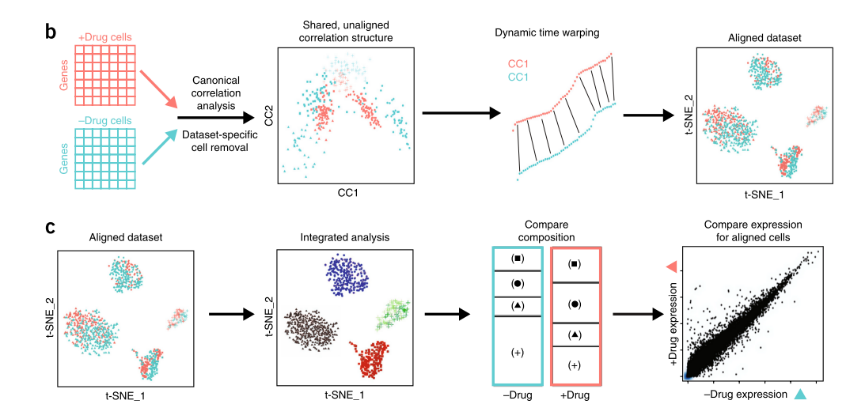


图2 单细胞转录组差异分析流程

# 2. Seurat介绍



Seurat是由SATIJA LAB开发专门用于单细胞分析的综合性工具，之所以叫Seurat，是因为聚类图让作者想起了乔治.修拉的画。也是目前应用最广的工具，特别是在单细胞转录组领域，基本上是十之八九单细胞转录组分析都会使用此分析工具，而且随着基于Seurat工具结果的第三方工具的出现，使得该软件更具有通用性。在安诺，Seurat工具输出的rds文件是几乎是所有单细胞高级分析的输入文件，包括monocle、SingleR、Cellassign等等。下面将针对rds文件的接口进行说明。

# 3. rds文件的来源

rds文件来源于所有Seurat分析完成后的S3或者S4对象文件，一般保存文件的命令为：

saveRDS(immune.combined, file = paste(prefix,'immune\_combined.rds',sep='\_'))

其中immune.combined为Seurat分析的S3或者S4对象，prefix为输出文件前缀名称，此结果一般是多样本差异比较分析的rds文件，单个样品分析的rds文件可能与此不一致。

# 4. rds文件读取

如果分析完成后，我们再次进行调整或者分析，或者使用此rds文件进行其他补充分析，我们需要对此rds文件再次进行读取，进行后续分析。

rds文件读取命令为：

rds<-readRDS(paste(prefix,'immune\_combined.rds',sep='\_'))

其中paste(prefix,'immune\_combined.rds',sep='\_')为rds文件名称，

**备注：**

用以下R进入交互模式

/annoroad/data1/bioinfo/PMO/yaomengcheng/Anaconda3/bin/R

require(Seurat)#载入Seurat包

# 5. rds文件结构说明

读取rds文件，我们需要了解rds文件的数据结构，所有数据结构属性，都可以用str(rds)命令查看。**下面为rds文件的数据结构：**

Formal class 'Seurat' [package "Seurat"] with 12 slots

..@ assays :List of 2

.. ..$ RNA :Formal class 'Assay' [package "Seurat"] with 7 slots

.. ..$ integrated:Formal class 'Assay' [package "Seurat"] with 7 slots

..@ meta.data :'data.frame': 34957 obs. of 24 variables:

.. ..$ orig.ident : chr [1:34957] "spleeA0d" "spleeA0d" "spleeA0d" "spleeA0d" ...

.. ..$ nCount\_RNA : num [1:34957] 4128 2865 3322 6944 3414 ...

.. ..$ nFeature\_RNA : int [1:34957] 1234 1106 1058 1903 1043 1872 1935 1244 1756 11 ...

.. ..$ stim : chr [1:34957] "Spleen0d" "Spleen0d" "Spleen0d" "Spleen0d" ...

.. ..$ percent.mt : num [1:34957] 3.78 5.65 4.64 3.86 3.43 ...

.. ..$ integrated\_snn\_res.0.5: Factor w/ 12 levels "1","10","11",..: 2 2 2 2 2 8 2 2 2 2 ...

**.. ..$ seurat\_clusters : Factor w/ 12 levels "1","2","3",..: 2 2 2 2 2 6 2 2 2 2 ...**

.. ..$ S.Score : num [1:28518] -0.00676 -0.07437 -0.0252 -0.06953 -0.00712 ...

.. ..$ G2M.Score : num [1:28518] -0.00946 -0.04809 0.08723 -0.03961 0.02987 ...

.. ..$ Phase : chr [1:28518] "G1" "G1" "G2M" "G1" ...

..@ active.assay: chr "RNA"

..@ active.ident: Factor w/ 12 levels "1","2","3","4",..: 2 2 2 2 2 6 2 2 2 2 ...

.. ..- attr(\*, "names")= chr [1:34957] "AAACCTGAGACAAAGG.spleeA0d" ...

..@ graphs :List of 2

.. ..$ integrated\_nn :Formal class 'Graph' [package "Seurat"] with 6 slots

.. ..$ integrated\_snn:Formal class 'Graph' [package "Seurat"] with 6 slots

..@ neighbors : list()

..@ reductions :List of 2

.. ..$ pca :Formal class 'DimReduc' [package "Seurat"] with 8 slots

.. ..$ umap:Formal class 'DimReduc' [package "Seurat"] with 8 slots

..@ project.name: chr "SeuratProject"

..@ misc :List of 1

.. ..$ markers:'data.frame': 45856 obs. of 7 variables:

.. .. ..$ p\_val : num [1:45856] 0 0 0 0 0 0 0 0 0 0 ...

.. .. ..$ avg\_log2FC: num [1:45856] 1.92 1.91 1.63 1.62 1.6 ...

.. .. ..$ pct.1 : num [1:45856] 0.496 0.935 0.98 0.762 0.947 0.969 0.982 0.975 0.889 0.997 ...

.. .. ..$ pct.2 : num [1:45856] 0.14 0.436 0.552 0.364 0.47 0.58 0.533 0.574 0.404 0.633 ...

.. .. ..$ p\_val\_adj : num [1:45856] 0 0 0 0 0 0 0 0 0 0 ...

.. .. ..$ cluster : Factor w/ 22 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...

.. .. ..$ gene : chr [1:45856] "Cxcl13" "Ptgs2" "Ugdh" "Eif4e" ...

..@ version :Classes 'package\_version', 'numeric\_version' hidden list of 1

.. ..$ : int [1:3] 3 1 1

..@ commands :List of 7

.. ..$ FindIntegrationAnchors :Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

.. ..$ PairwiseIntegrateReference :Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

.. ..$ ScaleData.integrated :Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

.. ..$ RunPCA.integrated :Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

.. ..$ RunUMAP.integrated.pca :Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

.. ..$ FindNeighbors.integrated.pca:Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

.. ..$ FindClusters :Formal class 'SeuratCommand' [package "Seurat"] with 4 slots

..@ tools :List of 1

.. ..$ Integration:Formal class 'IntegrationData' [package "Seurat"] with 7 slots

从上述结果可以看出，rds文件的以及对象有12 slots是，每一个slot是可以通过rds@[slots名称]来读取或者获取。

上述为rds对象整体的结构，但是我们一般关注的slots只有四个，因此我们应该重点关注一下四个slots的结构，这四个slots为：assays、meta.data、active.ident、reductions。

## 5.1 assays的结构如下

List of 2

$ RNA :Formal class 'Assay' [package "Seurat"] with 7 slots RNA原始结果

.. ..@ counts :Formal class 'dgCMatrix' [package "Matrix"] with 6 slots counts矩阵

.. ..@ data :Formal class 'dgCMatrix' [package "Matrix"] with 6 slots normalize矩阵

.. ..@ scale.data : num[0 , 0 ] scale矩阵

.. ..@ key : chr "rna\_"

.. ..@ var.features : logi(0)

.. ..@ meta.features:'data.frame': 16060 obs. of 0 variables

.. ..@ misc : NULL

$ integrated:Formal class 'Assay' [package "Seurat"] with 7 slots 多个样品合并后整合结果

.. ..@ counts :Formal class 'dgCMatrix' [package "Matrix"] with 6 slots counts矩阵

.. ..@ data :Formal class 'dgCMatrix' [package "Matrix"] with 6 slots normalize矩阵

.. ..@ scale.data : num [1:2000, 1:34957] 0.1146 0.6493 -0.0481 -0.2345 -0.406 ... scale矩阵

.. .. ..- attr(\*, "dimnames")=List of 2

.. ..@ key : chr "integrated\_"

.. ..@ var.features : chr [1:2000] "Trbv15" "Trbv17" "Trbv29" "Gzma" ...

.. ..@ meta.features:'data.frame': 2000 obs. of 0 variables

.. ..@ misc : List of 1

## 5.2 meta.data矩阵

'data.frame': 34957 obs. of 7 variables:

$ orig.ident : chr "spleeA0d" "spleeA0d" "spleeA0d" "spleeA0d" ...

$ nCount\_RNA : num 4128 2865 3322 6944 3414 ...

$ nFeature\_RNA : int 1234 1106 1058 1903 1043 1872 1935 1244 1756 1126 ...

$ stim : chr "Spleen0d" "Spleen0d" "Spleen0d" "Spleen0d" ...

$ percent.mt : num 3.78 5.65 4.64 3.86 3.43 ...

$ integrated\_snn\_res.0.5: Factor w/ 12 levels "1","10","11",..: 2 2 2 2 2 8 2 2 2 2 ...

$ seurat\_clusters : Factor w/ 12 levels "1","2","3",..: 2 2 2 2 2 6 2 2 2 2 ...

$Group ：chr [1:28518] "SQ" "SQ" "SQ" "SQ" ...

$Cell\_type

$ S.Score : num [1:28518] -0.00676 -0.07437 -0.0252 -0.06953 -0.00712 ...

$ G2M.Score : num [1:28518] -0.00946 -0.04809 0.08723 -0.03961 0.02987 ...

$ Phase : chr [1:28518] "G1" "G1" "G2M" "G1" ...

meta.data每行是一个cell，行名是barcode，是记录细胞特性的数据，可以通过以下命令对rds进行部分亚群数据的选取。如下面命令所示，仅仅选取seurat\_clusters=1的细胞。

Idents(rds) <- rds@meta.data$ seurat\_clusters

tmp<-subset(rds,idents = 1, invert = F)

备注：还可以rds@meta.data$Type<-cell\_types额外添加一列细胞特征属性，再通过以上命令选出部分的细胞类型。

## 5.3 active.ident矩阵

Factor w/ 12 levels "1","2","3","4",..: 2 2 2 2 2 6 2 2 2 2 ...

- attr(\*, "names")= chr [1:34957] "AAACCTGAGACAAAGG.spleeA0d"

## 5.4 reductions矩阵

List of 2

$ pca :Formal class 'DimReduc' [package "Seurat"] with 8 slots pca降维结果

.. ..@ cell.embeddings : num [1:34957, 1:30] -1.32 -2.93 -9 -3.04 ... 降维坐标轴

.. .. ..- attr(\*, "dimnames")=List of 2

.. ..@ feature.loadings : num [1:2000, 1:30] -0.001406 0.002483 -0.002013 0.018346 ...

.. .. ..- attr(\*, "dimnames")=List of 2

.. ..@ feature.loadings.projected: num[0 , 0 ]

.. ..@ assay.used : chr "integrated"

.. ..@ stdev : num [1:30] 6.71 4.13 3.75 3.51 3.28 ...

.. ..@ key : chr "PC\_"

.. ..@ jackstraw :Formal class 'JackStrawData' [package "Seurat"] with 4 slots

.. ..@ misc :List of 1

.. .. ..$ total.variance: num 1885

$ umap:Formal class 'DimReduc' [package "Seurat"] with 8 slots UMAP降维结果

.. ..@ cell.embeddings : num [1:34957, 1:2] 2.93 1.85 1.88 ... UMAP降维坐标轴

.. .. ..- attr(\*, "scaled:center")= num [1:2] -0.334 -0.541

.. .. ..- attr(\*, "dimnames")=List of 2

.. ..@ feature.loadings : num[0 , 0 ]

.. ..@ feature.loadings.projected: num[0 , 0 ]

.. ..@ assay.used : chr "integrated"

.. ..@ stdev : num(0)

.. ..@ key : chr "UMAP\_"

.. ..@ jackstraw :Formal class 'JackStrawData' [package "Seurat"] with 4 slots

.. ..@ misc : list()

上述所有矩阵结构后面红色字说明单是比较关注的数据，其他是一些中间结果，不太关注。

# 6 rds文件说明

下面将对上面四个比较关注的slots（assays、meta.data、active.ident、reductions）进行具体的说明。

## 6.1 assays

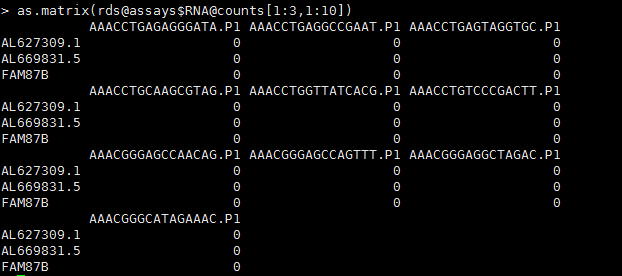
### 6.1.1 counts矩阵

此为我们输入或者标准化后的单个细胞-基因的表达矩阵，其中assays下有两个对象，分别为RNA和integrated，RNA为输入的原始细胞-基因表达矩阵，integrated为整合后的细胞-基因表达矩阵。在RNA或者integrated对象下面有counts、data、scale.data三个比较重要的矩阵，这三个矩阵分别为原始的UMI细胞-基因矩阵、normalize后的细胞-基因表达矩阵、scale后的细胞-基因矩阵。我们如果做其他补充分析的话，一般的输入矩阵都是使用RNA中的矩阵，其中counts矩阵使用最广泛。三个矩阵具体结构如下：

查看矩阵命令为：rds@assays$RNA@counts，从下图可以看出，其矩阵为稀疏矩阵(稀疏矩阵中所有的0都用点代替)，行的行名为基因名称，**列名为细胞名称（细胞名称=细胞barcode.样品名称）**，此矩阵结构也是单细胞分析结果中最常用的矩阵。

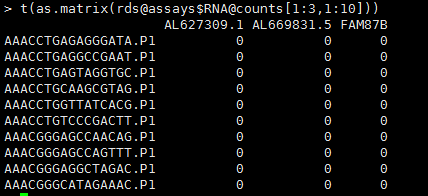


但是一般我们可能做其他分析的时候，不能使用稀疏矩阵作为输入，这时我们就需要把稀疏矩阵改成正常的表达矩阵，命令为：as.matrix(rds@assays$RNA@counts) #矩阵后的[1:3,1:10]数字为去前3行和前10列。

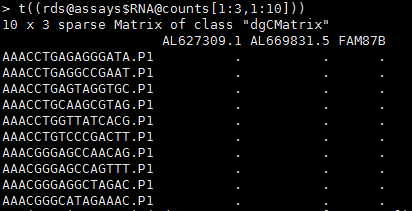


当然，也有的分析输入的矩阵与现在的行和列是相反的，这里就需要我们对矩阵进行转置，R中转置的函数很简单，只要使用t(maxtrix)即可。稀疏矩阵不能直接转置，我们可以通过两种方式进行转置：

一种是：直接将稀疏矩阵转换成普通矩阵，然后通过命令直接转置，转置命令如下：t(as.matrix(rds@assays$RNA@counts[1:3,1:10]))



另外一种是：直接对稀疏矩阵进行转置，需要导入library(Matrix)包，再转置。转置命令为：t((rds@assays$RNA@counts[1:3,1:10]))



### 6.1.2 data矩阵

data矩阵是对counts矩阵进行normalize后的矩阵，这个矩阵对每个细胞进行了一个文库校正，消除细胞之间的差异，因此此矩阵可以用于一般需要比较基因在不同细胞表达的情况。其获取矩阵和操作与counts矩阵一致，只是把最后的矩阵改成data即可。

### 6.1.3 scale.data矩阵

scale.data矩阵是在data矩阵的基础上进行标准化后的一个矩阵，一般做其他分析的时候尽量不要用此矩阵，此矩阵一般仅在Seurat分析的时候用。

## 6.2 meta.data矩阵

meta.data矩阵是我们另外一个使用特别常见的矩阵，里面有很多有用的信息，包括样品信息，亚群结果，样品来源等等。此矩阵一般有至少8列(head([rds@meta.data)](mailto:rds@meta.data)))，今后在矩阵增加分组的信息列，到时候名称为Group：

第一列：orig.ident；样品来源

第二列：nCount\_RNA；每个细胞的UMI数目

第三列：nFeature\_RNA；每个细胞的基因表达数目

第四列：stim；每个细胞所属样品

第五列：percent.mt；每个细胞的线粒体基因表达占比

第六列：percent.HB；每个细胞血红蛋白基因表达占比

第七列：integrated\_snn\_res.0.5；分辨率为0.5时细胞亚群编号，如果有多个分辨率，

可能有多个结果。

第八列：seurat\_clusters；当前默认的细胞亚群编号，默认是最后分辨率结果。

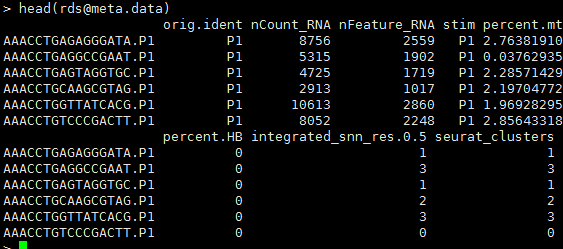
第九列：Group; 样本所属的组别

第十列：Cell\_type：cluster 注释得到的细胞名称

第十一列：S.Score：细胞周期S期评分

第十二列：G2M.Score：细胞周期G2/M期评分

第十三列：Phase：评估的细胞周期



这里的每一列，是画细胞umap图的时候参数可以选的列，比如stim、percent.mt、seurat\_clusters、nCount\_RNA、nFeature\_RNA，或者加上基因名称，下面有画图示范代码和画图结果图片。

library(ggplot2)

library(cowplot)

p\_tmp<<-theme(panel.grid=element\_blank(), legend.background = element\_rect(colour = NA),

legend.title = element\_blank(),legend.text = element\_text(color="black",size=30),

axis.text.x = element\_text(color="black",size=30),

axis.text.y = element\_text(color="black",size=30),

axis.title.x = element\_text(face="plain", color="black",size=30),

axis.title.y = element\_text(face="plain", color="black",size=30))

pdf("yao.pdf",w=30,h=15)

p1 <- DimPlot(rds, reduction = "umap", group.by = "stim",pt.size = 1)+p\_tmp

p2 <- DimPlot(rds, reduction = "umap", group.by = "seurat\_clusters",label = TRUE,label.size = 8,pt.size = 1)+p\_tmp

p3 <- FeaturePlot(rds, features="percent.mt",cols = c("lightgrey", "blue"),pt.size = 1)+p\_tmp

p4 <- FeaturePlot(rds, features="nCount\_RNA",cols = c("lightgrey", "blue"),pt.size = 1)+p\_tmp

p5 <- FeaturePlot(rds, features="nFeature\_RNA",cols = c("lightgrey", "blue"),pt.size = 1)+p\_tmp

p6 <- FeaturePlot(rds, features="CD8A",cols = c("lightgrey", "blue"),pt.size = 1)+p\_tmp

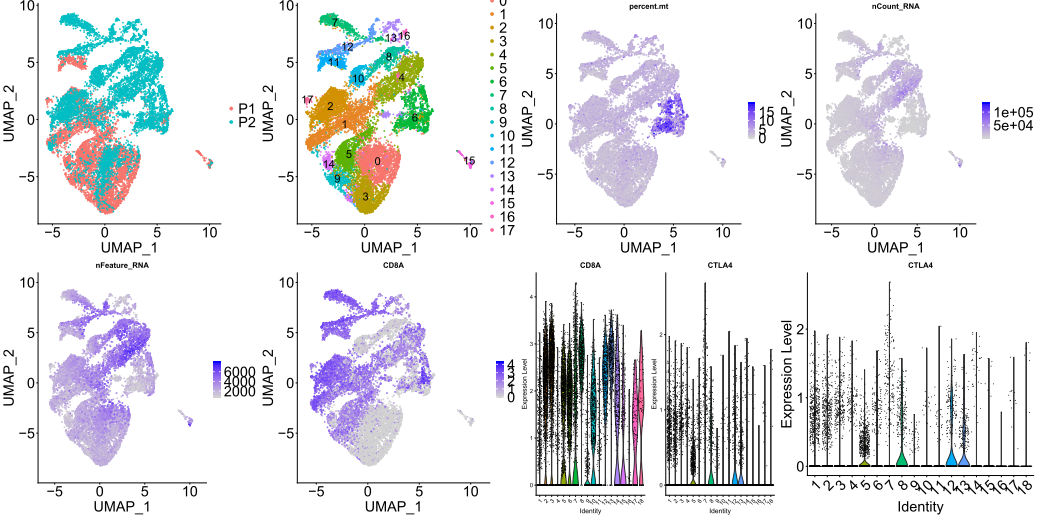
p7 <- VlnPlot(rds, features=c("CD8A","CTLA4"),pt.size = 0.2)+p\_tmp

p8 <- VlnPlot(rds, features=c("CTLA4"),pt.size = 0.2)+p\_tmp+theme(legend.position="none")

p9<-plot\_grid(p1, p2,p3,p4,p5,p6,p7,p8,ncol=4)

print(p9)

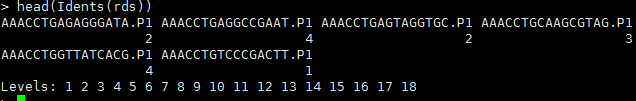
dev.off()



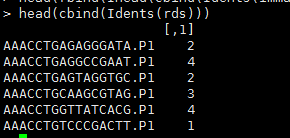
上图基本上包括了单细胞转录组分析的常见分析结果，分别为：样品分布图、亚群名称图、线粒体、每个细胞基因表达数、每个细胞UMI数目、基因表达分布图、两个基因一起小提琴图、一个基因的小提琴图。FeaturePlot和VlnPlot画图函数中的features参数可以是一个基因或者多个基因。

## 6.3 active.ident

这个是细胞亚群结果矩阵，一般通过Idents(rds)命令提取结果，这个结果一般与meta.data矩阵中的seurat\_clusters一致，但是如果是我们对细胞亚群重新命名的话，这个结果将不会一致。

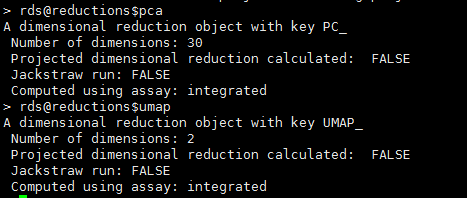


这个说是矩阵，其实就是一个向量，向量为细胞亚群编号，向量的名称为细胞编号。当然，我们经常想把他换成矩阵，其命令如下：rbind(Idents(rds))



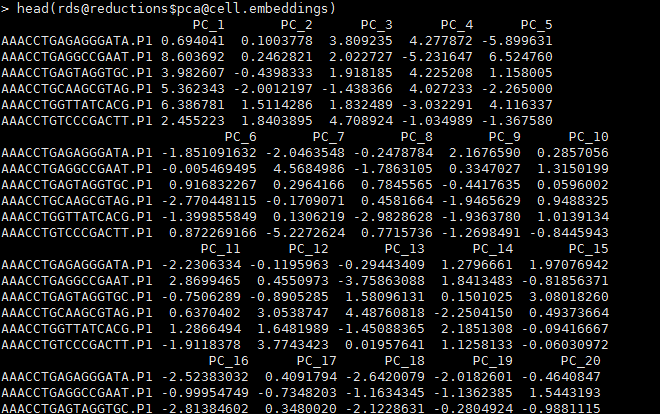
## 6.4 reductions

reductions对象下有降维聚类的结果，包括聚类的中间结果，一般我们对这两个对象不怎么使用，我们使用最多的是每个细胞的坐标轴，输出坐标轴文件，然后另外通过ggplot画图。其对象的有pca降维结果和umap降维结果：



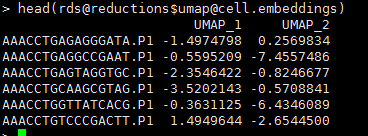
pca降维的坐标轴结果如下：

head(rds@reductions$pca@cell.embeddings)



umap降维坐标轴结果如下：

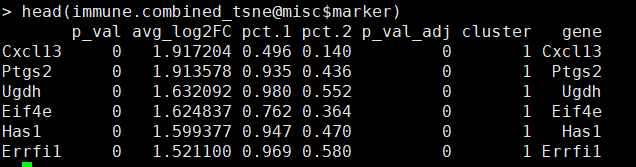
head(rds@reductions$umap@cell.embeddings)



上述四矩阵，是我们最常用的矩阵。一般来说，我们做的常规分析或者做其他工具开发的接口已经足够了。当如，如果想更进一步研究对象中每个数据的含义，可以参考官方说明进行研究。

## 6.5 mics

mics对象下存放聚类后markers等信息。



1. p\_val：假设检验后得到的原始P值；
2. avg\_log2FC：两组之间平均表达差异倍数的对数值。正值表示该基因在第一组中的表达更高；
3. pct.1：第一组中检测到表达该基因的细胞所占的百分比
4. pct.2：第二组中检测到表达该基因的细胞所占的百分比
5. p\_val\_adj：多重检验校正后得到的校正后的P值。
6. cluster：基因所属cluster；
7. gene：基因name

# 7 总结：

1. 如果做其他软件工具一般需要的数据主要两个，分别是counts矩阵和meta.data矩阵，一般来说其他软件输入数据主要使用这个两个矩阵。
2. 如果想要对Seurat进行进一步分析，可以对meta.data矩阵进行修改，增加列都是可以的。
3. 如果想自己根据Seurat结果，结合表达矩阵进行绘图，可能需要提取坐标轴结果和基因表达矩阵，这里建议基因表达矩阵尽量用RNA的data矩阵。
4. 现在rds文件几个默认名称规则：

**细胞名称：细胞barcode.样品名称（meta.data矩阵行名，或者表达矩阵的列名）**

**每个细胞所属样品：stim（meta.data矩阵中的列名）**

**每个样品所属分组：Group（如果有的的话，为meta.data矩阵的列名）**

**线粒体基因表达占比：percent.mt（meta.data矩阵的列名）**

**每个细胞表达的UMI数目：nCount\_RNA（meta.data矩阵的列名）**

**每个细胞表达的基因数目：nFeature\_RNA（meta.data矩阵的列名）**

**每个细胞表达血红蛋白的占比：percent.HB（meta.data矩阵的列名）**

**每个细胞亚群命名：seurat\_clusters（meta.data矩阵的列名）**

当然，如果有其他的信息，比如是时间序列、或者某个基因表达的情况，都可以添加到meta.data矩阵中。

# 8 参考说明：

Seurat工具说明文档： <https://cran.r-project.org/web/packages/Seurat/Seurat.pdf>

Seurat工具使用pipeline：<https://satijalab.org/seurat/>

修改人：陈静、苑赞、姚佳英、李萌、郭磊