

单细胞转录组数据分析标准流程报告

pbmc3k 数据集 Seurat v5 教程

2022 届河北医科大学

June 20, 2025

Contents

I 认知准备：从宏观理解单细胞分析	4
1 为什么需要单细胞测序?	4
2 本次分析的“作战地图”	4
3 分析环境与项目准备	4
II 数据净化：从原始数据到高质量矩阵	5
4 数据加载与 Seurat 对象创建	5
5 质量控制 (QC): 细胞的“海关安检”	6
5.1 指标一：细胞内基因数量 (nFeature_RNA)	6
5.2 指标二：细胞内总分子数 (nCount_RNA)	6
5.3 指标三：线粒体基因表达比例 (percent.mt)	7
6 数据标准化：消除技术偏差	8
III 模式发现：从高维数据中寻找结构	8
7 特征选择与数据缩放	8
8 线性降维 (PCA)	9
9 细胞聚类与非线性降维 (UMAP)	10
IV 生物学解读：从数字到细胞身份	11
10 寻找并导出各群落的 Marker 基因	11
10.1 使用热图宏观验证 Marker 基因	12
11 注释细胞类型：严谨的“侦探工作”	13
12 生成最终细胞图谱	13
附录：完整分析 R 脚本 (v7.0 最终专业版)	15

List of Figures

1	Seurat 对象初始化后的摘要信息。	6
2	QC 指标小提琴图。我们根据这三个指标的分布，设定合理的阈值（图中虚线所示意的位置）来过滤掉两端的异常细胞。	7
3	高变异基因 (HVGs) 筛选图。红点是被选出的 2000 个高变基因，它们将在后续分析中扮演主角。	9
4	PCA 结果可视化。左图显示细胞在 PC1 和 PC2 空间中已初步呈现分离趋势。右图帮助我们确定数据的“内在维度”。	10
5	细胞聚类与 UMAP 可视化初步结果。根据我们的参数，细胞被清晰地划分为 9 个不同的群落 (Cluster 0-8)。	11
6	最终细胞类型注释的 UMAP 图。该图是本次分析的核心成果，直观地展示了 pbmc3k 数据集中主要的免疫细胞亚群及其在转录组水平上的相对关系。 . . .	15

List of Tables

1	各细胞群落生物学注释汇总	13
---	------------------------	----

Part I

认知准备：从宏观理解单细胞分析

在敲下任何代码之前，让我们先建立对单细胞分析的宏观认知。

1 为什么需要单细胞测序？

【核心理念说明】

一个生动的类比：果汁 vs. 水果拼盘

传统的 RNA 测序 (Bulk RNA-seq) 就像是把一篮子不同类型的水果（比如苹果、香蕉、橙子）全部丢进榨汁机。最后你得到了一杯混合果汁，你能尝出里面大概有哪几种水果的味道，但你无法知道这篮水果里到底有多少个苹果、多少根香蕉。

而**单细胞 RNA 测序 (scRNA-seq)** 则像是把这篮水果一个一个拿出来，仔细地识别、分类和计数。最终，你得到的不是一杯模糊的果汁，而是一份清晰的清单：“苹果 5 个，香蕉 3 根，橙子 4 个……”。

在生物学研究中，这“水果”就是我们体内的细胞。组织和器官正是由成千上万种功能各异的细胞组成的。单细胞测序使我们能够以前所未有的分辨率，看清单个细胞的基因表达特征，从而揭示细胞的异质性、发现新的细胞亚群、描绘细胞的发育轨迹。

2 本次分析的“作战地图”

我们的整个分析流程就像一场精心策划的战役，可以分为四个主要阶段：

1. **数据净化**：拿到原始数据后，先进行严格的质量控制，把“老弱病残”的细胞数据剔除掉，并进行标准化，让所有数据站在同一起跑线上。
2. **模式发现**：在干净的数据中，通过降维 (PCA) 抓住主要矛盾，再通过聚类 (Clustering) 将相似的细胞“抱团”，形成不同的细胞群落。
3. **生物学解读**：为每个细胞群落寻找独特的“身份标识” (Marker 基因)，并通过这些标识来判断它们究竟是哪种类型的细胞 (T 细胞、B 细胞、还是单核细胞?)。
4. **可视化呈现**：将最终的细胞身份注释结果绘制成一张精美的“细胞星图” (UMAP 图)，直观展示样本中所有细胞的种类和关系。

3 分析环境与项目准备

此为基础准备工作，确保后续所有操作顺利进行。

- **安装 R 与 RStudio:** R 是分析引擎，RStudio 是操作界面。请确保两者均已正确安装。
- **安装核心 R 包:** 在开始分析前，请确保安装了本次分析所需的核心 R 包。
- **获取项目框架与数据:** 从课程的 GitHub 仓库下载项目框架，并根据指导在 RStudio 中设置好工作目录，运行数据下载脚本。

【技术要点】

R 包安装: 在 R/RStudio 的控制台中运行以下代码，安装分析所需的核心包。

```
1 ## 如果尚未安装 Seurat，请运行此代码
2 if (!requireNamespace("Seurat", quietly = TRUE)) {
3   install.packages("Seurat")
4 }
5 # 安装 dplyr 用于后续的数据整理
6 if (!requireNamespace("dplyr", quietly = TRUE)) {
7   install.packages("dplyr")
8 }
```

Part II

数据净化：从原始数据到高质量矩阵

本部分我们将完成从加载数据到获得可用于分析的、高质量表达矩阵的全过程。

4 数据加载与 Seurat 对象创建

首先，我们将 10x Genomics 平台产出的三个核心文件（barcodes, features, matrix）读入 R，并创建一个 Seurat 对象。

```
1 library(Seurat)
2 library(dplyr) % 加载dplyr包，方便后续数据处理
3
4 data_directory <- "03_data/filtered_gene_bc_matrices/hg19/"
5 pbmc.data <- Read10X(data.dir = data_directory)
6
7 pbmc <- CreateSeuratObject(counts = pbmc.data,
8                             project = "pbmc3k",
9                             min.cells = 3,
10                             min.features = 200)
```

【教学重点】

参数解读: `min.cells = 3` 表示一个基因至少要在 3 个细胞中被检测到才会被保留;
`min.features = 200` 表示一个细胞至少要检测到 200 个基因才被视为有效细胞。
这两个参数提供了第一道粗略的质控过滤。

```
> pbmc.data <- Read10X(data.dir = data_directory)
> pbmc.data
32738 x 2700 sparse Matrix of class "dgCMatrix"
[[ suppressing 49 column names 'AAACATACAACCAC-1', 'AAACATTGAGCTAC-1', 'AAACATTGATCAGC-1' ... ]]
[[ suppressing 49 column names 'AAACATACAACCAC-1', 'AAACATTGAGCTAC-1', 'AAACATTGATCAGC-1' ... ]]

MIR1302-10 . . . . .
. . . . .
FAM138A . . . . .
. . . . .
OR4F5 . . . . .
. . . . .
RP11-34P13.7 . . . . .
. . . . .
RP11-34P13.8 . . . . .
. . . . .
AL627309.1 . . . . .
. . . . .
RP11-34P13.14 . . . . .
. . . . .
RP11-34P13.9 . . . . .
. . . . .
AP006222.2 . . . . .
. . . . .
RP4-669L17.10 . . . . .
. . . . .

.....suppressing 2651 columns and 32718 rows in show(); maybe adjust options(max.print=, width=)
.....
[[ suppressing 49 column names 'AAACATACAACCAC-1', 'AAACATTGAGCTAC-1', 'AAACATTGATCAGC-1' ... ]]

KIR3DL2.1 . . . . .
. . . . .
AL590523.1 . . . . .
. . . . .
CT476828.1 . . . . .
. . . . .
```

Figure 1: Seurat 对象初始化后的摘要信息。

5 质量控制 (QC): 细胞的“海关安检”

这是至关重要的一步。我们通过三个核心指标来识别并剔除低质量的细胞数据。

5.1 指标一：细胞内基因数量 (nFeature_RNA)

这个指标反映了细胞的转录复杂性。

- **过低:** 可能意味着这个“细胞”其实是个空泡，或者细胞已经死亡，RNA 降解严重。
- **过高:** 可能意味着这是一个“双包体” (Doublet)，即两个细胞被错误地包裹进同一个反应微滴中，导致基因数量异常翻倍。

5.2 指标二：细胞内总分子数 (nCount_RNA)

这个指标主要与测序深度相关，但也反映了细胞的 RNA 总量。通常与 `nFeature_RNA` 呈正相关。

5.3 指标三：线粒体基因表达比例(percent.mt)

这是衡量细胞健康状况的“金标准”。

【核心理念说明】

线粒体是细胞的“能量工厂”。当细胞处于应激或濒死状态时，细胞膜的通透性会增加，导致细胞质中的 mRNA 大量流失，而线粒体相对较大，其内部的 mRNA 会被保留下来。因此，线粒体基因转录本所占的比例异常升高，往往是细胞状态不佳的强烈信号。

```
1 ## 计算线粒体基因比例（注意：人类线粒体基因以"MT-"开头，小鼠以"mt-"）
2 pbmc[["percent.mt"]] <- PercentageFeatureSet(pbmc, pattern = "^MT-")
3
4 # 可视化QC指标，为设定阈值提供依据
5 VlnPlot(pbmc, features = c("nFeature_RNA", "nCount_RNA",
  "percent.mt"), ncol = 3)
```

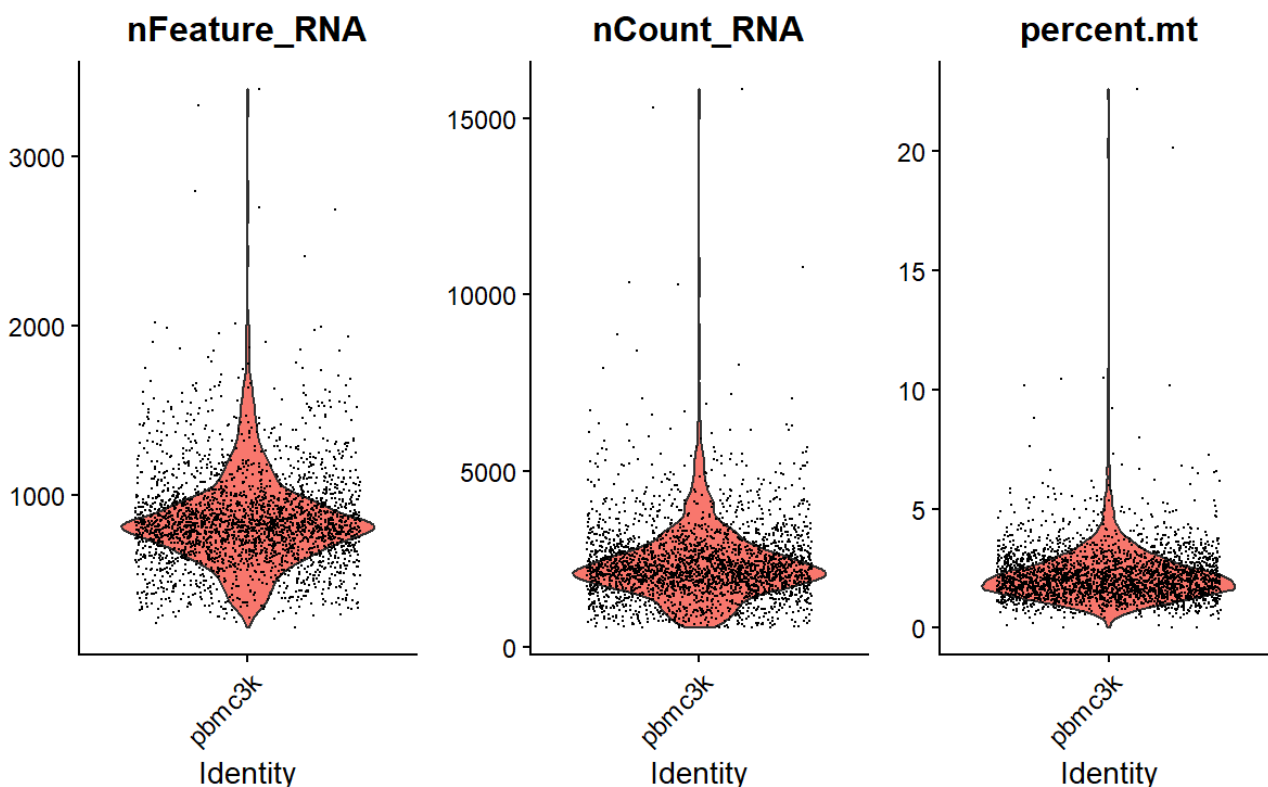


Figure 2: QC 指标小提琴图。我们根据这三个指标的分布，设定合理的阈值（图中虚线所示意的位置）来过滤掉两端的异常细胞。

【教学重点】

如何解读 QC 图并设定阈值？

在观察图 2 时，我们的目标是识别出分布两端的异常值。对于 nFeature_RNA，我们看到大部分细胞的基因数集中在 500 到 2000 之间，但有一小部分细胞的基因数超过

2500，这很可能是“双包体” (doublets)，我们应将其去除。对于 percent.mt，绝大多数细胞的线粒体比例都低于 5%，只有少数细胞的比例异常高，表明它们可能处于凋亡状态，也应去除。因此，我们设定 `nFeature_RNA > 200 & nFeature_RNA < 2500 & percent.mt < 5` 是一个基于数据分布的合理选择。

```
1 ## 根据观察执行过滤
2 pbmc <- subset(pbmc, subset = nFeature_RNA > 200 & nFeature_RNA <
  2500 & percent.mt < 5)
```

6 数据标准化：消除技术偏差

完成质控后，我们需要对数据进行标准化，以消除细胞间因测序文库大小（即 `nCount_RNA`）不同而引起的技术误差，确保后续的比较是在公平的基础上进行的。

```
1 ## 默认使用 LogNormalize 方法
2 pbmc <- NormalizeData(pbmc)
```

Part III

模式发现：从高维数据中寻找结构

在净化后的数据中，我们将通过一系列降维和聚类算法，发现隐藏在其中的细胞群体结构。

7 特征选择与数据缩放

- **寻找高变基因 (HVGs):** 从上万个基因中，找出在细胞间表达差异最大的 2000 个“明星基因”。这能帮助我们忽略背景噪音，聚焦于真正定义细胞身份的核心信号。
- **数据缩放 (Scaling):** 对每个高变基因的表达值进行中心化和标准化处理，防止少数高表达基因在后续分析中“一家独大”。

```
1 pbmc <- FindVariableFeatures(pbmc, selection.method = "vst",
  nfeatures = 2000)
2
3 # 仅对高变基因进行数据缩放（这是标准且推荐的做法）
4 pbmc <- ScaleData(pbmc, features = VariableFeatures(object = pbmc))
```


【教学重点】

为何只 Scale 高变基因? 我们后续的降维分析(PCA)是基于高变基因(Variable Features)进行的。因此,我们只需要对这些将要被用到的基因进行数据缩放即可。对所有基因进行缩放会极大地增加计算负担,且对最终结果没有正面影响,甚至可能引入噪音。

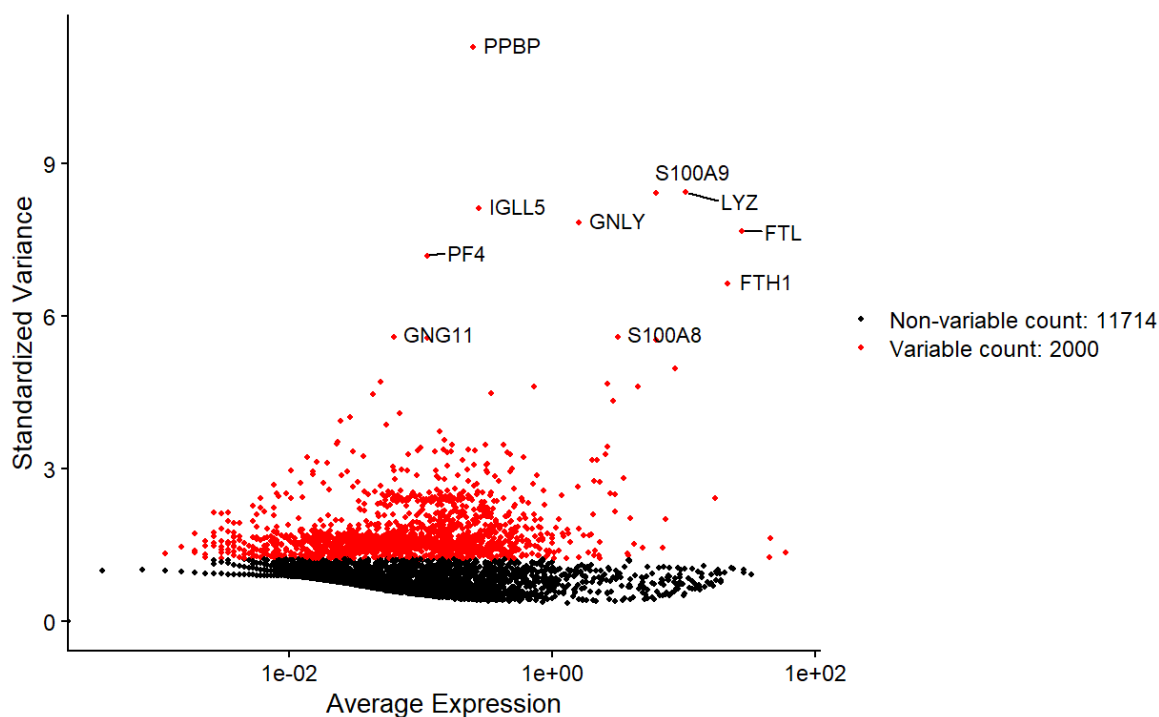


Figure 3: 高变异基因 (HVGs) 筛选图。红点是被选出的 2000 个高变基因, 它们将在后续分析中扮演主角。

8 线性降维 (PCA)

我们将 2000 个高变基因的复杂信息, 浓缩到少数几个“主成分”(PCs) 中。

【核心理念说明】

想象一下, 要描述一群人的特征, 你不需要测量他们的身高、体重、臂展、腿长等所有 2000 个指标。你可能会发现, 用“体型”(由身高和体重共同决定) 和“身材比例”(由臂展和腿长等决定) 这两个“主成分”就能很好地区分他们。PCA 做的就是类似的事情: 它将高度相关的基因信息组合成少数几个 PCs, 以抓住数据的主要变异方向。

```
1 pbmc <- RunPCA(pbmc, features = VariableFeatures(object = pbmc))
```

随后, 我们使用“肘部图”(Elbow Plot) 来决定保留多少个 PC 用于下游分析, 这是一个平衡“有效信号”与“随机噪音”的决策过程。

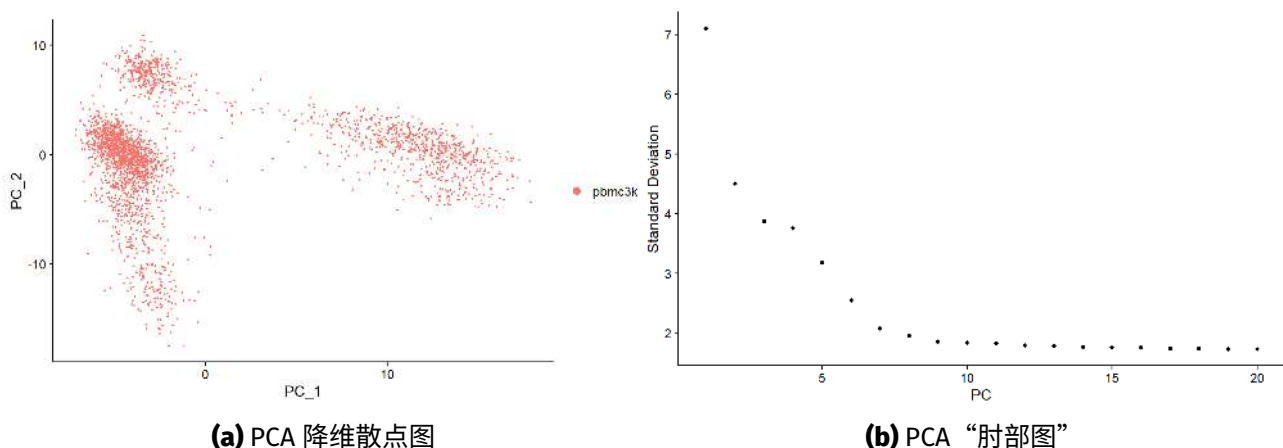


Figure 4: PCA 结果可视化。左图显示细胞在 PC1 和 PC2 空间中已初步呈现分离趋势。右图帮助我们确定数据的“内在维度”。

【教学重点】

解读“肘部图”以选择 PC 维度

这个图之所以被称为“肘部图”，是因为它形似手臂弯曲的肘部。图中，Y 轴表示每个主成分（PC）对数据方差的贡献度。我们期望前几个 PC 能解释大部分的数据变异，而后的 PC 则主要代表随机噪音。曲线的“拐点”或“肘部”（在本例中大约是 PC15 之后）标志着贡献度开始急剧下降并趋于平缓的点。选择这个点之前的 PC（即 1 到 15），意味着我们既保留了绝大部分的真实生物学信号，又避免了引入过多的技术噪音。

9 细胞聚类与非线性降维 (UMAP)

- **细胞聚类:** 基于选定的 15 个 PC 的“坐标”，我们首先构建一个细胞“社交网络”（‘Find-Neighbors’），然后利用图算法（如 Louvain 算法）在网络中寻找联系紧密的“小团体”（‘FindClusters’），这些小团体就是我们的细胞群落。
- **UMAP 可视化:** UMAP 是一种强大的非线性降维算法，非常适合将高维的聚类结果可视化到二维平面上，同时能很好地保持细胞群落的局部和全局结构。

```
1 pbmc <- FindNeighbors(pbmc, dims = 1:15)
2 pbmc <- FindClusters(pbmc, resolution = 0.5)
3 pbmc <- RunUMAP(pbmc, dims = 1:15)
```

【动手练习与思考】

如何选择最佳的聚类‘resolution’？

‘resolution’ 参数控制着聚类的精细程度。

- **值越低:** 得到的细胞群落数量越少，聚类结果更粗糙。
- **值越高:** 得到的细胞群落数量越多，聚类结果更精细，可能将大的细胞群细分为多个亚群。

在实际研究中，‘resolution = 0.5’ 是一个常用的初始值，但最佳值取决于数据集和研究目的。探索性分析时，可以尝试一系列的值（如 0.3, 0.5, 0.8, 1.2），观察 UMAP 图上细胞群落的划分是否与已知的生物学知识相符，从而选择一个最能揭示生物学意义的分辨率。

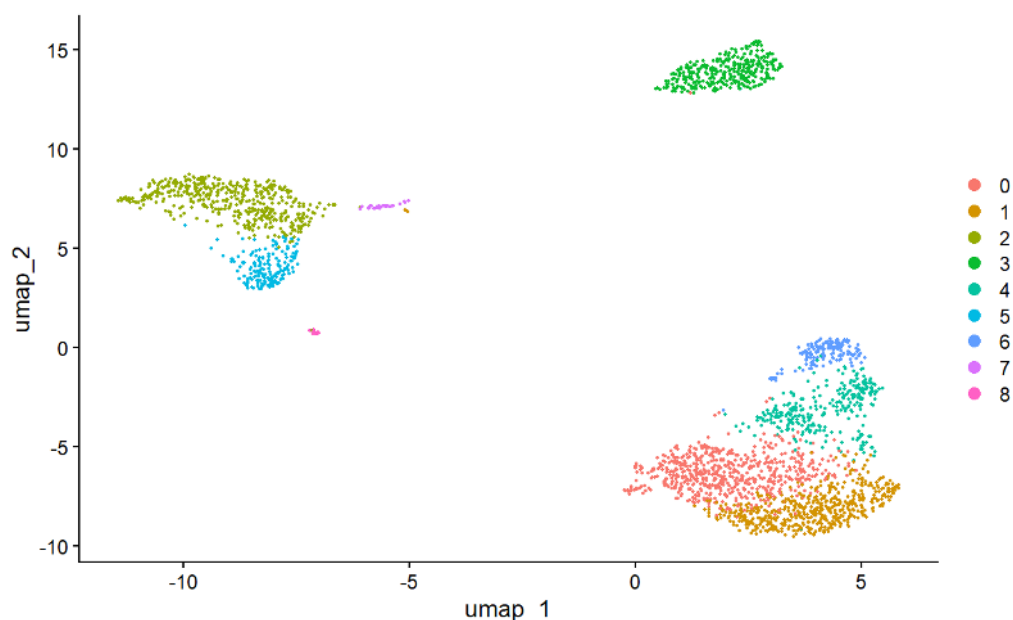


Figure 5: 细胞聚类与 UMAP 可视化初步结果。根据我们的参数，细胞被清晰地划分为 9 个不同的群落 (Cluster 0-8)。

Part IV

生物学解读：从数字到细胞身份

这是整个分析流程中最激动人心的部分：我们将赋予这些数字化的细胞群落明确的生物学身份。

10 寻找并导出各群落的 Marker 基因

Marker 基因，即差异表达基因，是指在某个特定细胞群落中表达水平显著高于其他所有群落的基因。它们是每个群落最独特的“身份名片”。

```
1 ## 寻找所有群落的Marker基因
2 pbmc.markers <- FindAllMarkers(pbmc,
3                                only.pos = TRUE,
```

(Continued)

```
4 min.pct = 0.25,  
5 logfc.threshold = 0.25)
```

【教学重点】

参数解读: `only.pos = TRUE` 表示我们只关心上调的基因; `min.pct = 0.25` 要求一个基因至少在目标群落 25% 的细胞中被检测到; `logfc.threshold = 0.25` 要求该基因在目标群落中的平均表达量比在其他群落中高出 $\log_2(0.25)$ 倍。这两个参数帮助我们筛选出既普遍又特异的 Marker 基因。

【技术要点】

结果保存: 分析过程中的重要结果, 如 Marker 基因列表, 应当及时保存为文件, 方便后续查阅、分享, 也无需重复计算。

```
1 ## 将完整的 Marker 列表写入文件  
2 write.csv(pbmc.markers, file = "pbmc3k_all_markers.csv", row.names  
  = FALSE)  
3  
4 # 筛选每个 cluster 的 top 15 Marker 并另存为一个文件  
5 top15.markers <- pbmc.markers %>%  
6   group_by(cluster) %>%  
7   slice_max(n = 15, order_by = avg_log2FC)  
8  
9 write.csv(top15.markers, file = "top15_markers_per_cluster.csv",  
  row.names = FALSE)
```

10.1 使用热图宏观验证 Marker 基因

除了逐个验证基因, 我们还可以用热图 (Heatmap) 从整体上检查 Top Marker 基因的表达模式, 这是展示聚类与差异表达分析结果的“金标准”图。

使用 DoHeatmap 可视化各群落 Top 10 Marker 基因

```
1 # 从已计算的 pbmc.markers 中提取每个 cluster 的 top 10 marker  
2 top10.markers <- pbmc.markers %>% group_by(cluster) %>% top_n(n =  
  10, wt = avg_log2FC)  
3  
4 # 绘制热图, 由于基因太多, 可能需要调整窗口大小查看  
5 DoHeatmap(pbmc, features = top10.markers$gene) + NoLegend()
```

理想情况下, 热图会呈现出清晰的、呈对角线分布的区块, 证明我们找到的 Marker 基因确实在对应的群落中特异性高表达。

11 注释细胞类型：严谨的“侦探工作”

这是一个结合生物学知识和数据证据的推理过程。

1. **查阅 Marker 列表:** 我们打开刚才生成的 `top15_markers_per_cluster.csv` 文件。
2. **知识库比对:** 我们将列表中每个 Cluster 的关键基因(通常是 `avg_log2FC` 最大、`p_val_adj` 最小的那些)与已知的细胞类型 Marker 基因数据库(如 CellMarkerDB、PanglaoDB)或相关文献进行比对。
3. **逻辑推理与注释:** 例如,当我们发现 Cluster 3 的 Top Marker 是 MS4A1 (编码 CD20 蛋白)和 CD79A 时,所有证据都强烈指向这个群落是 B 细胞。
4. **可视化验证 (关键步骤):** 注释完成后,我们可以通过在 UMAP 图上可视化某个关键 Marker 的表达,来直观地验证我们的注释是否正确。

```
1 ## 示例：可视化B细胞的经典Marker：MS4A1 (CD20)
2 FeaturePlot(pbmc, features = "MS4A1", label = TRUE)
3 VlnPlot(pbmc, features = "MS4A1")
```

【动手练习与思考】

截图练习: 这一步是很好的练习机会,请您亲自运行代码,并将生成的 FeaturePlot / VlnPlot 截图保存下来,可以命名为 `Validation_MS4A1.png` 并思考如何将其插入到文档中。

12 生成最终细胞图谱

最后,我们将注释好的细胞身份赋予 Seurat 对象,并绘制出最终的“细胞身份地图”。

Table 1: 各细胞群落生物学注释汇总

Cluster ID	注释细胞类型	关键 Marker 基因及理由
0	记忆 CD4+ T 细胞	IL7R, LTB。IL7R (CD127) 是记忆 T 细胞的关键受体。
1	初始 CD4+ T 细胞	CCR7, LEF1, TCF7。这三个是公认的初始 T 细胞“三驾马车”。
2	CD14+ 单核细胞	CD14, LYZ, S100A9。是经典单核细胞的强烈信号。
3	B 细胞	MS4A1 (CD20), CD79A。B 细胞最特异、最核心的 Marker。
4	CD8+ T 细胞	CD8A, CD8B。CD8+ T 细胞身份的“金标准”。

Table 1 – continued from previous page

Cluster ID	注释细胞类型	关键 Marker 基因及理由
5	FCGR3A+ 单核细胞	FCGR3A (CD16), MS4A7。典型的非经典单核细胞特征。
6	NK 细胞	GNLY, GZMB, NKG7。高表达一系列强大的细胞毒性基因。
7	树突状细胞 (DC)	FCER1A, CLEC10A。经典的树突状细胞 Marker。
8	血小板	PPBP, PF4。血小板特异性趋化因子。

```

1 ## 创建新旧ID的对应关系并重命名
2 new.cluster.ids <- c("Memory CD4 T", "Naive CD4 T", "CD14+
    Monocyte", "B cell",
3                       "CD8 T cell", "FCGR3A+ Monocyte", "NK cell",
4                       "Dendritic Cell", "Platelet")
5 names(new.cluster.ids) <- levels(pbmc)
6 pbmc <- RenameIdents(pbmc, new.cluster.ids)
7
8 # 绘制最终带标签的UMAP图
9 DimPlot(pbmc, reduction = "umap", label = TRUE, pt.size = 0.5,
    label.size = 4) + NoLegend()

```

【技术要点】

保存最终成果: 千万不要忘记保存你辛勤工作的结晶！将处理好的最终 Seurat 对象保存下来，未来可以随时加载进行新的探索，无需从头再来。

保存最终处理好的 Seurat 对象

```

1 # 将最终的 pbmc 对象保存为 .rds 文件
2 saveRDS(pbmc, file = "pbmc3k_final_annotated.rds")
3
4 # 未来若要使用，只需运行以下代码即可加载：
5 # pbmc_final <- readRDS("pbmc3k_final_annotated.rds")

```

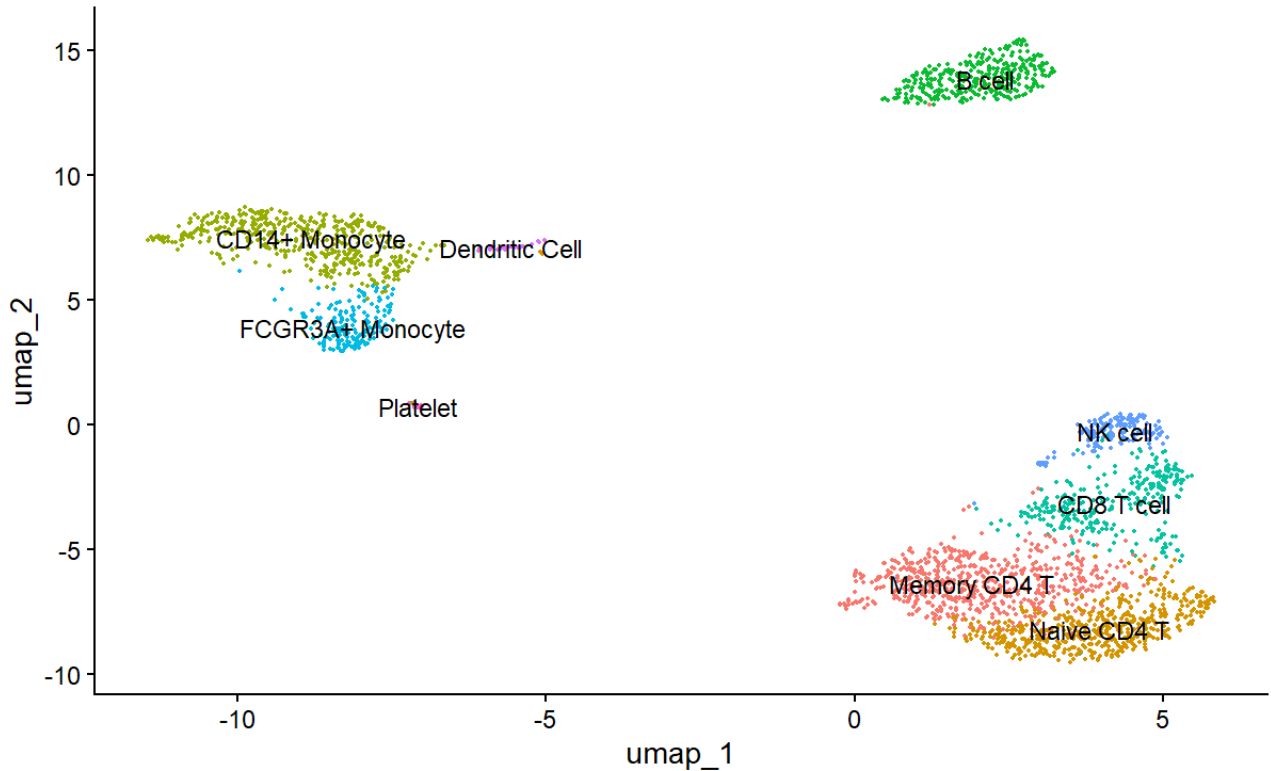


Figure 6: 最终细胞类型注释的 UMAP 图。该图是本次分析的核心成果，直观地展示了 pbmc3k 数据集中主要的免疫细胞亚群及其在转录组水平上的相对关系。

附录：完整分析 R 脚本 (v7.0 最终专业版)

scRNA-seq Seurat V5 标准流程脚本

完整 R 脚本

```

1 # --- 1. 环境设置与数据加载 ---
2 library(Seurat)
3 library(dplyr)
4
5 data_directory <- "03_data/filtered_gene_bc_matrices/hg19/"
6 pbmc.data <- Read10X(data.dir = data_directory)
7 pbmc <- CreateSeuratObject(counts = pbmc.data,
8                             project = "pbmc3k",
9                             min.cells = 3,
10                             min.features = 200)
11
12 # --- 2. 质量控制 (QC) 与数据净化 ---

```


(Continued)

```
13 # 计算线粒体基因比例
14 pbmc[["percent.mt"]] <- PercentageFeatureSet(pbmc, pattern = "^MT-")
15 # VlnPlot(pbmc, features = c("nFeature_RNA", "nCount_RNA",
16   "percent.mt"), ncol = 3)
17
18 # 根据QC图的结果进行过滤
19
20 # --- 3. 数据标准化 ---
21 pbmc <- NormalizeData(pbmc)
22
23 # --- 4. 特征选择与数据缩放 (标准流程) ---
24 pbmc <- FindVariableFeatures(pbmc, selection.method = "vst",
25   nfeatures = 2000)
26 # 仅对高变基因进行缩放, 这是最高效和推荐的做法
27
28 pbmc <- ScaleData(pbmc, features = VariableFeatures(object = pbmc))
29
30 # --- 5. 线性降维 (PCA) ---
31 pbmc <- RunPCA(pbmc, features = VariableFeatures(object = pbmc))
32 # ElbowPlot(pbmc) # 可视化以确定PC维度
33
34 # --- 6. 细胞聚类与非线性降维 (UMAP) ---
35 # 使用前15个PC进行下游分析
36 pbmc <- FindNeighbors(pbmc, dims = 1:15)
37 pbmc <- FindClusters(pbmc, resolution = 0.5)
38 pbmc <- RunUMAP(pbmc, dims = 1:15)
39
40 # --- 7. 寻找差异表达基因 (Marker Genes) 并保存结果 ---
41 pbmc.markers <- FindAllMarkers(pbmc, only.pos = TRUE, min.pct =
42   0.25, logfc.threshold = 0.25)
43 # 保存完整的marker列表
44 write.csv(pbmc.markers, file = "pbmc3k_all_markers.csv", row.names
45   = FALSE)
46 # 筛选并保存每个cluster的top15 markers
47 top15.markers <- pbmc.markers %>%
48   group_by(cluster) %>%
49   slice_max(n = 15, order_by = avg_log2FC)
50 write.csv(top15.markers, file = "top15_markers_per_cluster.csv",
51   row.names = FALSE)
52
53 # --- 8. 重命名细胞群落并最终可视化 ---
54 new.cluster.ids <- c("Memory CD4 T", "Naive CD4 T", "CD14+
55   Monocyte", "B cell",
56   "CD8 T cell", "FCGR3A+ Monocyte", "NK cell",
57   "Dendritic Cell", "Platelet")
58 names(new.cluster.ids) <- levels(pbmc)
59 pbmc <- RenameIdents(pbmc, new.cluster.ids)
```


(Continued)

```
54 # 绘制最终UMAP图
55 # DimPlot(pbmc, reduction = "umap", label = TRUE, pt.size = 0.5) +
    NoLegend()
56
57 # --- 9. 保存最终的Seurat对象 ---
58 saveRDS(pbmc, file = "pbmc3k_final_annotated.rds")
```