

涉密论文 ☐ 公开论文 ☐

浙江大學

# 本科生毕业论文(设计)



# 题目 模块化易部署的生物信息学分析平台的搭建

姓名与学号 夏涵 3140105252

指导教师 陈 铭

年级与专业      **14 级**      生物科学

所在学院 生命科学学院

提交日期 \_\_\_\_\_

---

## 致 谢

首先我要感谢浙江大学生命科学学院陈铭教授提供给我在我所感兴趣的生物信息学实验室进行我本科生阶段的毕业设计的机会以及指导性意见。其次我要感谢浙江大学生物系信息学实验室博士生周银聪师兄，感谢他在整个毕设期间不厌其烦地给我悉心指导。我还要感谢本科阶段生物学课程的授课教师的付出。最后我要感谢实验室其他师兄师姐在我毕业设计过程中给予我的帮助以及修改意见。

## 摘 要

不论是单细胞还是整个组织，使用 RNA 测序（RNAseq）进行的转录谱分析已经成为量化全基因表达模式的有力方法。有关 RNA-seq 数据分析的软件层出不穷。然而对于计算机知识背景相对薄弱的生物研究人员来说，在使用相关生信工具时却面临着安装配置困难、重现性差等问题，同时对于 RNA-seq 数据分析初学者来说，如何选择一套合适的 RNA-seq 软件工具也是一个棘手的问题。EZRS（Easy RNA-seq）是一款基于 Docker 容器技术，整合了一套完整 RNA-seq 数据分析流程的网页 app。用户可通过下载 Docker 镜像文件将 EZRS 部署在自己服务器上，实现开发环境复现并通过友好的交互页面进行 RNA-seq 数据分析。此外，用户还可以通过 Table Manager 以及 START 模块对 RNA-seq 数据分析结果进行可视化分析。

**关键词：**RNA-seq, Docker, Shiny

---

## Abstract

Transcriptional profiling using RNA sequencing (RNAseq) has become a powerful method for quantifying whole gene expression patterns in various contexts from single cells to whole tissues. There is an endless stream of software for analyzing RNA-seq data. However, for biological researchers with a relatively weak computer knowledge background, they are faced with difficulties in installation and configuration and poor reproducibility when using related bioinformatics tools. At the same time, how to choose a set of right RNA-seq software for beginners is also a thorny issue. EZRS (Easy RNA-seq) is a web app that integrates a complete RNA-seq data analysis process based on Docker container technology. Users can deploy the EZRS on their own server by downloading Docker image files to achieve reproduction of the development environment and conduct RNA-seq data analysis through friendly interaction pages. In addition, users can visually analyze the results of RNA-seq data analysis through the Table Manager and START modules.

**Key words:** RNA-seq, Docker, Shiny

---

# 1 引言

## 1.1 RNA-seq 简介

转录组是特定的发育阶段或生理条件下，细胞中完整的一套转录物的定量和定性描述。了解转录组对于解释基因组的功能元件、揭示细胞和组织的分子成分以及了解发育和疾病至关重要。

研究人员已经开发出各种各样的技术用于转录组的定性与定量研究，例如基于杂交原理的荧光标记 cDNA 微阵列技术。虽然这项技术具有高通量的特性且开销较小，但是它仍有许多缺点：它依赖于已知基因组序列；由于交叉杂交导致的高背景水平；以及由于信号的背景和饱和度而导致的有限的动态检测范围。除此之外，微阵列所得到的数据在被用于不同样本间比较前，还需要进行较为复杂的标准化过程。

相对于基于杂交的微阵列技术，基于测序的方法可以较为直接地确定 cDNA。新一代测序（NGS）技术已经彻底改变了转录组研究的进展。新开发的深度测序技术可以以较高效率获取关于转录本生物学的定量和定性信息。通过测量样本中所有基因的 mRNA 水平的 RNA-seq 技术已然成为了探究生物全局转录组变化的利器。

为了产生 RNA-seq 的数据，首先要从 RNA 样品中提取全套的 mRNA，然后破碎并逆转录成 cDNA 文库。这些短片段的 cDNA 通过聚合酶链式反应扩增并通过机器测序，产生数百万短的 reads。之后这些 reads 被匹配到参考基因组或参考转录本上。研究人员可以通过对特定基因区域内匹配到的 reads 数量的测量来了解该基因的表达丰度。这些 reads 还可以在没有已知基因组序列的情况下重新拼接以创建新的转录本。

与提供有限的基因调控信息的微阵列技术相比，RNA-seq 提供更加全面的转录组信息。RNA-seq 在基因表达分析的定性定量测量方面取得了许多重大改进，拥有多角度分析检测的能力：外显子，SNP 水平的表达；剪接作用；跨整个基因的转录后 RNA 编辑；亚型和等位基因特异性表达<sup>[1]</sup>等。

## 1.2 生物信息学工具局限性

随着第二代测序技术的普及，生物学数据量不断增加，曾经可以手动处理的生物学数据现如今越来越依赖于生信软件自动处理。一个简单的生信软件就能处理大批量生物数据集，若生物学家能熟练掌握运用生信工具，他们的效率将会得到大大地提升。越来越多的生信工具被开发出来用来进行学术研究以及被发表刊登在学术期刊上。

现今的生物信息学工具面临着以下几个严峻的问题：

### 已开发工具获取问题

往往开发这些工具的开发人员片面地追求发表文章的数量，而对已经开发并发表的工具维护不到位，致使后期试图使用这些工具进行研究的生物学家，只能面对着一堆有问题、缺乏一致性、混乱的工具而束手无策。对于某些已发表的工具来说，如果期刊不使用第三方服务器去管理这些工具的话，读者只能通过联系作者来获取工具。若无法与作者取得联系，此工具的存在将毫无意义。此外，如果开发人员自行运维工具，如果资助方停止资助服务器的开销，亦或者服务器迁址都将导致工具的获取问题。与此相关的一项研究表明，在许多情况下，生物信息学网址通常在描述它的文章发表后二至三年内变为不可用状态<sup>[2]</sup>。

### 工具的配置安装困难

使用者的操作系统、软件、硬件不同，以及开发人员所写的安装配置帮助文件注释不清等问题都将导致工具安装配置困难。比如，研究人员想要使用一款用 C/C++ 开发的工具。然而，在正常使用之前，研究人员可能需要预先下载一些编译工具对工具进行编译。即使一些基于 R、Python 或 Java 开发的软件与平台无关，但安装之前又需要各种预安装的依赖存在。以上问题对于一些无计算机背景的生物学家来说是一项极大的挑战。在亚利桑那大学的研究人员的一项研究中发现，在已开发的生信工具中只有不到 50% 的软件可以成功地安装，这一现象仍在被很多开发人员重复着<sup>[3]</sup>。

### 文件上传问题

随着二代测序数据分析的普及，越来越多的科研人员进入了相关领域进行学习，一些生物信息学云平台应运而生（如 Galaxy<sup>[4]</sup>）。这些平台收集并筛选了大

量的参考数据，整理出了用来满足研究人员各种各样需求的工具，以其强大的计算能力（相对于 PC 而言）给生信工作者带来了不少方便。随着数据量的不断增加，原始数据上传至云平台耗时代价巨大，将大量数据上传至云平台计算将不再是一个最佳选择。

### “代码腐烂”问题

软件依赖不是静态元素，它将定期接收维护人员更新和修复错误，被添加新功能或弃用旧功能（甚至自己的整个依赖关系）。任何这些更改都可能会更改代码生成的结果。由于其中一些更改可能确实可以有效地解决的错误或早期的基础代码问题，因此通常复现原始结果困难重重。一些工作流程软件，虚拟机，持续集成服务以及软件开发的最佳实践等技术解决方案将很大程度上解决许多经常令人不快的复现性问题。然而，研究人员在学习这些不属于他们专业领域的工具和方法方面面临重大障碍，配置相关环境使他们无法集中于生信工作。

## 1.3 EZRS 简介

Docker<sup>[5]</sup>是一种基于 Linux 的开源的容器技术，它每天都会被成千上万的用户使用，可将工具与主机服务器的操作系统隔离开来，允许用户以快速且可重现的方式部署工具。不同的容器可以运行不同版本的相同工具或库，而他们之间不会存在任何干扰。

基于以上问题，我们开发了一款基于 Docker 技术的简单易部署的 RNA-seq 网页 app——EZRS，用于帮助计算机背景不足而又有生信数据分析需求的生物工作研究人员进行 RNA-seq 数据分析、理解 RNA-seq 分析过程。EZRS 提供友好的交互界面，使用者只需要遵循简单地流程便可以生成一系列常用 RNA-seq 分析结果数据。同时，使用者可以使用网页中的 Table Manager 简单处理生成的结果表，并可作为数据输入传入已被整合进此网页的 START<sup>[6]</sup>（基于网页的 RNA-seq 数据可视化工具）中进行可视化。

## 2 工具及方法

EZRS 整合了 8 个常见的 RNA-seq 数据分析工具供用户使用。同时整合了 START 为用户提供获得良好的可视化。自开发了 Tabel Manager 工具，供用户简



单地加工分析获得的数据成可直接供 **START** 可视化的数据。整个 **Web App** 使用基于 **R** 语言的 **Shiny** 框架开发，在 **Docker** 容器中运行。**EZRS** 整体架构如图 2.1 所示。



图 2.1 EZRS 架构总览

2.1 RNA-seq 数据分析工具

2.1.1 fastq-dump

“sra”文件是 NCBI 推出的存储来自 NGS（包括 Illumina, 454, IonTorrent, Complete Genomics, PacBio 和 OxfordNanopores）高通量数据的格式，可通过 NCBI SRA toolkit 软件进行格式转换。文章中会提供 NCBI GEO 数据库 (<https://www.ncbi.nlm.nih.gov/geo/>) 的登录号，登陆后通过 `ftp` 即可下载到文章

作者所使用的测序数据。根据测序方法的不同（单端测序/双端测序），通过 `fastq-dump` 可以实现将 “.sra” 文件转换为做 RNA-seq 数据分析时更常用的 “.fastq” 格式。EZRS 中关于此工具，主要使用以下两条命令：

```
1. # single-end 单端测序
2. $ fastq-dump file_name.sra -o ./fastq_results
3. # pair-end 双端测序
4. $ fastq-dump -split-3 file_name.sra -o ./fastq_results
```

### 2.1.2 FastQC

FastQC<sup>[7]</sup>是一款基于 Java 的软件，一般都是在 linux 环境下使用命令行运行，它可以快速多线程地对测序数据进行质量评估（Quality Control）最终将产生以下方面的测评结果：总结信息、基本信息、序列测序质量统计、每个 tail 的测序情况、每条序列的测序质量统计、GC 含量统计、序列平均 GC 含量分布图、序列测序长度统计、序列 Adapter、重复短序列。EZRS 中关于此工具，主要使用以下命令生成 pdf 格式的质量评估报告：

```
1. $ fastqc -o ./fastqc_result -f fastq file_name.fastq
```

### 2.1.3 fastx\_trimmer & fastq\_quality\_filter

根据用户的需求，在此质量控制环节，用户可以指定截去 reads 的头部的碱基个数，以及筛选达到指定得分线的 reads。EZRS 中关于此工具，主要使用了以下命令：

```
1. # trimmer
2. $ fastx_trimmer -Q 33 -f 12 -I file_name.fastq -o ./fastq_results/trimmed.fastq
3. # filter
4. $ fastq_quality_filter -Q 33 -q 20 -p 80 -i trimmed.fastq -o ./fastq_results/filtered.fastq
```

### 2.1.4 HISAT2

HISAT2<sup>[8]</sup>是一种用于将新一代测序 reads（DNA 和 RNA）映射到人类基因组（以及单个参照基因组）的快速且灵敏的比对程序，设计并实现了图形 FM 索引（GFM）。除了使用代表人类基因组的一个全球 GFM 索引之外，HISAT2 还使用了很大一组覆盖整个基因组小的 GFM 索引（每个指数代表 56 Kbp 的基因组区

域，其中 55,000 个指标覆盖人类人口)。这些小型索引（称为本地索引）结合几种比对策略，可实现测序 reads 的快速准确对齐。这种新的索引方案被称为分层图形 FM 索引（HGFM）。RNA-Seq 基因组比对工具 HISAT2 是 TopHat2/Bowtie2 的继任者，使用改进的 BWT 算法<sup>[9]</sup>，实现了更快的速度和更少的资源占用。

在本 EZRS web app 中，为了方便用户操作，取消了使用 hisat2-build 提取目标信息（如外显子或可变剪接）建立索引的步骤，用户可以自由选择四种常用的参考基因组信息（人类、小鼠、果蝇、拟南芥）作为参考基因组进行比对。

EZRS 中关于此工具，主要使用以下命令：

```
1. # 单端测序结果
2. $ hisat2 -p 2 --dta -x ./index_files/index_file_prefix -U ./fastq_results/filtered.fastq -S file_name.sam
3. # 双端测序结果
4. hisat2 -p 2 --dta -x ./index_files/index_file_prefix -1 ./fastq_results/filtered_1.fastq -
   2 ./fastq_results/filtered_2.fastq -S file_name.sam
```

### 2.1.5 samtools

SAM（序列比对/映射）格式是用于存储大核苷酸序列比对的通用格式。SAM 工具提供了各种实用程序来处理 SAM 格式的比对，包括按位置格式进行排序，合并，索引和生成比对。

EZRS 中关于此工具，主要使用 samtools<sup>[10]</sup>将 sam 格式文件转换为二进制的易储存的 bam 格式的功能，并同时进行排序：

```
1. $ samtools sort -@ 2 -m 200M -o file_name.bam file_name.sam
```

### 2.1.6 stringtie & gffcompare

StringTie<sup>[11]</sup>是一个能够将 RNA-Seq 比对结果拼接成潜在转录本的快速且高效的整合工具。它使用新颖的网络流动算法以及可选的从头装配步骤来组装和定量表示每个基因座的多个剪接变体的全长转录物。

EZRS 中关于此工具，主要使用了其转录本拼接，整合以及计算 FPKM 功能，gffcompare 主要用来进行转录本注释文件比较：

```
1. # 转录本拼接
2. $ stringtie -p 2 -G ./index_files/file_name.gtf -o file_name.gtf -l file_name ./bam_files/file_name.bam
```

```
3. # 整合 gtflist.txt 中包含需要整合的转录本的 gtf 文件路径
4. $ stringtie --merge -p 2 -G ./index_files/file_name.gtf -o stringtie_merged.gtf gtflist.txt
5. # 转录本注释文件比较
6. $ gffcompare -r ./index_files/file_name.gtf -G -o merged ./stringtie_merged.gtf
7. # 计算 FPKM
8. $ stringtie -e -p 2 -G ./stringtie_merged.gtf -A genes.gtf -o transcripts.gtf ./bam_files/file_name.bam
```

### 2.1.7 HTSeq-count

HTSeq<sup>[12]</sup>是一个 Python 软件包,提供各种工具处理来自高通量测序分析的数据。一个常用的功能是输入比对后的 bam 文件以及一个基因组 gtf 文件,计算有多少读数映射到每一个特征区域上。EZRS 中关于此工具,主要使用了 HTSeq-count 功能进行针对每一个基因的 reads 计数:

```
1. $ htseq-count -q -f bam -s no -
   i gene_name ./bam_files/file_name.bam ./index_files/file_name.gtf > file_name.count
```

## 2.2 Shiny 交互网页开发包

Shiny<sup>[13]</sup>是 R<sup>[14]</sup>中的一个 Web 开发框架,使得 R 的使用者不必完全了解 css、js,而只需要了解一些 html 的知识就可以快速完成 web 开发,且 shiny 包集成了 bootstrap、jquery、ajax 等特性,极大解放了作为统计语言的 R 的生产力,使得非传统程序员的 R 使用者不必依赖于前端、后端工程师就可以自己依照业务完成一些简单的数据可视化工作,快速验证想法的可靠性。

开源 Shiny Server 提供了一个平台,可以在该平台上将多个 Shiny 应用程序托管在单个服务器上,每个服务器都有自己的 URL 或端口。在 EZRS 开发过程中,所使用的 shiny 框架分为 ui.R 和 server.R 两个部分,接下来将会分别从这两个方面进行介绍。

### 2.2.1 Shiny ui.R

在 ui.R 部分使用了 dashboard R 包,界面部分主要包含网页页眉(dashboardHeader),导航边栏(dashboardSidebar)以及工作窗口(dashboardBody)部分。Dashboard 提供了十分优雅的交互界面的优雅框架,用户只需要遵循简单的 dashboard 框架,即可在框架中添加自己网站的元素。EZRS 中页眉部分只包括 app 名称,没有设计其按键,网页主要元素都包含于边栏和工作窗口部分。

```
1. ## ui.R ##
2. library(shiny)
3. library(shinydashboard)
4. shinyUI(
5.   dashboardPage(
6.     dashboardHeader(),
7.     dashboardSidebar(
8.       sidebarMenu(menuItem())
9.     ),
10.    dashboardBody(
11.      tabItems(tabItem())
12.    )
13. )
```

左侧边栏部分，按照用户的操作习惯，从上往下依次设置了：创建工作目录按钮，一键 RNA-seq (One Click)，分步处理 (Step by Step)，结果表格处理 (Table Manager)，START 可视化 (START) 元素。其中分步处理和 START 可视化部分又包含子菜单。分布处理子菜单按照 RNA-seq 的流程提供了一系列导航链接；START 可视化子菜单提供了一套不同的可视化功能的导航链接。

在工作窗口部分根据边栏导航元素 item 的 id 不同，可对不同 item 所对应的工作窗口界面进行设计。窗口中主要由文本输入 (textInput())、下拉菜单 (selectInput())、勾选框 (checkboxInput()) 以及操作按钮 (actionButton()) 组成，用于将从界面部分获得的用户输入传递至 server.R 后台进行逻辑处理。同时，表格输出 (tableOutput())、文本输出 (verbatimTextOutput()) 又从后台接受逻辑输出将信息返回给用户。

START 中的 ui 部分通过 source 函数直接将源码应用到了 ui.R 相对应位置中。

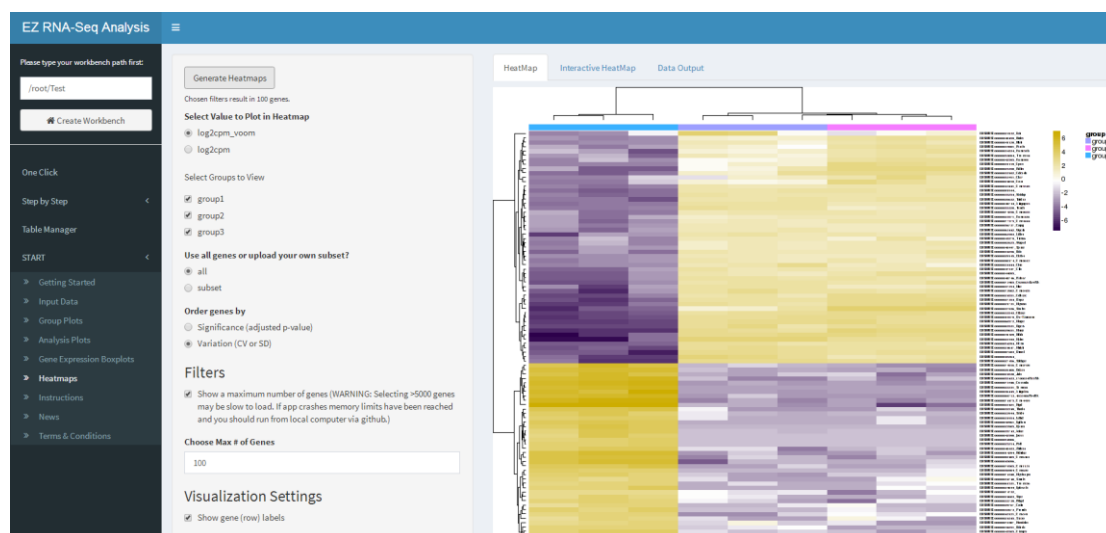


图 2.3 EZRS 界面

## 2.2.2 Shiny server.R

Shiny server 部分主要通过 `input$arguments` 来接受来自 `ui` 部分传进来的参数。One Click 部分设计了两个函数，分别针对单端测序和双端测序的一键处理功能，每一个函数中包含有相应的一套 RNA-seq 流程。`observe` 函数在用户产生输入时，即会更新执行 `observe` 函数中的代码。`observeEvent` 函数用于实现观测是否存在相应的操作按钮输入，若用户点击相应操作按钮，则执行函数内部逻辑，负责启动主要的 RNA-seq 流程。`output` 函数将获取函数内操作的输出并且传递到前端部分，用于展示表格以及系统命令的输出结果。`system` 函数用于执行 RNA-seq 工具的相关执行命令。

```

1.  ## server.R ##
2.  library(shiny)
3.  function_name <- function(){
4.    shinyServer(function(input, output, session) {
5.      observe({ })
6.      observeEvent(input$ab1, {
7.        system()
8.      })
9.      output$value0 <- renderText({})
10. }

```

“Step by Step”和以及“One Click”功能模块中的各个 RNA-seq 工具所产生的

中间文件都将会被保存在名称相对应的工作目录下（如下图 2.4），方便研究人员分析中间数据以及学习 RNA-seq 数据分析流程。每一个工具所需的数据会根据需要从上级步骤中自动读取。Table Manager 模块所产生的结果表会保存在 table\_manager 文件夹目录下（如下图 2.5）。

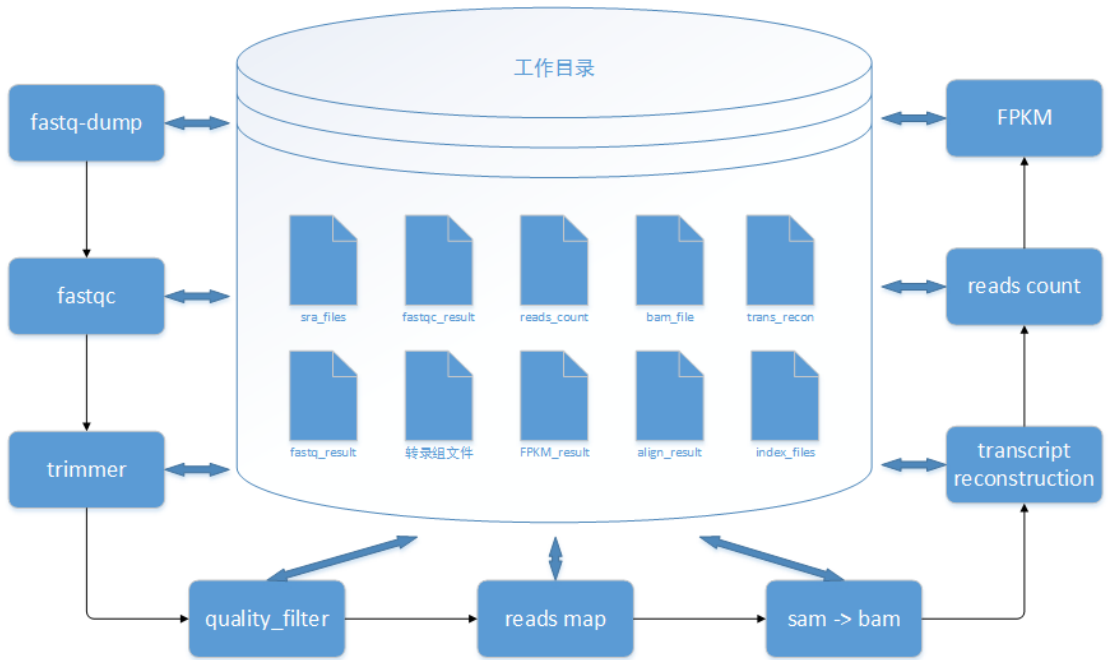


图 2.4 RNA-seq 分析软件数据流图

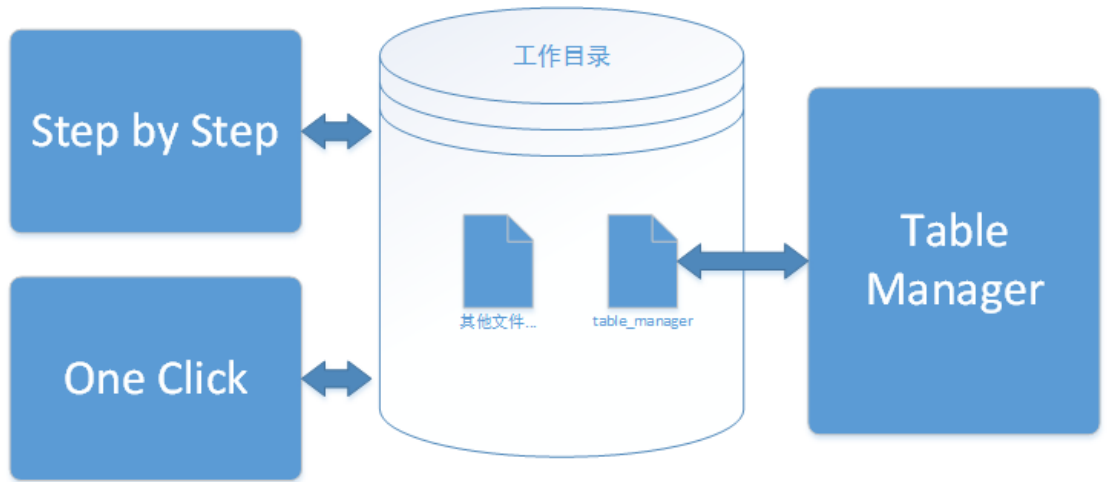
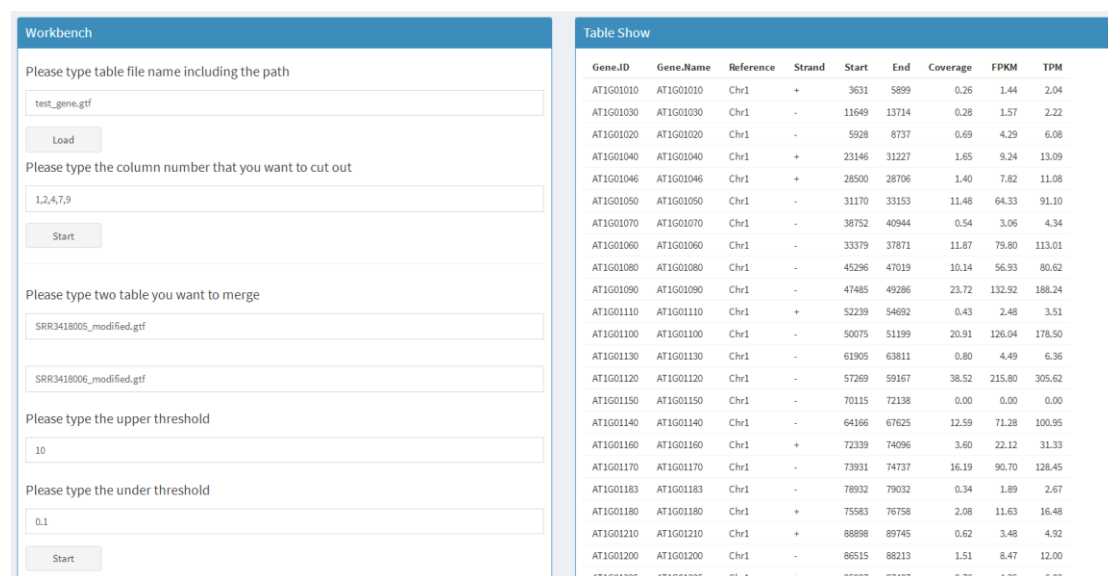


图 2.5 EZRS 模块数据流图

START 中的 server 部分通过 source 函数直接将源码应用到了 server.R 相对应位置中。

## 2.3 Table Manager 工具

Table Manager 是自主开发的基于 R 语言的整合于 EZRS 中的建议结果表格处理工具。经过前几步的 RNA-seq 数据分析之后，用户将获得最终的 reads 计数和 FPKM 数据。Table Manager 提供简单的表格数据处理功能，包括在线表格预览、删除表格多余列、合并表格、差异表达基因过滤功能。最后产生的数据格式可直接用于 START 可视化分析。



Gene.ID	Gene.Name	Reference	Strand	Start	End	Coverage	FPKM	TPM
AT1G01010	AT1G01010	Chr1	+	3631	5899	0.26	1.44	2.04
AT1G01030	AT1G01030	Chr1	-	11649	13714	0.28	1.57	2.22
AT1G01020	AT1G01020	Chr1	-	5928	8737	0.69	4.29	6.08
AT1G01040	AT1G01040	Chr1	+	23146	31227	1.65	9.24	13.09
AT1G01046	AT1G01046	Chr1	+	28500	28706	1.40	7.82	11.08
AT1G01050	AT1G01050	Chr1	-	31170	33153	11.48	64.33	91.10
AT1G01070	AT1G01070	Chr1	-	38752	40944	0.54	3.06	4.34
AT1G01060	AT1G01060	Chr1	-	33379	37871	11.87	79.80	113.01
AT1G01080	AT1G01080	Chr1	-	45296	47019	10.14	56.93	80.62
AT1G01090	AT1G01090	Chr1	-	47485	49286	23.72	132.92	188.24
AT1G01110	AT1G01110	Chr1	+	52239	54692	0.43	2.48	3.51
AT1G01100	AT1G01100	Chr1	-	50075	51199	20.91	126.04	178.50
AT1G01130	AT1G01130	Chr1	-	61905	63811	0.80	4.49	6.36
AT1G01120	AT1G01120	Chr1	-	57269	59167	38.52	215.80	305.62
AT1G01150	AT1G01150	Chr1	-	70115	72138	0.00	0.00	0.00
AT1G01140	AT1G01140	Chr1	-	64166	67625	12.59	71.28	100.95
AT1G01160	AT1G01160	Chr1	+	72339	74096	3.60	22.12	31.33
AT1G01170	AT1G01170	Chr1	-	73931	74737	16.19	90.70	128.45
AT1G01183	AT1G01183	Chr1	-	78932	79032	0.34	1.89	2.67
AT1G01180	AT1G01180	Chr1	+	75583	76758	2.08	11.63	16.48
AT1G01210	AT1G01210	Chr1	+	88898	89745	0.62	3.48	4.92
AT1G01200	AT1G01200	Chr1	-	86515	88213	1.51	8.47	12.00
AT1G01225	AT1G01225	Chr1	+	95987	97407	0.76	4.25	6.02

图 2.3 图片左侧窗口为工作区，右侧窗口提供表格预览

由于 RNA-seq 功能将产生大量的基因 FPKM 数据，若将所有的基因数据进行可视化，最终得到的可视化结果可能并不理想。所以，差异表达基因过滤功能将会根据用户所选择的阈值来过滤产生差异表达较大的基因。在 server 端，Table Manager 首先将对照组和实验组的由 stringtie 工具生成的 FPKM 数据删除多余列，只保留 Gene ID 和 FPKM 值并进行按照基因名合并。之后将所有 FPKM 值加 1（防止 log 之后出现负值，而加 1 之后并不会影响观测差异表达的基因）取



$\log_2$ 。之后删除所有对照组和实验组 FPKM 列值之和小于 1 的行（防止在下一步实验组和对照组 FPKM 值之比中，由于基因表达量  $\log$  后接近零而出现的伪差异表达现象）。最后，根据用户输入的阈值，筛选实验组和对照组 FPKM 值之比满足需求的基因，生成表格。

## 2.4 Docker 容器

### 2.4.1 Docker 简介

Docker<sup>[15]</sup>是一个轻量、简单、快速,用来创建、移植、运行各种 APP 的开源应用容器引擎。用户可以将任何应用程序连同它的依赖项打包、上传，以供其他用户在虚拟环境下轻松配置部署应用程序。Docker 包括三个核心的结构：镜像、容器以及库。镜像文件可以存储包括数据库之类复杂的应用，以待用户添加储存数据。容器则是提供给不同用户相似的应用程序运行环境，以实现结果的复现性。当一款应用已经存在于 Docker 的容器中后，用户将不再需要设置和维护软件以适应不同的语言环境或者工具。Docker Hub 允许用户查找、管理以及从社区、官方或者私人渠道提取镜像文件，同时用户可以完全免费使用公共库。Docker Hub (<https://hub.docker.com/>) 是 Docker 基于云的注册表服务，基于 Docker 的镜像和容器的整个开发流程中的工作流自动化为用户提供了很大的便利<sup>[16]</sup>。

应用程序可以使用 Docker 技术独立运行容器，并且每个管理命令（启动，停止，启动等）可以在数秒或数毫秒内执行。成千上万的容器可以同时单个主机上运行<sup>[17]</sup>，从而确保一项任务的失败不会导致整个过程的中断：新容器可能会被快速初始化继续完成整个过程，从而提高整体效率。

Docker 的简单脚本 Dockerfile（类似于 Makefile）定义了如何构建镜像。使用比其他配置工具（例如 Chef, Puppet, Ansible）或持续集成（CI）平台（例如 Travis CI, 可发布 CI）更简单的语法；用户只需要熟悉 shell 脚本和 Linux 分发软件环境（例如基于 Debian 的 apt-get）即可开始编写 Dockerfiles。虽然工具的镜像文件可能非常大，但 Dockerfile 只是一个很容易存储和共享的小型纯文本文件。Dockerfile 包含所有软件依赖关系，直到操作系统级别，并由 Docker

构建工具构建，使构建在不同机器上的最终构建结果不太可能不同。其他用户可直接通过编辑脚本来扩展或定制镜像文件。

由于 Docker 将软件环境定义为特定的操作系统和一系列库（如 Ubuntu 或 Debian 发行版），因此可以显著减少“代码腐烂”问题。在发行时使用分阶段发布模型，其中包含稳定，测试和不稳定的阶段，并且进行大量测试以发现潜在的问题<sup>[18]</sup>，同时还为每个阶段的软件定期提供安全更新。

生物信息学工具可以使用 Docker 封装来开发可移植，易复现的工作流。Docker 技术允许应用程序使用一个自给自足的包独立运行，因此十分适合生物信息学研究人员使用。该软件包可以在各种计算平台上以便携的方式高效地分发和执行<sup>[19]</sup>。至今，大量的基于 Docker 的生物信息学工具使用 Perl、BioPerl、python、BioPerl、R 语言已经被开发出来供学者使用。最为知名的 Galaxy 平台也开发出 Docker Galaxy。Docker 技术能够在不需要额外配置的情况下在不同环境中执行相同的功能和服务<sup>[20]</sup>，从而以高效率创建可复现的工具。因此研究人员只关注从获取的序列中挖掘信息，而不是确定如何安装和使用软件。

## 2.4.2 EZRS 中 Docker 的应用

EZRS 开发过程，在一个 ubuntu16.04 系统并配置好 shiny-server 的 Docker 容器中，保证了容器间互相隔离的开发状态。开发过程中，根据需求不断安装 R 包、RNA-seq 分析工具等。随着开发的进行也产生了大量废弃的 R 包。在最终产品完成测试后，根据需求重新编写了 Dockerfile，去除了多余的 R 包，使整个 Dockerfile 更加精简。用户可以方便地根据安装配置章节中的指导，选择从 Docker 仓库中直接安装我们已经 build 完成的镜像，也可以选择通过 Dockerfile 重新 build 镜像。

## 2.5 START 可视化 App

### 2.5.1 START 简介

START 应用程序是一个使用 Shiny 框架完全采用开源 R 编程语言编写基于

网络的应用程序。它实现了完全跨平台，可以从安装有 **R** 的任何计算机本地启动。或者用户可以在本地或远程服务器上使用他们的转录组数据来托管他们自己的应用程序，以便其他用户可以在不安装 **R** 或应用程序的情况下从网站访问应用程序和数据。**START** 应用程序已经在 **Chrome** 浏览器环境中得到最广泛的测试。用户可以通过防火墙或基于 **Web** 的认证服务来保护他们的数据。

**START App** 可以轻松上传和查看 **RNAseq** 数据，提供多种类型的可视化，可产生直接用于发表论文的图片 and 直观的基于 **Web** 的图形用户界面。此外，**START App** 可作为基于 **Web** 的工作流程的一部分——从原始 **RNA** 序列数据到与 **BrowserGenomes.org** 结合使用<sup>[21]</sup>，而不需要计算生物学家用于直接研究设计的实验。因此，即便生物学家计算机处理数据方面能力受限，他们也可以很轻松的调整数据结构，上传至 **START App**，即可以得到令人满意的可视化结果。



图 2.6 START 原界面

## 2.5.2 START 的整合

原版 **START** 的网页页眉导航栏被内嵌至 **EZRS** 的导航边栏中，原版的工作区域被内嵌至 **EZRS** 的 **dashboardbody** 中。

### 3 结果与讨论

#### 3.1 结果综述

EZRS 是我们开发的一款可供计算机经验相对欠缺的生物研究人员进行 RNA-seq 数据分析的、设计友好的、简洁易部署的 Web App。用户可选择从 Docker Hub 上直接安装已 build 好的 EZRS 镜像文件,也可选在通过 Dockerfile 重新 build EZRS。EZRS 提供两种数据分析模式——“Step by Step”分步处理数据,或者“One Click”模块进行一键处理护具。对于 stringtie 工具产生的 FPKM 数据,用户可以轻松地通过 Table Manager 功能筛选差异表达基因,并且生成可直接用于 START 可视化的数据。操作流程概览请参照图 3.1。

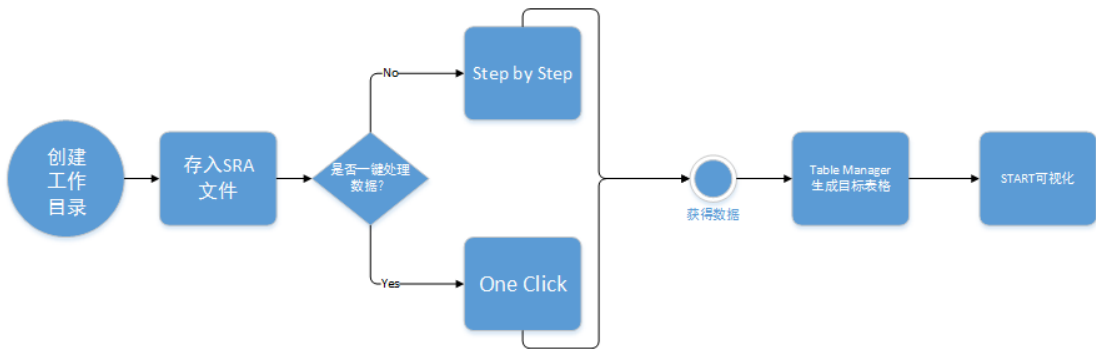


图 3.1 操作流程图

#### 3.2 案例展示

本案例使用的测序数据来自 NCBI GEO，检索号为 GSE80568，测序平台为 HiSeq 2500，单末端测序（SE）。数据集包括拟南芥幼苗的 RNA-seq 和 ChIP-seq 数据，原用于分析构建拟南芥抗渗透压胁迫下的转录调控网络，考虑到运行时间的问题，本案例只选取 GSE80568 中的对照组与实验组两组数据，无生物学重复。其中实验组为脱落酸处理 8 小时的拟南芥幼苗，对照组为乙醇处理 8 小时的拟南芥幼苗。具体样本信息如下表所示：

表 3.1 案例样本信息

组别	实验处理	SRA 编号	文件大小（GB）
实验组	ABA, 8h	SRA3418005	1.8
对照组	EtOH, 8h	SRA3418006	1.6

基因组数据来自 EZRS 提供的拟南芥基因组，来自拟南芥数据库 TAIR，包括拟南芥染色体基因组序列和注释文件。

若使用“**One Click**”功能，在以下界面的“sra 文件数量”参数栏填写 2，按照默认参数进行数据处理（参数设置如下图），最终将在工作目录下各个文件夹中生成相应文件。

One Click

Please type the .sra files path

/root/Test/sra\_files

Please type the .sra files num

2

Please choose if pair-end

unique

Please type the trim start position

12

Please type the trim Q value

33

Please type the fileter q value

20

Please type the fileter p value

80

Please type the fileter Q value

33

Please choose your sample genome type

arabidopsis

Please type the thread num

2

Please type the memory space

200M

Start

图 3.2 One Click 参数功能展示

若使用“**Step by Step**”功能（如图 3.3.a），按照菜单栏中的顺序依次处理数据。根据界面参数设置提示进行设置，某些步骤中包含一些高级参数设置，可以单击“**Advanced**”按钮展开进行参数设置（如图 3.3.f 红框内为高级设置区域）。点

击“Start”按钮，按钮下方将会出现“processing”字样（如图 3.3.b），表示程序正在后台处理数据，请稍后。处理完毕时，若顺利完成，提示框中将会显示“OK”（如图 3.3.d），若出现错误提示框中将会显示“Error”（如图 3.3.c），请重新检查文件参数是否正确。其中，Fastqc 工具还将生成一份质检报告，用户可以在点击“Click here to check the QC\_result”查看报告（如图 3.3.d）。

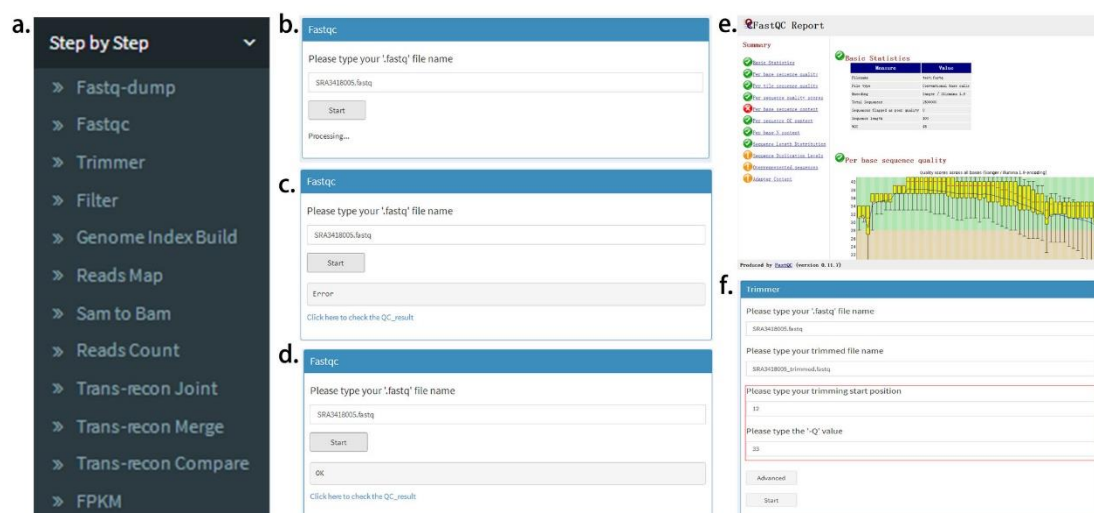


图 3.3 部分功能展示 (a).Step by Step 模块子菜单展示。(b).正在处理状态展示。(c).处理错误状态展示。(d).处理正确状态展示。(e).fastqc 质检报告展示。(f).高级功能菜单展示。

如果按照步骤运行（由于本次生物学重复为 1，且不期望得到新的转录本信息，故没有进行转录组重建的三个步骤），最终将会产生 reads 数量统计和 FPKM 两个可分析文件。

我们选取 FPKM 文件进行下一步分析。将在“FPKM\_result”文件夹中产生两个文件 SRR3418005\_gene.gtf 和 SRR3418006\_gene.gtf。在“Table Manager”模块中，首先对每一个表格进行加工处理，去掉多余列。如图所示，首先读取 SRR3418005\_gene.gtf（如图 3.4.a），发现我们只需要保留第一和第八列，故在第二栏中删除其他列。处理后将生成 SRR3418005\_modified.gtf，可见只保留了目标列（如图 3.4.b）。两个文件经过相同步骤处理后，便可以进行下一步合并步骤。

在合并栏中，首先输入两个需要合并的文件名，此次为 SRR3418005\_modified.gtf、SRR3418006\_modified.gtf。然后选择筛选阈值（具体

含义详见第二章“Table Manager”部分), 程序将自动按照 Gene ID 进行合并, 同时筛选满足阈值的差异表达基因。最终生成的 result.csv 文件可直接提交至“START”模块进行可视化(可视化部分的介绍可参考“START”模块中的介绍)。

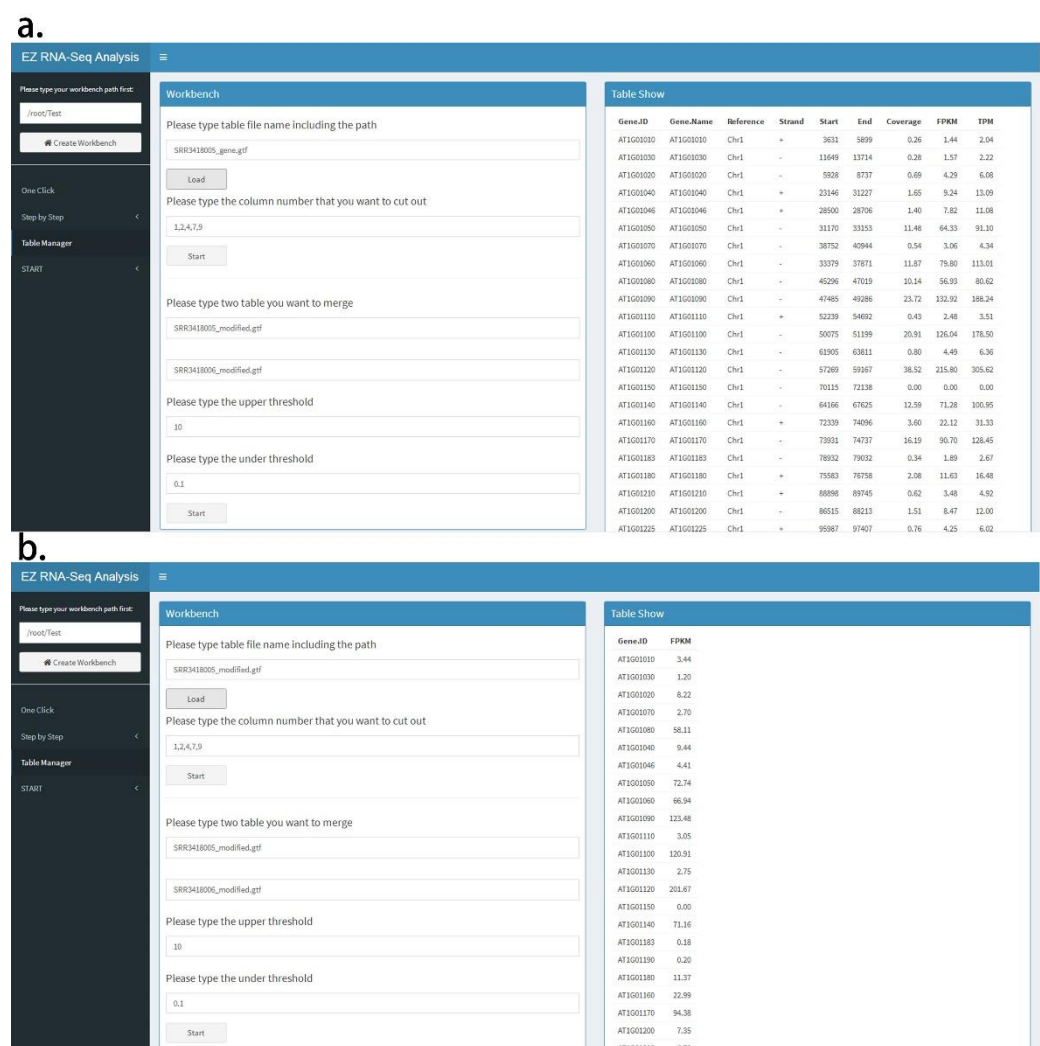


图 3.4 Table Manager 模块展示 (a).表格删除列处理前。(b).表格删除列处理后。

### 3.3 讨论

在生物信息学研究过程中,数据的处理通常需要使用到一系列由不同开发者开发的工具。由于开发者的个人对开发环境的偏好不同、不同的数据处理过程所适合的开发语言不同等因素,这些工具缺少一个统一的开发标准。同时,大部分生物信息学工具的使用缺少良好的图形交互界面,对于计算机知识相对较为薄弱的生物研究人员来说,阅读每一个工具的使用说明并针对自己的实验选择相应参



数使用命令行对数据进行加工无疑是一大挑战。

EZRS 选择了一套现今使用较多且适用面广的一套 RNA-seq 工具，研究人员将不会面对大量 RNA-seq 工具无从下手。这一套流程中，每一步产生的数据都可直接作为下面步骤的输入，研究人员将不必再对某些数据进行二次加工来满足下游分析工具的输入要求。EZRS 简介的交互页面也将给用户带来良好的使用体验。

研究人员需要良好的可视化提取有用信息，单纯生成的 reads 计数和 FPKM 文件可能包含上万个基因，需要一定 R 语言处理数据能力才能使用 R 包（譬如火山图、热图等）进行可视化。针对于这一点，EZRS 的“Table Manager”模块基于 R 语言，提供了简单的数据加工功能。处理完毕的数据可直接作为输入，提交给整合在 EZRS 中的 START 可视化 app，其中包含有各种功能强大的可视化模块。用户可以通过 START 选择参数，定制高质量的可视化图片（如图 3.5）。

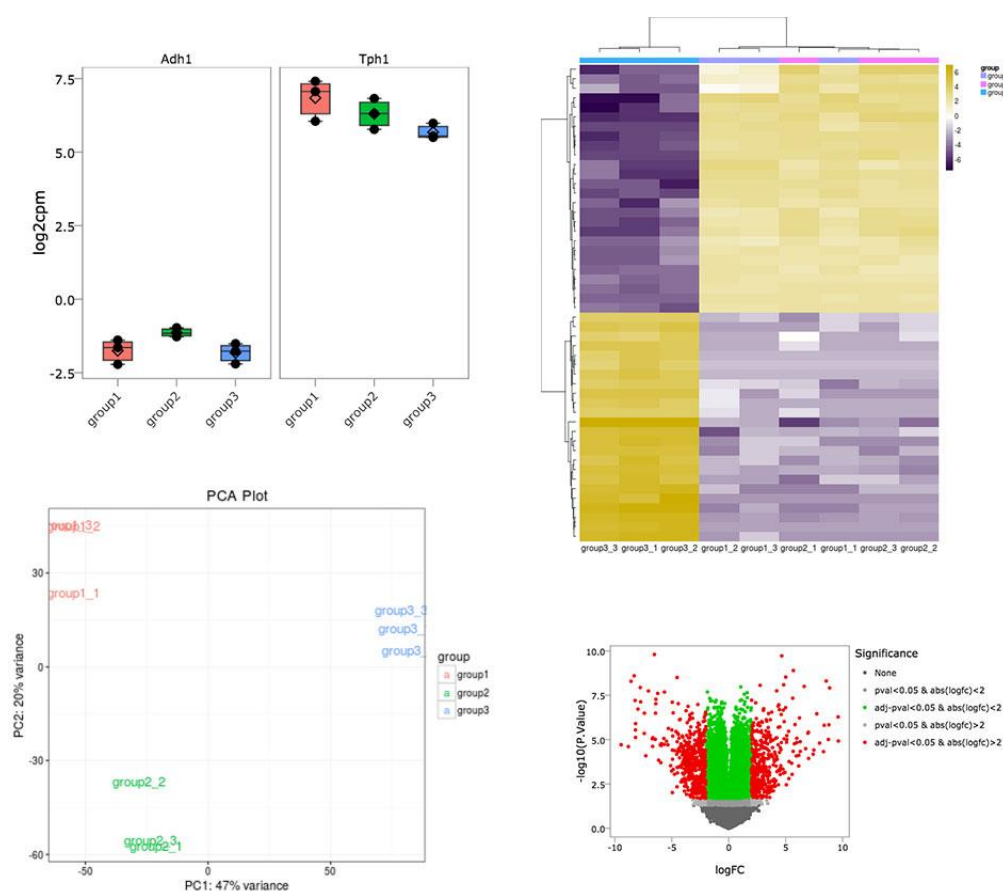


图 3.5 START 可视化展示



现如今，即便一些工具配有详细的配置说明书，但是仍然无法避免一些依赖问题。不同的环境中已安装的依赖不同，因此，相同的配置过程可能对不同的环境不适用。随着软件依赖的不断更新，即便完成了配置过程，最终 RNA-seq 过程中某一环节的输出文件将有可能不能直接作为下游工具的输入。现今，论文中实验结果的复现性问题也困扰着科研工作者。

EZRS 使用 Docker 技术，原理类似于虚拟机镜像，通过向其他研究人员提供一个二进制镜像来解决依赖性问题，其中所有软件都已经安装，配置和测试（镜像文件还可以包含研究所需的所有数据文件，可简化数据分布）。用户只需要从 Docker Hub 上直接下载部署已打包好的 EZRS 镜像文件，即可轻松复现开发环境。

### 3.4 展望

EZRS 中的一些功能还不完善，譬如“Table Manager”功能中还没有实现对同一个实验组中不同的生物学重复进行归并功能；在分步的 RNA-seq 步骤中，每一步还未实现函数化。在今后的开发维护过程中，将会根据用户的反馈信息，进一步开发、维护 EZRS。

同时，基于 Docker 可能解决某些现有方法对重现复杂计算环境造成的可重复研究挑战的缺点。因此生信领域相关研究实验室可利用此项技术开发更便捷，易部署，持久，灵活且具有私有仓库的生信工具来满足实验需求。EZRS 将会进一步改进整体框架结构，使其它生信工具更易于整合进来，扩展其功能，为今后相似的软件开发提供一个模板。

## 4 安装与配置

Docker 镜像包含开发的工具，依赖性和运行环境。该镜像基于 Ubuntu: 16.04 的基本镜像，并通过 apt-get 安装所有必需的库。该近镜像还会从其构建的路径中复制一些基因组文件和参考数据。所有构建镜像的 Docker 命令都可以在以下 Dockerfile 中找到：

```

1. FROM ubuntu:16.04
2. MAINTAINER Xiahao '3140105252@zju.edu.cn'
3. COPY ./sources.list /etc/apt/
4. ENV DEBIAN_FRONTEND noninteractive
5. RUN apt-get update \
6.     && apt-get -y upgrade \
7.     && apt-get install -y apt-utils dialog python-software-properties apt-transport-https software-
    properties-common \
8.     && apt-key adv --keyserver keyserver.ubuntu.com --recv-
    keys E298A3A825C0D65DFD57CBB651716619E084DAB9 \
9.     && add-apt-repository -
    y 'deb [arch=amd64,i386] https://mirrors.ustc.edu.cn/CRAN/bin/linux/ubuntu xenial/' \
10.    && apt-get update \
11.    && apt-get install -y wget r-base supervisor \
12.        libcurl4-openssl-dev libccol-dev libssl-dev libglu1-mesa-dev libglu1-mesa-dev \
13.        libcairo2-dev libxt-dev gdebi-core pandoc pandoc-citeproc libxml2-dev
14. COPY ./packages.R .
15. COPY ./tools/* /usr/local/bin/
16. RUN Rscript packages.R \
17.     && rm packages.R
18. COPY /*.deb shiny-server.deb
19. RUN gdebi -n shiny-server.deb \
20.     && rm -f shiny-server.deb
21.
22. # config for locale
23. RUN apt-get install -y locales locales-all python-pip\
24.     && apt-get autoremove
25. ENV LC_ALL en_US.UTF-8
26. ENV LANG en_US.UTF-8
27. ENV LANGUAGE en_US.UTF-8
28.
29. RUN pip install -i https://pypi.tuna.tsinghua.edu.cn/simple --upgrade pip
30. RUN pip install -i https://pypi.tuna.tsinghua.edu.cn/simple HTSeq
31. COPY ./test_demo /srv/shiny-server/
32. EXPOSE 3838
33. COPY shiny-server.sh /usr/bin/shiny-server.sh
34. RUN chmod +x /usr/bin/shiny-server.sh
35. CMD ["/usr/bin/shiny-server.sh"]

```

所需要的 R 包安装写在了 packages.R 文件：

```
1. source("http://bioconductor.org/biocLite.R")
2. chooseBioCmirror(graphics=FALSE,ind=11)
3. chooseCRANmirror(graphics=FALSE, ind=37)
4. tools_cran = c("shiny","datasets","shinyjs","shinydashboard","shinycssloaders","shinyBS","plotly","heatmaply","reshape2","ggplot2","ggthemes","gplots","ggvis","dplyr","tidyr","DT",
5.               "RColorBrewer","pheatmap","markdown","NMF","scales")
6. install.packages(tools_cran)
7. tools_bioc = c("limma","edgeR")
8. biocLite(tools_bioc,ask=FALSE)
```

可以通过将上述指令粘贴到 **DockerFile** 并运行 **Docker** 构建来构建映像，但更重要的是，可以通过 **docker pull**(假设安装了 **Docker** 引擎的 **GNU/Linux** 系统) 从 **Docker** 中央注册表中获取镜像：

```
1. $ docker pull mchenlab/ezrs
```

通过以下命令运行 **Docker** 容器，**path1** 为用户电脑上的目录，**path2** 是对应服务器原始数据存放位置。

```
1. $ docker run -d -p 0.0.0.0:3838:3838 -v /path1:/path2 mchenlab/ezrs:latest
```

## 5 参考文献

- [1] Xiong H, Brown J B, Boley N, et al. DE-FPCA: Testing Gene Differential Expression and Exon Usage Through Functional Principal Component Analysis[J]. 2014:129-143.
- [2] Klein M, Van d S H, Sanderson R, et al. Scholarly context not found: one in five articles suffers from reference rot.[J]. Plos One, 2014, 9(12):e115253.
- [3] Moraila G, Shankaran A, Shi Z, et al. Measuring Reproducibility in Computer Systems Research[J]. 2014:1-37.
- [4] Giardine B, Riemer C, Hardison R C, et al. Galaxy: a platform for interactive large-scale genome analysis.[J]. Genome Research, 2005, 15(10):1451-1455.
- [5] Docker,Inc . Docker - build, ship, and run any app, any- where. 2016.  
<https://www.docker.com/>. Accessed 20 January 2017.

- 
- [6] Nelson J W, Sklenar J, Barnes A P, et al. The START App: A Web-Based RNAseq Analysis and Visualization Resource[J]. *Bioinformatics*, 2016.
- [7] Andrews S. FASTQC. A quality control tool for high throughput sequence data.  
<http://www.bioinformatics.babraham.ac.uk/projects/fastqc/>. Accessed 29 September 2014.
- [8] Kim D, Langmead B, Salzberg S L. HISAT: a fast spliced aligner with low memory requirements.[J]. *Nature Methods*, 2015, 12(4):357-360.
- [9] Jouni Sirén. Indexing graphs for path queries with applications in genome research[J]. *IEEE/ACM Transactions on Computational Biology & Bioinformatics*, 2014, 11(2):375-388.
- [10] Li H, Handsaker B, Wysoker A, et al. The Sequence Alignment/Map format and SAMtools[J]. *Bioinformatics*, 2009, 25(16):2078-2079.
- [11] Pertea M, Pertea G M, Antonescu C M, et al. StringTie enables improved reconstruction of a transcriptome from RNA-seq reads.[J]. *Nature Biotechnology*, 2015, 33(3):290-295.
- [12] Anders S, Pyl P T, Huber W. HTSeq--a Python framework to work with high-throughput sequencing data.[J]. *Bioinformatics*, 2015, 31(2):166-9.
- [13] Chang, W. et al. (2016) shiny: Web Application Framework for R. R package version 0.13.1.  
<http://CRAN.R-project.org/package=shiny>
- [14] Team R D C. R : A language and environment for statistical computing[J]. *Computing*, 2013, 1:12-21.
- [15] Docker Inc. Docker - An open platform for distributed applications for developers and sysadmins. <http://www.docker.com/>. Online; Accessed 5-February-2015
- [16] Cheng G, Lu Q, Ma L, et al. BGDMdocker: a Docker workflow for data mining and visualization of bacterial pan-genomes and biosynthetic gene clusters[J]. *Peerj*, 2017, 5(1):e3948.
- [17] Ali A A, El-Kalioby M, Abouelhoda M. The Case for Docker in Multicloud Enabled Bioinformatics Applications[C]// *International Conference on Bioinformatics and Biomedical Engineering*. Springer International Publishing, 2016:587-601.
- [18] Ooms J. Possible Directions for Improving Dependency Versioning in R[J]. *R Journal*, 2013, 5(1):197-206.

- 
- [19] Aranguren M E, Wilkinson M D. Enhanced reproducibility of SADI web service workflows with Galaxy and Docker[J]. *Gigascience*, 2015, 4(1):1-9.
- [20] Folarin A A, Dobson R J, Newhouse S J. NGSeasy: a next generation sequencing pipeline in Docker containers[J]. *F1000 Research*, 2015.
- [21] Schmidburgk J L, Hornung V. BrowserGenome.org: web-based RNA-seq data analysis and visualization.[J]. *Nature Methods*, 2015, 12(11):1001.

## 作者简历

姓名：夏涵 性别：男 民族：汉 出生年月：1996-07-14 籍贯：安徽省马鞍山市

2011.09-2014.07 马鞍山市第二中学

2014.09-2018.07 浙江大学攻读学士学位

获奖情况：基础学科拔尖学生一等奖学金，优秀学生，优秀学生三等奖学金

参加项目：果蝇 mir-375 表达的时空特征及其功能研究，浙江大学竺可桢学院赴甘肃、青海省敦煌、西宁市非遗保护调研，声带麻痹与线粒体功能相关基因缺陷的研究，体外诱导人羊膜上皮干细胞分化为精子的研究

发表的学术论文：J.K.Wang, X.Q. Ding, H.Xia, Y.Wang, L.Tang, R.Xiong. "A LiDar based end to end controller for robot navigation using deep neural network". (2017) International Conference on Unmanned System.

