

Linux系统和R语言简明教程

主讲人：李国超

邮箱：liguochao@cncb.ac.cn

主讲时间：2025年11月21日

主讲地点：国家生物信息中心

目录 | CONTENTS



01

Linux系统概览与终端
基础

02

基础Linux命令：系统管理、
文件操作、文本处理

03

重定向、管道操作、
自动化脚本

04

R语言与R studio入门

05

使用data.table进行
数据处理

06

使用ggplot2
进行科研绘图

01

Linux系统概览与终端基础

Linux操作系统概览及其特点

Linux 是一套免费使用和自由传播的类 Unix 操作系统，是一个基于 POSIX 和 UNIX 的多用户、多任务、支持多线程和多 CPU 的操作系统

系统名称	说明	代表意义	
MS-DOS	微软早期命令行操作系统	原始命令行环境的代表	
UNIX	Unix系统，最早的多用户多任务系统之一	现代系统的祖先	
Windows	微软的图形化系统	主流商业桌面系统	
macOS	Unix-like 图形系统	BSD系图形系统代表	
Linux	开源类Unix操作系统	发展自Unix，免费、稳定、强大	

绝大多数生物信息分析软件都是为Linux系统开发的

生物信息学软件普遍依赖Linux系统，因为其在高性能计算、稳定性和工具兼容性方面远优于图形化操作系统。



为什么选Linux?

1. 大数据量需讲求性能

- 生物序列、甲基化、转录组数据体量巨大
- CPU/内存并行吞吐优先于 UI 界面

2. 超级计算机几乎都跑Linux

- 根据 Top500.org 数据, ~90% + 节点部署 Linux
- 类 Unix 环境在大规模并行计算与调度方面更成熟



常见的生物信息分析软件

1. 本地高性能C/C++工具

- 用途: 比对 (如Bowtie)、组装 (SOAP 系列)、BLAST/BLAT等
- 要求: 仅在类nix环境下编译/运行, Windows下运行不便

2. Java通用工具 (跨平台)

- 用途: 变异调用 (GATK)、QC (Picard)、Viewer (IGV)
- 特点: 只需jar便可运行, Linux/Windows/Unix共用



脚本工具及扩展

1. 基础脚本

- 语言: Perl、Python
- 功能: 文本查找、数据匹配等轻量操作 (无需额外系统依赖)

2. 扩展包依赖

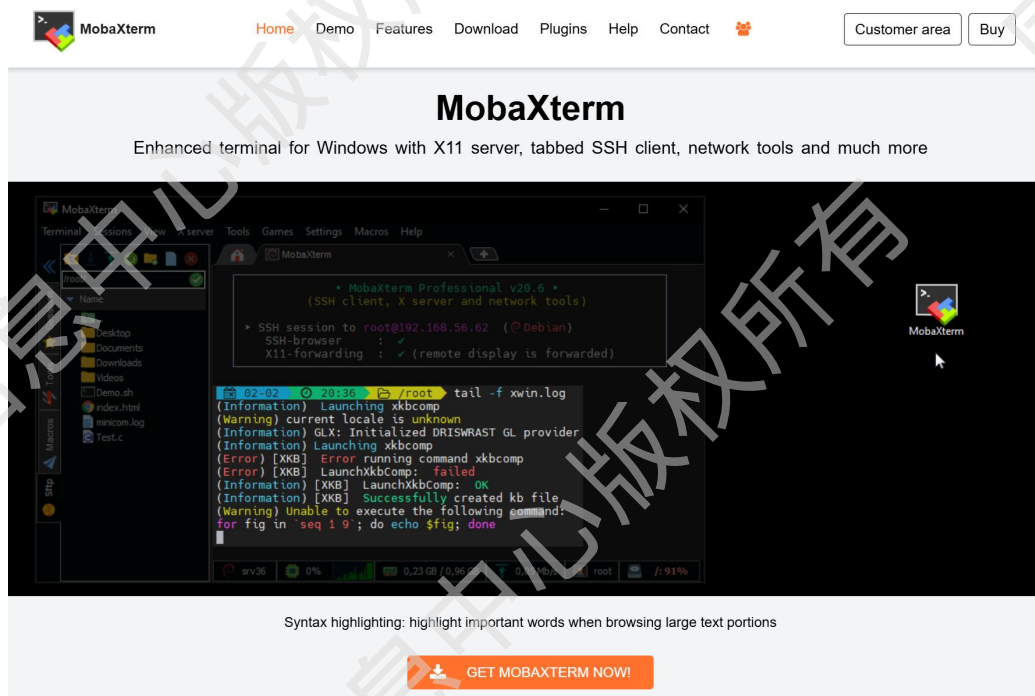
- 常见扩展: Bioperl、Biopython、Bioconductor
- 需依赖底层动态库 (如 zlib、libcurl、R、Fortran 库), Windows/macOS 下配置易出兼容问题

Linux系统凭借其开源免费、多用户多任务、安全稳定、可移植与可定制等优势，成为生物信息领域的首选操作系统

SSH远程登录软件推荐

MobaXterm是非常实用的SSH远程登陆软件，可通过该软件登陆集群节点（192.168.121.201），登陆节点仅允许编辑程序、脚本等操作

- 随着Linux在服务器端应用的普及，Linux系统管理越来越依赖于远程。Linux远程登录三种方式telnet，SSH， vnc. 由于SSH在安全性，功能和性能上都比较较好，推荐使用SSH。



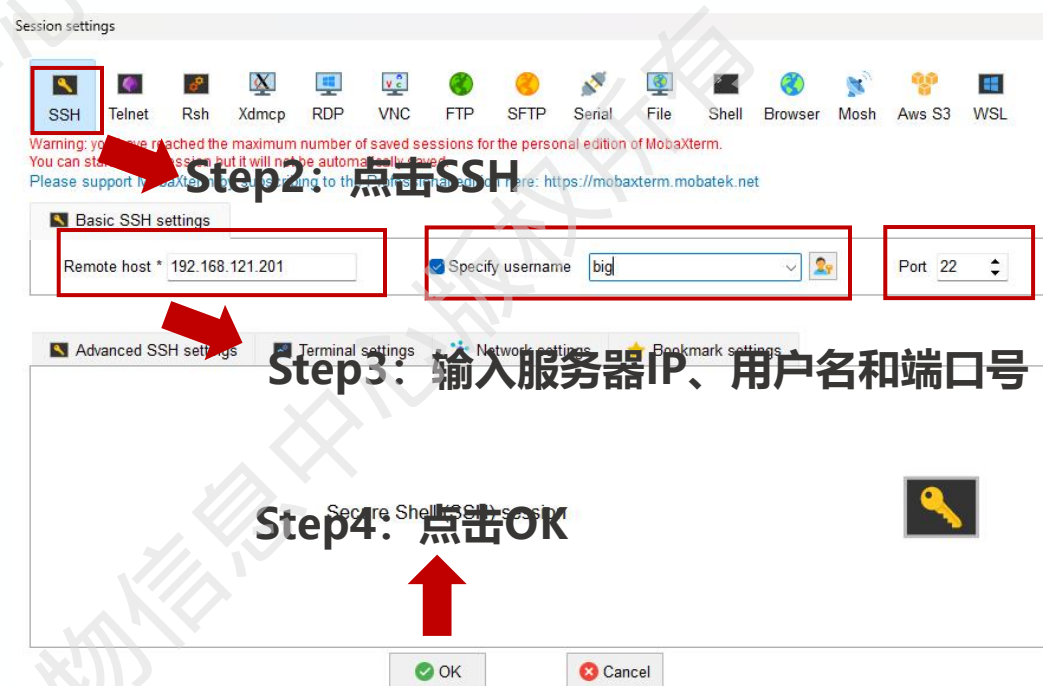
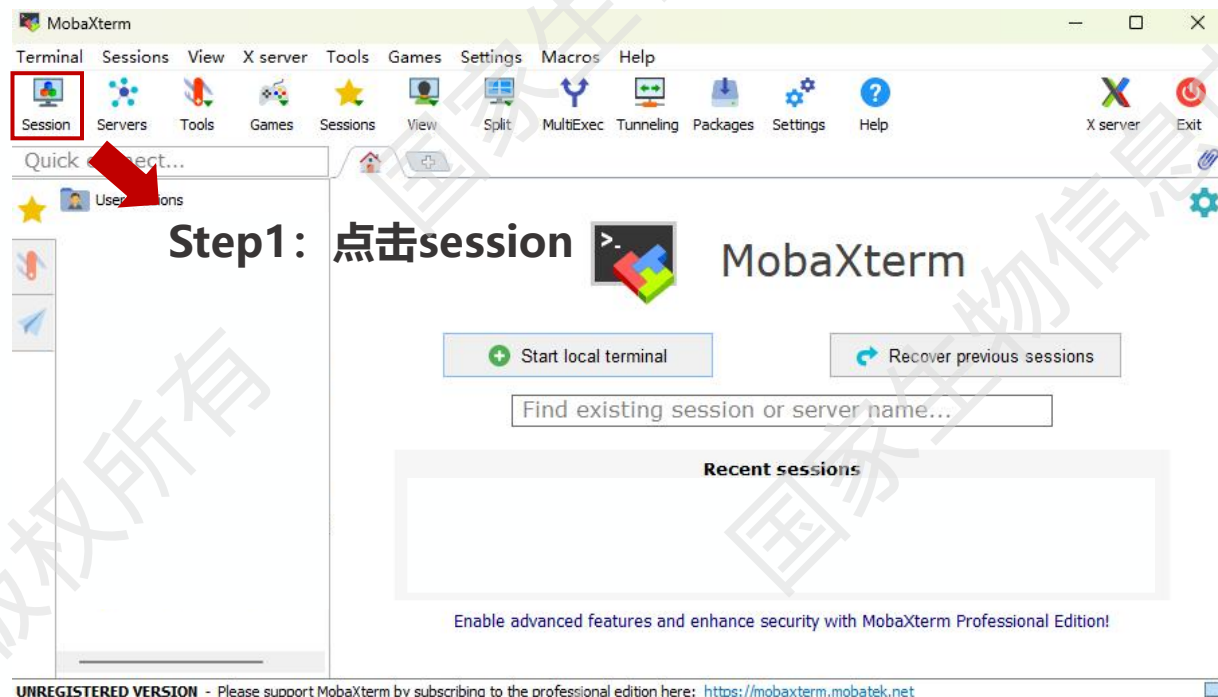
MobaXterm的官方网站：
<https://mobaxterm.mobatek.net/>

利用MobaXterm软件远程登录节点

通过MobaXterm的Session→SSH界面填写服务器IP、用户名和端口号（默认为22）并点击确认

步骤要点:

- 点击 “Session” → 选择 “SSH”
- 填写 Remote host (服务器 IP/域名)、Username (远程用户), 端口默认 22



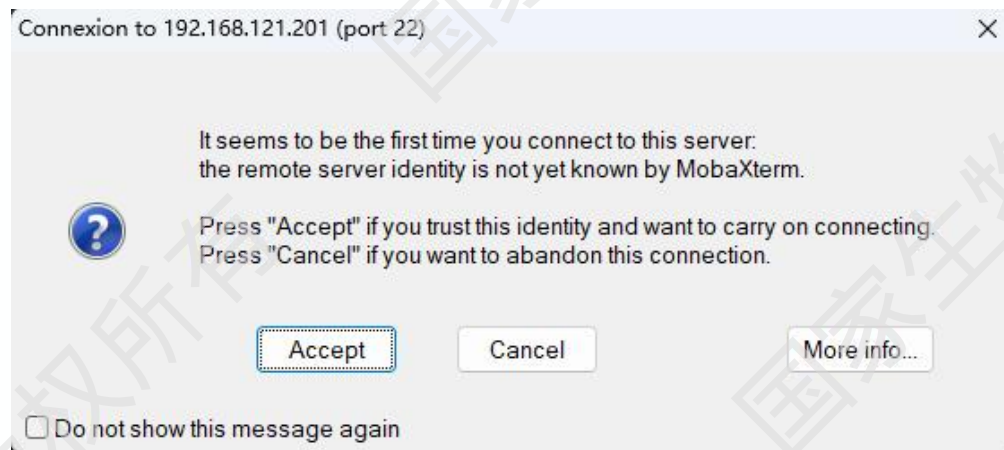
利用MobaXterm软件远程登录节点

弹出终端窗口后输入密码即可连接成功，进入远程终端

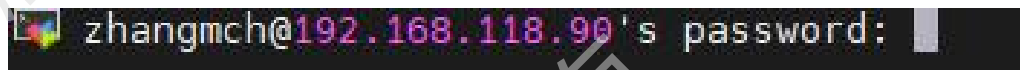
步骤要点：

- 点击 “Session” → 选择 “SSH”
- 填写 Remote host (服务器 IP/域名)、Username (远程用户)，端口默认 22
- “OK” → 弹出终端窗口 → 输入密码 (或自动密钥认证)
- 登录成功后，出现 username@hostname:~\$ 提示符

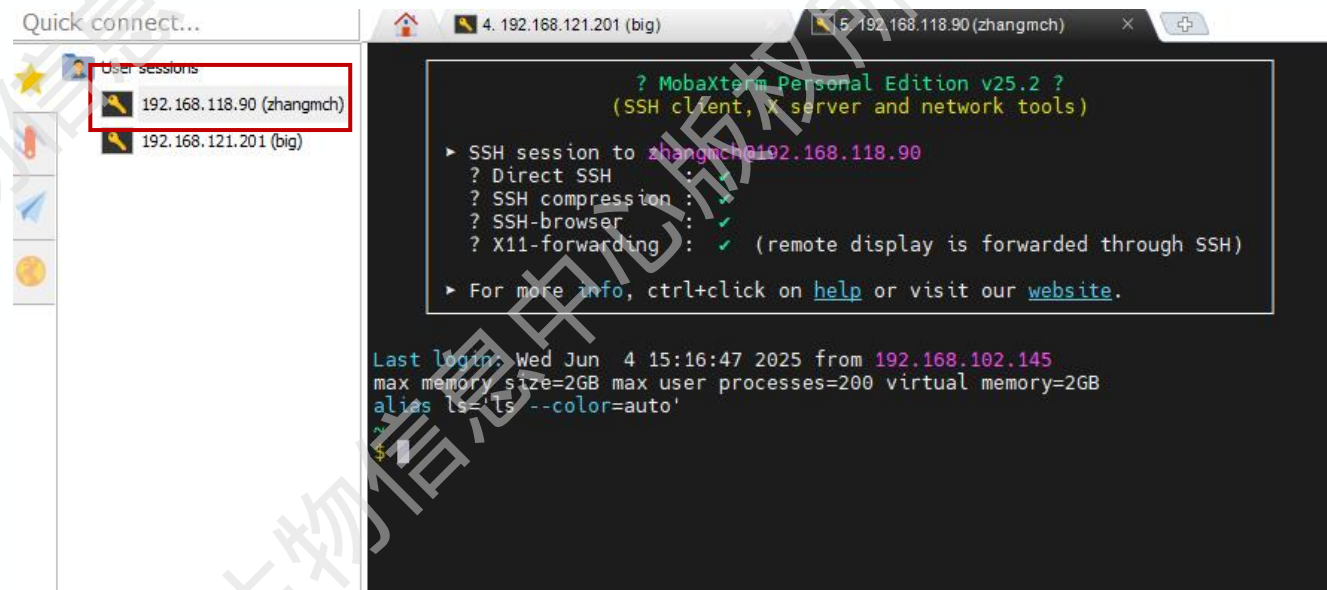
① 如果是第一次远程登陆节点，会弹出是否同意链接的提示框，点击同意



② 在终端窗口中输入对应的密码，就可以正常登陆



③ 远程登陆节点成功，且该服务器已经默认保存，后续登陆仅需双击红框中链接



02

基础Linux命令：系统管理、 文件操作、文本处理

常用基础Linux命令与Windows鼠标操作对比

分类	Linux命令	Windows操作
系统管理	ls	文件管理器，展示当前路径下的所有文件夹（目录）和文件
	cd	鼠标双击某文件夹，打开它
	pwd	win11文件管理器的顶端，显示当前路径
	mkdir	鼠标右键，新建文件夹
文件操作	cp	鼠标右键，复制文件夹/文件
	mv	鼠标右键，剪切+粘贴文件夹/文件，或者重命名文件夹/文件
	rm	鼠标右键，删除文件夹/文件
文本处理	cat	鼠标双击打开一个文本文件.txt，用记事本查看其内容
	head	Linux系统特有：查看文件开头的几行
	tail	Linux系统特有：查看文件最后的几行
	less	Linux系统特有：一页一页查看文件，而非一次性全部显示
	grep	查找某关键词是否在文件中，类似Word、Excel的“筛选”
	sort	对文本文件中的内容进行排序，类似Excel的“排序”
	uniq	去除文件中重复的行，只保留唯一的行
	cut	对表格文件进行纵向切割，输出特定的列
	wc -l	统计文件的行数

基础Linux命令：系统管理

常用基础Linux命令与Windows鼠标操作对比

分类	Linux命令	Windows操作
系统管理	ls	文件管理器，展示当前路径下的所有文件夹（目录）和文件
	cd	鼠标双击某文件夹，打开它
	pwd	win11文件管理器的顶端，显示当前路径
	mkdir	鼠标右键，新建文件夹
文件操作	cp	鼠标右键，复制文件夹/文件
	mv	鼠标右键，剪切+粘贴文件夹/文件，或者重命名文件夹/文件
	rm	鼠标右键，删除文件夹/文件
文本处理	cat	鼠标双击打开一个文本文件.txt，用记事本查看其内容
	head	Linux系统特有：查看文件开头的几行
	tail	Linux系统特有：查看文件最后的几行
	less	Linux系统特有：一页一页查看文件，而非一次性全部显示
	grep	查找某关键词是否在文件中，类似Word、Excel的“筛选”
	sort	对文本文件中的内容进行排序，类似Excel的“排序”
	uniq	去除文件中重复的行，只保留唯一的行
	cut	对表格文件进行纵向切割，输出特定的列
	wc -l	统计文件的行数

基础命令之ls

ls命令全称为list directory contents，其功能为列出当前目录或指定路径下的文件和子目录

主要作用：

- 1. 查看目录内容：快速浏览目录下的文件和文件夹名称
- 2. 获取详细信息：结合不同选项，可查看权限、大小、修改时间等属性
- 3. 脚本及管道：常与管道 (|) 和重定向结合，用于文件过滤、统计、批处理

基本用法：

ls [选项]

- 不带参数：列出当前目录下所有文件（不含隐藏文件）
- 常见选项：
 - -a (all)：显示所有文件，包括以“.”开头的隐藏文件
 - -l (long)：以长格式列出，包含权限、链接数、所有者、所属组、大小等信息
 - -h (human-readable)：与-l一起使用，将文件大小以可读格式显示
 - -t (time)：按修改时间排序，默认最新的排在最前
 - -r (reverse)：将输出结果倒序排列，可与其他选项组合
 - -R (recursive)：递归显示所有子目录内容
 - -S：按文件大小排序，大文件优先
 - -d：仅显示目录本身，而不列出其内部内容

基本示例：

- (1) 列出当前目录下所有文件（不含隐藏文件）

```
$ ls
data  GSE130001_RAW.tar  GSM3729178  GSM3729179  scmethbank  script
(base) zhangmch@bserver 16:18:25 /data40T/zhangmch/single-cell
```

解释：默认输出仅显示非隐藏条目。

- (2) 显示隐藏文件与长格式详细信息

```
$ ls -al
total 32628
drwxr-xr-x  7 zhangmch baoym_group  4096 Nov 23  2023 .
drwxr-xr-x 17 zhangmch baoym_group  4096 Jun  4 16:06 ..
drwxr-xr-x  4 zhangmch baoym_group  4096 Nov 29  2023 data
-rw-r--r--  1 zhangmch baoym_group 33382400 Nov 23  2023 GSE130001_RAW.tar
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 GSM3729178
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 GSM3729179
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 scmethbank
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 script
(base) zhangmch@bserver 16:20:26 /data40T/zhangmch/single-cell
```

解释：通过 -a 显示隐藏文件 (.env)，-l 提供文件属性、文件数、拥有者、所属的group、文件大小、建档日期、文件名等详细信息

- (3) 按修改时间倒序（最新在后）显示

```
$ ls -ltr
total 32620
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 scmethbank
-rw-r--r--  1 zhangmch baoym_group 33382400 Nov 23  2023 GSE130001_RAW.tar
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 GSM3729178
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 GSM3729179
drwxr-xr-x  2 zhangmch baoym_group  4096 Nov 23  2023 script
drwxr-xr-x  4 zhangmch baoym_group  4096 Nov 29  2023 data
(base) zhangmch@bserver 16:22:00 /data40T/zhangmch/single-cell
```

解释：-t 按时间排序，-r 倒序，-l 长格式。

基础命令之cd

cd命令全称为change directory，其功能为切换当前工作目录到指定路径

主要作用：

1. 快速导航文件系统层级
2. 切换到项目目录、脚本所在目录或父目录，准备后续文件操作
3. 在脚本中结合 cd 保证后续命令在正确路径下执行

基本用法：

cd [目标路径]

- **绝对路径**：以 / 开头，直接跳转到指定目录
 - 示例：cd /home/zhangmc/project
- **相对路径**：基于当前工作目录（PWD）跳转
 - 示例：cd ./src/（进入当前目录下的 src）
- **特殊符号**：
 - **cd ~ 或 cd**：回到用户主目录（等价于 cd /home/用户名）
 - **cd .**：留在当前目录（无实际效果）
 - **cd ..**：切换到父目录（上一级）
 - **cd -**：切换到上一次所在目录

基本示例：

(1) 切换到绝对路径

```
$ cd /data40T/zhangmch/
(base) zhangmch@bserver 16:30:16 /data40T/zhangmch
$ pwd
/data40T/zhangmch
(base) zhangmch@bserver 16:30:19 /data40T/zhangmch
```

(2) 切换到相对路径 假设当前路径为/data40T/zhangmch

```
$ cd download/
(base) zhangmch@bserver 16:31:21 /data40T/zhangmch/download
$ pwd
/data40T/zhangmch/download
(base) zhangmch@bserver 16:31:23 /data40T/zhangmch/download
```

(3) 返回主目录与上一次目录

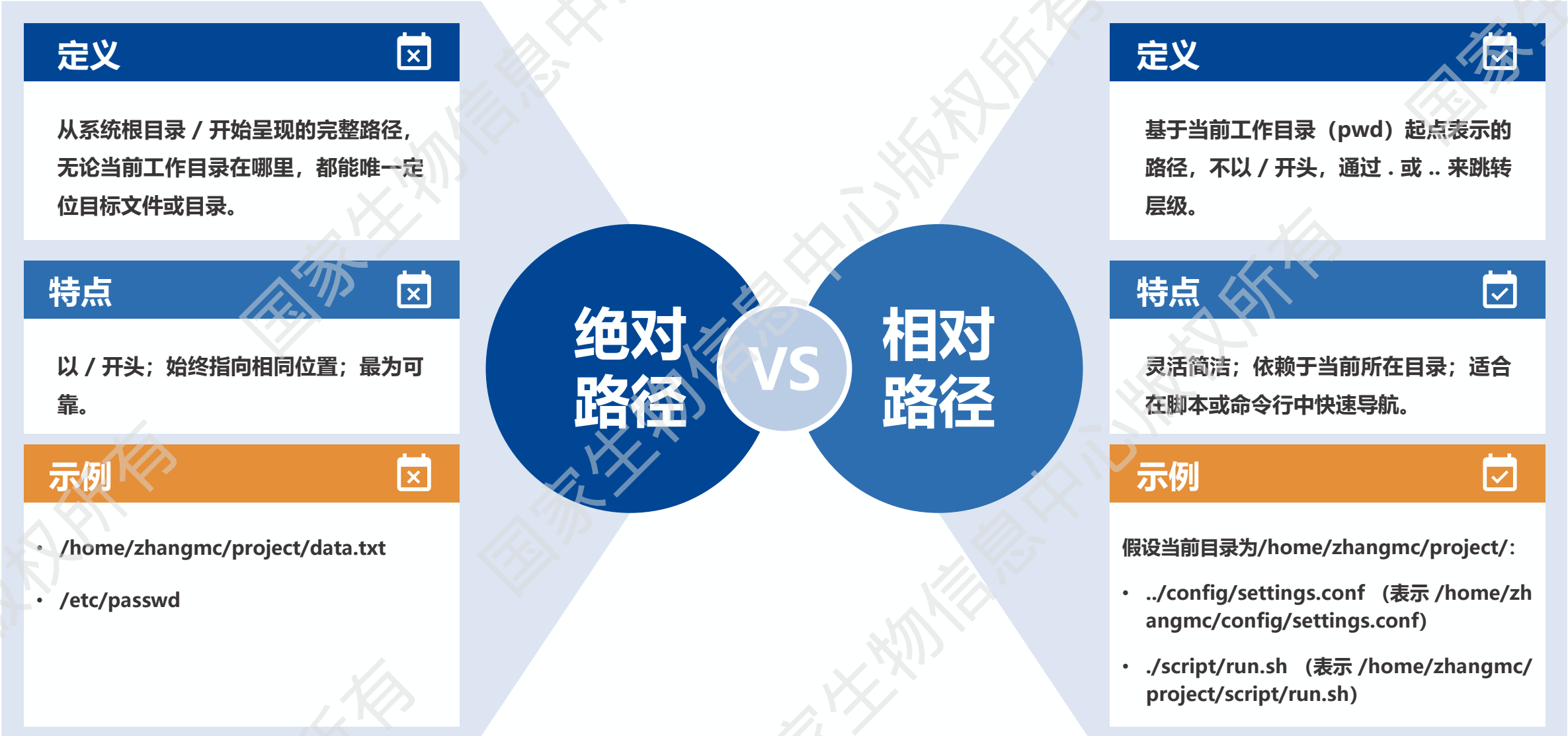
```
$ cd ~
(base) zhangmch@bserver 16:33:09 ~
$ pwd
/home/zhangmch
(base) zhangmch@bserver 16:33:11 ~
$ cd /var/log
(base) zhangmch@bserver 16:33:15 /var/log
$ cd -
/home/zhangmch
(base) zhangmch@bserver 16:33:19 ~
```

(4) 连续跳转

```
$ cd ../download/../single-cell/
(base) zhangmch@bserver 16:35:08 /data40T/zhangmch/single-cell
$ pwd
/data40T/zhangmch/single-cell
(base) zhangmch@bserver 16:35:08 /data40T/zhangmch/single-cell
```

文件路径表示法

绝对路径以根目录开头，能够唯一定位文件；相对路径基于当前工作目录进行层级跳转，虽简洁但依赖上下文



基础命令之pwd

pwd命令全称为print working directory，其功能为在终端中显示当前工作目录的绝对路径

主要作用：

1. **确认当前路径**：在多级目录跳转后，帮助用户快速定位目前所在目录
2. **脚本中使用**：为脚本提供当前目录信息，可结合变量或路径拼接，保证路径操作的正确性

基本用法：

pwd [选项]

- **不带参数**：打印当前目录的绝对路径
- **常见选项**：（该部分中的符号链接会在后面进行介绍）
 - **-L (Logical)**：显示符号链接逻辑路径（默认行为）
 - **-P (Physical)**：显示物理路径，解析所有符号链接后输出真实目录

基本示例：

```
$ pwd -L
/data40T/zhangmch/single-cell
(base) zhangmch@bserver 16:10:52 /data40T/zhangmch/single-cell
```

```
$ tree .
.
├── data
│   ├── 20231129.RData
│   └── data
│       ├── sce.output.merge.GSE130001.Rdata
│       ├── GSE130001_RAW
│       │   ├── GSM3729178
│       │   │   ├── barcodes.tsv.gz
│       │   │   ├── features.tsv.gz
│       │   │   └── matrix.mtx.gz
│       │   └── GSM3729179
│       │       ├── barcodes.tsv.gz
│       │       ├── features.tsv.gz
│       │       └── matrix.mtx.gz
│       └── GSE130001_RAW.tar
├── GSM3729178
│   ├── barcodes.tsv.gz
│   ├── fatures.tsv.gz
│   └── matrix.mtx.gz
├── GSM3729179
│   ├── barcodes.tsv.gz
│   ├── fatures.tsv.gz
│   └── matrix.mtx.gz
├── scmethbank
│   ├── align.py
│   ├── bis_result.py
│   ├── deduplicate.py
│   ├── QC.py
│   ├── rate.py
│   ├── scBSmap_H.py
│   └── trim.py
└── script
    ├── single-cell.R
    └── single-cell.RData

9 directories, 24 files
(base) zhangmch@bserver 16:06:17 /data40T/zhangmch/single-cell
```

基础命令之mkdir

mkdir命令全称为make directory，其功能为在指定路径下创建一个或多个新目录

主要作用：

1. **新建目录**：根据项目需求创建文件夹，用于组织文件和代码
2. **批量创建**：可一次性创建多级目录（结合 -p 选项）
3. **设置权限**：使用 -m 选项为新目录指定初始权限

基本用法：

mkdir [选项] 目录名...

- **不带参数**
- **常用选项：**
- **-p：（递归创建父目录）**
 - 效果：若上级目录不存在则一并创建，且已存在目录不报错
 - 示例：mkdir -p project/src/utils
- **-m：（为新目录设置权限掩码） - （该部分中的权限掩码会在后面进行介绍）**
 - 效果：直接按指定权限创建目录，无需之后再用 chmod 修改
 - 示例：mkdir -m 755 data
- **-v：（显示创建过程）**
 - 效果：每创建一个目录都会输出提示信息
 - 示例：mkdir -v logs archives

基本示例：

- (1) 创建单级目录 - 在当前目录下生成名为 reports 的文件夹

```
$ mkdir reports
(base) zhangmch@bserver 16:48:18 /data40T/zhangmch/single-cell
```

- (2) 递归创建多级目录 - 同时创建 project/src、project/bin、project/docs，若 project 不存在也一并生成

```
$ mkdir -p project/{src,bin,docs}
(base) zhangmch@bserver 16:48:41 /data40T/zhangmch/single-cell
```

- (3) 指定权限创建目录 - 创建 secret 目录并赋予权限 rwx-----

```
$ mkdir -m 700 secret
(base) zhangmch@bserver 16:48:59 /data40T/zhangmch/single-cell
```

- (4) 批量创建并显示过程 - 递归创建三层目录，并在终端输出每个新建目录的路径

```
$ mkdir -pv data/{raw,processed,output}
mkdir: created directory 'data/raw'
mkdir: created directory 'data/processed'
mkdir: created directory 'data/output'
(base) zhangmch@bserver 16:49:16 /data40T/zhangmch/single-cell
```

```
$ ls
total 32632
drwxr-xr-x 7 zhangmch baoym_group 4096 Jun  4 16:49 data
-rw-r--r-- 1 zhangmch baoym_group 33382400 Nov 23 2023 GSE130001_RAW.tar
drwxr-xr-x 2 zhangmch baoym_group 4096 Nov 23 2023 GSM3729178
drwxr-xr-x 2 zhangmch baoym_group 4096 Nov 23 2023 GSM3729179
drwxr-xr-x 5 zhangmch baoym_group 4096 Jun  4 16:48 project
drwxr-xr-x 2 zhangmch baoym_group 4096 Jun  4 16:48 reports
drwxr-xr-x 2 zhangmch baoym_group 4096 Nov 23 2023 scmethbank
drwxr-xr-x 2 zhangmch baoym_group 4096 Nov 23 2023 script
drwx----- 2 zhangmch baoym_group 4096 Jun  4 16:48 secret
```

基础Linux命令：文件操作

常用基础Linux命令与Windows鼠标操作对比

分类	Linux命令	Windows操作
系统管理	ls	文件管理器，展示当前路径下的所有文件夹（目录）和文件
	cd	鼠标双击某文件夹，打开它
	pwd	win11文件管理器的顶端，显示当前路径
	mkdir	鼠标右键，新建文件夹
文件操作	cp	鼠标右键，复制文件夹/文件
	mv	鼠标右键，剪切+粘贴文件夹/文件，或者重命名文件夹/文件
	rm	鼠标右键，删除文件夹/文件
文本处理	cat	鼠标双击打开一个文本文件.txt，用记事本查看其内容
	head	Linux系统特有：查看文件开头的几行
	tail	Linux系统特有：查看文件最后的几行
	less	Linux系统特有：一页一页查看文件，而非一次性全部显示
	grep	查找某关键词是否在文件中，类似Word、Excel的“筛选”
	sort	对文本文件中的内容进行排序，类似Excel的“排序”
	uniq	去除文件中重复的行，只保留唯一的行
	cut	对表格文件进行纵向切割，输出特定的列
	wc -l	统计文件的行数

文件操作之cp

敲击该命令，然后按回车，拷贝演示文件到自己的工作目录下
`cp -rf /data/biosoft/big/Linux_and_R/ ~/`

cp命令全称为copy，其功能为在文件系统中复制文件或目录。需根据需求选择合适参数，避免误操作

主要作用：

1. **不加参数，复制文件**：将一个或多个源文件复制到目标文件或目录
2. **复制目录**：结合递归选项，可复制整个目录及其子内容
3. **备份与更新**：内置选项可以实现基于时间或版本的复制备份

基本用法：

cp [选项] 源文件 目标文件或目录

- 常用选项：
- **-r 或 -R**：（递归复制目录及其所有子文件、子目录）
 - 示例：`cp -r src_dir/ dest_dir/`
- **-a**：（以归档模式复制，相当于 `-dR --preserve=all`，保留符号链接、权限、时间戳、所有者等）
 - 示例：`cp -a project/ project_backup/`
- **-p**：（保留源文件的权限、时间戳和所有者信息）
 - 示例：`cp -p file.txt /backup/`
- **-v**：（显示详细复制过程，输出每个复制操作的源和目标路径）
 - 示例：`cp -v a.txt b.txt`

基本示例：

(1) 复制单个文件 – 将GSE130001_RAW.tar复制到/data40T/zhangmch

```
$ cp GSE130001_RAW.tar /data40T/zhangmch/  
(base) zhangmch@bserver 17:43:18 /data40T/zhangmch/single-cell
```

(2) 复制并重命名 – 将GSE130001_RAW.tar复制为temp.tar

```
$ cp GSE130001_RAW.tar temp.tar  
(base) zhangmch@bserver 17:44:33 /data40T/zhangmch/single-cell
```

(3) 递归复制目录 – 将script目录及其所有内容复制到script_backup

```
$ cp -r script script_backup  
(base) zhangmch@bserver 17:45:28 /data40T/zhangmch/single-cell
```

(4) 归档模式复制 – 以完整归档模式复制script目录，保留符号链接、权限、时间戳等

```
$ cp -a script script_backup  
(base) zhangmch@bserver 17:46:54 /data40T/zhangmch/single-cell
```

文件操作之mv

mv命令全称为move，其功能为移动文件或目录位置，或重命名文件/目录。使用时务必确认目标路径与权限

主要作用：

1. **移动文件或目录**：将源路径下的文件/目录转移到目标路径
2. **重命名**：若目标路径指向同一目录下的新名称，则实现重命名
3. **脚本应用**：批量重构文件结构、整理目录、修改文件名等

基本用法：

mv [选项] 源文件 目标文件或目录

- **不加参数**
- 常用选项：
 - **-i (interactive)**：交互式模式，覆盖前提示确认
 - 示例：mv -i old.txt /backup/
 - **-f (force)**：强制移动，不提示，若目标存在则直接覆盖
 - 示例：mv -f report.pdf /home/user/docs/
 - **-n (no-clobber)**：不覆盖目标，如果目标已存在则跳过该文件
 - 示例：mv -n data.csv /data/archive/
 - **-v (verbose)**：显示详细操作过程，列出每个移动或重命名的源与目标
 - 示例：mv -v *.log /var/log/old_logs/

基本示例：

- (1) 移动文件到目录 – 将GSE130001_RAW.tar移动到script相对路径中

```
$ mv GSE130001_RAW.tar script
(base) zhangmch@bserver 17:57:17 /data40T/zhangmch/single-cell
```

- (2) 批量移动并覆盖（带提示） - 逐个提示是否覆盖目标文件

```
$ mv -i *.tar script/
mv: overwrite 'script/GSE130001_RAW.tar'? yes
(base) zhangmch@bserver 17:59:07 /data40T/zhangmch/single-cell
```

- (3) 重命名文件 – 在同一目录下将matrix.mtx.gz重命名为matrix_backup.mtx.gz

```
$ mv matrix.mtx.gz matrix_backup.mtx.gz
(base) zhangmch@bserver 18:00:53 /data40T/zhangmch/single-cell/GSM3729178
```

- (4) 重命名并移动目录 – 将GSM3729178目录移动到新位置并重命名为GSM3729178_new

```
$ mv -v GSM3729178/ /data40T/zhangmch/single-cell/GSM3729178_new
'GSM3729178/' -> '/data40T/zhangmch/single-cell/GSM3729178_new'
```

- (5) 防止覆盖已有文件 – 如果matrix_new.mtx.gz文件已存在则跳过

```
$ mv -n matrix.mtx.gz matrix_new.mtx.gz
(base) zhangmch@bserver 18:03:46 /data40T/zhangmch/single-cell/GSM3729178
```

基础命令之rm

rm命令全称为remove，其功能为删除一个或多个文件和目录。使用时请小心谨慎，避免误删

主要作用：

1. **删除文件**：永久移除指定的文件
2. **删除目录**：结合递归选项，可删除非空目录及其内容
3. **脚本清理**：常用于清理临时文件、清空目标目录等自动化操作

基本用法：

rm [选项] 文件或目录...

- 常用选项：
- **-f**：强制删除（忽略不存在的文件，不显示错误提示）
 - 示例：rm -f old.txt
- **-i**：交互式删除（删除前逐一询问用户确认）
 - 示例：rm -i data.csv
- **-r**：递归删除（删除目录及其所有子文件和子目录）
 - 示例：rm -r project/backup
- **-v**：显示过程（删除每个文件或目录时输出提示信息）
 - 示例：rm -rv logs/

基本示例：

- (1) 删除单个文件 – 直接删除newfile.txt

```
$ rm newfile.txt
(base) zhangmch@bserver 17:04:49 /data40T/zhangmch/single-cell
```

- (2) 强制删除多个文件 – 强制删除temp1.txt和temp2.txt文件

```
$ rm -f temp1.txt temp2.txt
(base) zhangmch@bserver 17:05:17 /data40T/zhangmch/single-cell
```

如果temp1.txt和temp2.txt不存在，则不显示错误信息

- (3) 递归删除目录

```
$ rm -r project/
(base) zhangmch@bserver 17:06:34 /data40T/zhangmch/single-cell
```

- (4) 交互式确认删除递归目录

```
$ rm -r -i project/
rm: descend into directory 'project/'? yes
rm: remove directory 'project/src'? yes
rm: remove directory 'project/docs'? yes
rm: remove directory 'project/bin'? yes
rm: remove directory 'project/'? yes
(base) zhangmch@bserver 17:22:40 /data40T/zhangmch/single-cell
```

逐个询问是否删除project下面的所有文件夹及对应文件

基础Linux命令：文本处理

常用基础Linux命令与Windows鼠标操作对比

分类	Linux命令	Windows操作
系统管理	ls	文件管理器，展示当前路径下的所有文件夹（目录）和文件
	cd	鼠标双击某文件夹，打开它
	pwd	win11文件管理器的顶端，显示当前路径
	mkdir	鼠标右键，新建文件夹
文件操作	cp	鼠标右键，复制文件夹/文件
	mv	鼠标右键，剪切+粘贴文件夹/文件，或者重命名文件夹/文件
	rm	鼠标右键，删除文件夹/文件
文本处理	cat	鼠标双击打开一个文本文件.txt，用记事本查看其内容
	head	Linux系统特有：查看文件开头的几行
	tail	Linux系统特有：查看文件最后的几行
	less	Linux系统特有：一页一页查看文件，而非一次性全部显示
	grep	查找某关键词是否在文件中，类似Word、Excel的“筛选”
	sort	对文本文件中的内容进行排序，类似Excel的“排序”
	uniq	去除文件中重复的行，只保留唯一的行
	cut	对表格文件进行纵向切割，输出特定的列
	wc -l	统计文件的行数

文件内容查看之head

head的主要作用是显示文本文件开头多行内容，默认输出前10行，常用于快速查看日志或数据文件的头部概况

基本用法:

head [选项] [文件...]

常用选项:

选项	含义	示例
-n <行数>	指定输出前 <行数> 行	head -n 5 logfile.txt
-c <字节数>	指定输出前 <字节数> 字节	head -c 100 data.csv
-q	静默模式，不输出文件名前缀	head -q file1 file2
-v	输出文件名前缀（多文件时可区分来源）	head -v file1 file2

注意事项:

- 对于非常大的文件，仅输出头部避免全量加载造成延迟
- 如果需要查看尾部，应使用 tail 命令
- 若想分页查看，可结合管道：
 - head -n 50 largefile.txt | less

基本示例:

(1) 默认查看前10行 – 默认不加-n时，head等同于head -n 10

```
$ head align.py
import os
import sys

if __name__ == '__main__':
    bin_path = "bismark"
    output_path = sys.argv[2]
    input_path = sys.argv[1]
    contraction_path = "/data40T/zongwt/genome/hg19_lambda/"
    params = "--bowtie2 --non_directional -p 4 -unmapped --bam --quiet"
(base) zhangmch@bserver 17:41:03 /data40T/zhangmch/single-cell/scmethbank
```

(2) 查看前128字节 - -c适用于查看二进制文件或固定字节段

```
$ head -c 128 single-cell.R
#单细胞scRNA_Seq代码
library(Seurat)
library(minidr)
#上面的包在/home/zhangmch/R4.3.0-package/minidr_1.3.2.tar.gz中
lib(base) zhangmch@bserver 17:40:26 /data40T/zhangmch/single-cell/script
```

(3) 同时查看多个文件的前3行 – 多文件场景配合-v/-q可控制是否

显示文件名标签

```
$ head -n 3 align.py QC.py bis_result.py
==> align.py <==
import os
import sys

==> QC.py <==
import os
import sys

==> bis_result.py <==
import os
import sys
(base) zhangmch@bserver 17:41:40 /data40T/zhangmch/single-cell/scmethbank
```


less是一个强大的文本查看器，用于分页显示文件内容，允许前后翻页和快速搜索，常用于查看大文件

less [选项] [文件...]

28

文件内容查看之grep

grep支持在文本文件中根据指定模式（字符串或正则表达式）搜索并打印匹配行，是非常强大的文本搜索工具

基本用法：

grep [选项] PATTERN [文件...]

常用选项：

选项	含义	示例
-i	忽略大小写匹配	grep -i "error" logfile.txt
-v	反向匹配，只显示不包含模式的行	grep -v DEBUG logfile.txt
-n	显示匹配行的行号	grep -n "TODO" script.sh
-r/-R	递归搜索目录下所有文件	grep -R "main()" src/
-c	只输出匹配行的计数	grep -c "WARNING" syslog
-l	只列出包含匹配的文件名	grep -l "password" *.conf

注意事项：

- 正则语法差异：默认使用基本正则，需 -E 才能启用扩展正则。
- 大文件性能：对于超大文件或二进制文件，建议先用 head/tail 限制范围，或使用 fgrep (grep -F) 进行固定字符串匹配。
- 文件名匹配：-l 列出文件名时，不会输出具体内容。
- 高亮支持：部分终端需加 --color=always 才能正确高亮。

基本示例：

(1) 基本搜索：输出所有包含“chr15”的行
grep 'chr15' all_crack.mcpqi | head -n 20

```
grep 'chr15' all_crack.mcpqi | head -n 20
chr15 98991482 98991484 Cgi: 30 340 30 228 17.2 65.5 0.8 99.999999999999 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 2824622 20295805 Cgi: 100 983 100 627 20.3 63.8 1.87 71.546 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 2786992 2786944 Cgi: 20 252 20 164 15.9 65.9 0.73 77.1335 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 8575718 8575802 Cgi: 20 284 20 184 15.2 62.1 0.79 66.420574265743 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 74410712 74410314 Cgi: 15 262 15 130 14.9 64.4 0.73 91.111333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 24675466 24676105 Cgi: 30 630 30 444 10.5 69.5 0.77 100.0 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 35121881 35122230 Cgi: 30 358 36 243 20.1 67.9 0.67 92.262420574206 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 3589230 3589305 Cgi: 20 230 27 183 16.1 64.6 1.2 96.404025 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 89333159 89333725 Cgi: 51 566 51 381 18 67.3 0.8 80.0 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 39676800 39676800 Cgi: 50 400 50 312 20.8 65 0.99 78.1235 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 18139840 18139860 Cgi: 23 228 23 152 20.2 66.7 0.91 89.152777777777 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 40935244 40936374 Cgi: 30 430 30 281 14 65.3 0.65 92.444333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 26891754 26894206 Cgi: 57 752 57 581 15.2 66.6 0.68 63.446069999999 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 28168341 28168330 Cgi: 47 489 47 337 19.2 60.9 0.83 75.595371426574 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 66789968 66781605 Cgi: 74 637 74 452 23.2 71 0.93 88.333333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 77904203 77904203 Cgi: 23 228 23 184 17.8 71 0.72 86.420574265743 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 96997447 96997401 Cgi: 19 214 19 150 17.8 70.1 0.73 96.495021791344 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 63321468 63321386 Cgi: 32 418 32 288 15.3 68.9 0.65 90.8771875 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 181752508 181764920 Cgi: 708 12312 708 7209 13 59.2 0.75 86.5483846153847 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
```

(2) 忽略大小写搜索：不考虑大小写的情况下匹配chr15字符
grep -i 'chr15' all_crack.mcpqi | head -n 20

```
grep -i 'chr15' all_crack.mcpqi | head -n 20
chr15 98991482 98991484 Cgi: 30 340 30 228 17.2 65.5 0.8 99.999999999999 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 2824622 20295805 Cgi: 100 983 100 627 20.3 63.8 1.87 71.546 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 2786992 2786944 Cgi: 20 252 20 164 15.9 65.9 0.73 77.1335 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 8575718 8575802 Cgi: 20 284 20 184 15.2 62.1 0.79 66.420574265743 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 74410712 74410314 Cgi: 15 262 15 130 14.9 64.4 0.73 91.111333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 24675466 24676105 Cgi: 30 630 30 444 10.5 69.5 0.77 100.0 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 35121881 35122230 Cgi: 30 358 36 243 20.1 67.9 0.67 92.262420574206 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 3589230 3589305 Cgi: 20 230 27 183 16.1 64.6 1.2 96.404025 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 89333159 89333725 Cgi: 51 566 51 381 18 67.3 0.8 80.0 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 39676800 39676800 Cgi: 50 400 50 312 20.8 65 0.99 78.1235 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 18139840 18139860 Cgi: 23 228 23 152 20.2 66.7 0.91 89.152777777777 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 40935244 40936374 Cgi: 30 430 30 281 14 65.3 0.65 92.444333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 26891754 26894206 Cgi: 57 752 57 581 15.2 66.6 0.68 63.446069999999 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 28168341 28168330 Cgi: 47 489 47 337 19.2 60.9 0.83 75.595371426574 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 66789968 66781605 Cgi: 74 637 74 452 23.2 71 0.93 88.333333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 77904203 77904203 Cgi: 23 228 23 184 17.8 71 0.72 86.420574265743 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 96997447 96997401 Cgi: 19 214 19 150 17.8 70.1 0.73 96.495021791344 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 63321468 63321386 Cgi: 32 418 32 288 15.3 68.9 0.65 90.8771875 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 181752508 181764920 Cgi: 708 12312 708 7209 13 59.2 0.75 86.5483846153847 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
```

(2) 扩展正则，同时匹配chr1和chr15字符
grep -E "chr1|chr15" all_crack.mcpqi | head -n 20

```
grep -E "chr1|chr15" all_crack.mcpqi | head -n 20
chr1 98991482 98991484 Cgi: 30 340 30 228 17.2 65.5 0.8 99.999999999999 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 2824622 20295805 Cgi: 100 983 100 627 20.3 63.8 1.87 71.546 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 2786992 2786944 Cgi: 20 252 20 164 15.9 65.9 0.73 77.1335 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 8575718 8575802 Cgi: 20 284 20 184 15.2 62.1 0.79 66.420574265743 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 74410712 74410314 Cgi: 15 262 15 130 14.9 64.4 0.73 91.111333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 24675466 24676105 Cgi: 30 630 30 444 10.5 69.5 0.77 100.0 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 35121881 35122230 Cgi: 30 358 36 243 20.1 67.9 0.67 92.262420574206 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 3589230 3589305 Cgi: 20 230 27 183 16.1 64.6 1.2 96.404025 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 89333159 89333725 Cgi: 51 566 51 381 18 67.3 0.8 80.0 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 39676800 39676800 Cgi: 50 400 50 312 20.8 65 0.99 78.1235 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 18139840 18139860 Cgi: 23 228 23 152 20.2 66.7 0.91 89.152777777777 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 40935244 40936374 Cgi: 30 430 30 281 14 65.3 0.65 92.444333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 26891754 26894206 Cgi: 57 752 57 581 15.2 66.6 0.68 63.446069999999 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 28168341 28168330 Cgi: 47 489 47 337 19.2 60.9 0.83 75.595371426574 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 66789968 66781605 Cgi: 74 637 74 452 23.2 71 0.93 88.333333333333 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 77904203 77904203 Cgi: 23 228 23 184 17.8 71 0.72 86.420574265743 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 96997447 96997401 Cgi: 19 214 19 150 17.8 70.1 0.73 96.495021791344 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 63321468 63321386 Cgi: 32 418 32 288 15.3 68.9 0.65 90.8771875 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
chr15 181752508 181764920 Cgi: 708 12312 708 7209 13 59.2 0.75 86.5483846153847 Ltp_Normal_Non-syndromicCleftIpsidPalate_Smooths_Female_Rep2
```

文件属性统计之wc、sort和uniq



01. wc

统计文件的行数、单词数、字符数

选项	含义	示例
-l	统计行数	wc -l file.txt
-w	统计单词数	wc -w file.txt
-c	统计字节数	wc -c file.txt
-m	统计字符数	wc -m file.txt

最常用的是wc -l来统计行数



02. sort

按字母、数字或其他顺序
对文本文件中的内容进行排序

选项	含义	示例
-n	按数字排序	sort -n numbers.txt
-r	逆序排序	sort -r file.txt
-k	按指定列排序	sort -k 2 file.txt
-u	去除重复行	sort -u file.txt

最常用的是sort -k对比对文件排序



03. uniq

去除文件中重复的行，
只保留唯一的行

选项	含义	示例
-c	统计每一行出现的次数	uniq -c file.txt
-d	只显示重复出现的行	uniq -d file.txt
-u	只显示不重复的行	uniq -u file.txt

最常用的是uniq -c去重显示行次数

文件属性统计之wc、sort和uniq

01. wc

(1) 查看文件行数: `wc -l file.txt`

```
$ wc -l SRR20307773_1.fastq
116332784 SRR20307773_1.fastq
/p300s/methbank_baoym/zhangmch/scWGBS
```

(2) 查看文件单词数: `wc -w file.txt`

```
$ wc -w SRR20307773_1.fastq
232665568 SRR20307773_1.fastq
/p300s/methbank_baoym/zhangmch/scWGBS
```

(3) 查看文件字节数: `wc -c file.txt`

```
$ wc -c SRR20307773_1.fastq
11181669244 SRR20307773_1.fastq
/p300s/methbank_baoym/zhangmch/scWGBS
```

02. sort

(1) 按照字母顺序排序: `sort file.txt`

```
a
b
e
d
s
a
file2.txt (END)

$ sort -n file2.txt
a
a
b
d
e
s
/p300s/methbank_baoym/zhangmch/scWGBS
```

(2) 按照数字大小排序: `sort -n numbers.txt`

```
1 123
2 456
3 789
4 101
5 201

$ sort file1.txt
101
123
201
456
789
/p300s/methbank_baoym/zhangmch/scWGBS
```

(3) 逆序排序: `sort -r file.txt`

```
1 123
2 456
3 789
4 101
5 201

$ sort -r file1.txt
789
456
201
123
101
/p300s/methbank_baoym/zhangmch/scWGBS
```

03. uniq

(1) 去重并显示每行出现的次数 - `uniq -c file2.txt`

```
a
b
e
d
s
a
b
s
d
r
r
w
file2.txt (END)

$ uniq -c file2.txt
1 a
1 b
1 e
1 d
1 s
1 a
1 b
1 s
1 d
2 r
1 w
/p300s/methbank_baoym/zhangmch/scWGBS
```

(2) 显示重复的行 - `uniq -d file2.txt`

```
$ uniq -d file2.txt
r
/p300s/methbank_baoym/zhangmch/scWGBS
```

(3) 显示唯一的行 - `uniq -u file2.txt`

```
$ uniq -u file2.txt
a
b
e
d
s
a
b
s
d
w
/p300s/methbank_baoym/zhangmch/scWGBS
```

必须是连续出现的行才可以将次数进行累计

命令组合使用:

- 结合 `sort` 和 `uniq` 去除文件中的重复行并排序 -> `sort file.txt | uniq`
- 统计文件中每个单词的出现次数 -> `sort file.txt | uniq -c`
- 备注: 管道符会在下一个章节中讲解

文件字段处理之cut

cut用于从文本行中提取指定的字段或字符范围，常用于拆分和筛选以定界符分割的数据列

基本用法：

```
cut [选项] [文件...]
```

基本示例：

```
name,age,city,score
Alice,30,New York,85
Bob,25,Los Angeles,90
Charlie,28,Chicago,88
David,35,Houston,92
```

常用选项：

选项	含义	示例
-f <字段列表>	按字段号提取，以 - 或 , 分隔多个字段	cut -f 1,3 file.tsv
-d <分隔符>	指定字段分隔符（默认是制表符 \t）	cut -d',' -f2 file.csv
-c <字符范围>	按字符位置提取，如 1-5、-4、3-	cut -c1-10 file.txt
--complement	取反选项，输出不在指定字段/字符范围内的内容	cut -d: --complement -f1 /etc/passwd

(1) 提取第一列和第四列

```
cut -d',' -f1,4 data.csv
```

```
$ cut -d',' -f1,4 data.csv
name,score
Alice,85
Bob,90
Charlie,88
David,92
/p300s/methbank_baoym/zhangmch/scWGBS
```

(2) 提取每行前五个字符

```
cut -c1-5 data.csv
```

```
$ cut -c1-5 data.csv
name,
Alice
Bob,2
Charl
David
/p300s/methbank_baoym/zhangmch/scWGBS
```

注意事项：

- 默认分隔符：若不指定 -d，cut 会将制表符视作分隔符，其他分隔符需显式指定
- 字段越界：请求的字段或字符范围超出行长度，则只输出现有部分，不报错
- 兼容性：GNU cut 与 BSD cut 在处理 -c 范围表达式时行为略有差异，请根据环境测试
 - GNU cut在绝大多数实现中按字节截取，而BSD cut是字符感知的，会正确按字符分隔。

03

重定向、管道操作、 自动化脚本

Linux神器之管道符 |

管道符 | 将左侧命令的输出流连接到右侧命令的输入流，相当于在内存中开辟一个缓冲区（匿名管道），自动完成读写同步

```
head -20 data.txt | tail -10 | less -N
```

显示data.txt文件的前20行

针对data.txt文件的前20
行文本，显示其后10行

显示data.txt文件的第11行到第20行

展示文本及其对应的行号

使用less命令展示文件第10行到第20行，并显示行号

管道符左边命令的

输出会作为管道符

右边命令的输入

标准流与文件描述符

标准输入 (fd 0)、标准输出 (fd 1) 和标准错误 (fd 2) 分别对应命令的输入、正常输出和错误输出，并可通过重定向灵活地将它们导向文件或其他流



基本重定向符号

重定向通过符号（如 `>`, `>>`, 等）将命令的标准输入、标准输出或标准错误流指向或合并到文件或其他流，以实现数据流的灵活控制

符号	含义	示例
<code>></code>	将 <code>stdout</code> (<code>fd1</code>) 重定向到文件, 覆盖写入	<code>ls > files.txt</code>
<code>>></code>	将 <code>stdout</code> (<code>fd1</code>) 重定向到文件, 追加写入	<code>echo "log" >> app.log</code>
<code> </code>	将前一个命令的 <code>stdout</code> (<code>fd1</code>) 连接到后一个命令的 <code>stdin</code> (<code>fd0</code>)	<code>cat file.txt grep "error"</code>

综合应用1

```
(base) [big@training ~/Linux_and_R]$ head volcano.txt
gene_name      log2FoldChange  padj
gene_1    2.47060583608243    0.00533799333497882
gene_2   -3.34977522570527    0.0315927188843489
gene_3    0.0166387200737722    0.023365664370358
gene_4    0.236834604350435    0.0236930448748171
gene_5    2.05732659732233    0.0581022798828781
gene_6   -7.7252321678422    0.0618015656247735
gene_7    1.26776736071744    0.0223199534219934
gene_8   -1.83832437217891    0.0179345544800162
gene_9    3.84302196150537    0.032099739741534
```

图中是模拟生成的人2万个基因在case vs control比较时，得到的差异表达情况表

使用linux命令组合统计padj这一列小于0.05的基因数量：

```
cat volcano.txt | tail -n+2 | cut -f3 | awk '$1 < 0.05' | wc -l > output
```

代码解释：打印volcano.txt中的全部内容 | 从第二行开始打印 | 只打印第三列 | 只输出小于0.05的行 | 统计行数，即基因数 > 将结果输出到名为“output”的文件中

```
(base) [big@training:~/Linux_and_R]$ cat volcano.txt | tail -n+2 | cut -f3 | awk '$1 < 0.05' | wc -l > output
(base) [big@training:~/Linux_and_R]$ cat output
12491
```

综合应用2

```
(base) [big@training:~/Linux_and_R]$ head all_contig_annotations.csv
barcode,is_cell,contig_id,high_confidence,length,chain,v_gene,d_gene,j_gene,c_gene,full_length,productive,fwr1,fwr1_nt,cd1,cd1_nt,fwr2,fwr2_nt,cd2,cd2_nt,fwr3,fwr3_nt,cd3,cd3_nt,fwr4,fwr4_nt,reads,umis,raw_clonotype_id,raw_c
onsensus_id,exact_subclonotype_id
AAACCTGAGGCCCTTG-1,false,AAACCTGAGGCCCTTG-1_contig_1,false,500,TRB,TRBV5-1,,TRBJ1-1,TRBC1,true,true,,,,,,,,,CASSLVGGNTEAFF,TGCGCCAGCAGCTTGGTCGGGGCAACACTGAAGCTTCTTT,,14,4,,
AAACCTGAGGTGCTTT-1,true,AAACCTGAGGTGCTTT-1_contig_1,true,503,TRB,TRBV7-9,,TRBJ2-1,TRBC2,true,true,DTGVSQDPHKITKRGQNVTRCQPI,GATACTGGAGTCTCCAGGACCCAGACACAAGATCAAAAGAGGGGACAGAATGTAACCTTCAGGTGTGATCCAATT,SEHNR,TCTGAACACAACCGC,LYWY
RQTLGGQPEFLTY,CTTTATTGGTACCGACAGACCCCTGGGGCAGGGCCAGAGTTTCTGACTTAC,FQNEAQ,TTCCAGAAATGAAGCTCAA,LEKSRLLSDRFSAERPKGFSFSTLEIQRTQEQGSAMYL,CTAGAAAAATCAAGGCTGCTCAGTGATCGGTTCTCTGCAGAGAGGCCCTAAGGGATCTTCTCCACCTTGAGATCCAGCGCACAGAGCAGGGGCACTCGG
CCATGTATCTC,CASSSQQFMDEQFF,TGTGCCAGCAGCTCAGGACAGTTTATGGATGAGCAGTCTTC,GPGRRLTVL,GGGCCAGGGACACGGCTCACCCTGCTAG,11348,3,clonotype19,clonotype19_consensus_1,1
AAACGGGCAGACGCCT-1,false,AAACGGGCAGACGCCT-1_contig_1,false,462,TRB,TRBV7-3,TRBD1,TRBJ2-5,,true,true,,,,,,,,,CASSPGGPEDQYF,TGTGCCAGCAGCCCGGGACCGGGCCAGAGGACCAGTACTTC,,14,4,,
AAACGGGTCGTGGACC-1,false,AAACGGGTCGTGGACC-1_contig_1,false,475,TRB,TRBV28,,TRBJ1-6,TRBC1,true,true,,,,,,,,,CASSLSGGQGGYSPLHF,TGTGCCAGCAGTTTATCCGTCAGGGCGGTTATTCACCCCTCCACTTT,,3184,1,,
AAAGATGCATTGTGCA-1,false,AAAGATGCATTGTGCA-1_contig_1,false,524,TRB,TRBV7-9,TRBD1,TRBJ2-7,TRBC2,true,true,,,,,,,,,CASSRGTGASREQYF,TGTGCCAGCAGCCCGGGACCGGGCCCTCAGAGAGCAGTACTTC,,1126,1,,
AAAGATGGTTCCACAA-1,false,AAAGATGGTTCCACAA-1_contig_1,false,404,TRA,TRAV13-2,,false,false,,,,,,,,,12,3,,
AAAGATGCACTCCTG-1,false,AAAGATGCACTCCTG-1_contig_1,false,477,TRB,TRBV5-1,,TRBJ1-5,TRBC1,true,true,,,,,,,,,CASSLTGIGPQHF,TGCGCCAGCAGCTTGACAGGATCGGGCCCCAGCATTTT,,3344,1,,
AAAGCAAAGTAACCCCT-1,false,AAAGCAAAGTAACCCCT-1_contig_1,false,654,TRB,TRBV19,,TRBJ2-7,TRBC2,true,true,,,,,,,,,CAVGQDGLYEYQF,TGTGCCGTAGGCGAGGACGACTGTACGAGCAGTACTTC,,2874,1,,
AAAGTAGAGACTTGAA-1,false,AAAGTAGAGACTTGAA-1_contig_1,false,656,TRB,TRBV5-6,,TRBJ2-3,TRBC2,true,true,,,,,,,,,CASSLEASGDTQYF,TGTGCCAGCAGCTTGAGGCTAGTGGGGATACGCACTATTTT,,1690,1,,
```

图中是使用cellranger软件进行单细胞免疫组mapping得到的初步结果，每行是一个cell barcode的mapping结果，表格的各列以逗号分隔，其中第二列包含了cellranger对该barcode是否代表真实细胞的质控判断结果（is_cell = true/false）

使用linux命令组合统计 is_cell 这一列为true和false的barcode个数：

```
cat all_contig_annotations.csv | tail -n+2 | cut -d',' -f2 | sort | uniq -c
```

代码解释：打印all_contig_annotations.csv中的全部内容 | 从第二行开始打印 | 按逗号分割每一列，只打印第二列 | 排序 | 去重复的同时输出重复个数

```
(base) [big@training:~/Linux_and_R]$ cat all_contig_annotations.csv | tail -n+2 | cut -d',' -f2 | sort | uniq -c
874 false
45 true
```

04

R语言与R studio入门

R语言概览

R语言是一种用于统计分析、数据可视化和科学计算的开源编程语言和软件环境，广泛应用于生物信息学分析。它拥有丰富的内置统计函数、灵活的数据结构和庞大的扩展生态，支持可重现研究与交互式数据探索。



开源自由	统计功能丰富	可视化强大	数据结构多样	生态系统庞大	交互与扩展
R采用GPL协议，用户可以免费获取、使用和修改源代码	支持时间序列、生存分析和贝叶斯建模等统计分析	提供灵活的低级绘图接口，满足自定义需求	向量与矩阵高效处理数值计算任务	Bioconductor 专门支持多种生物信息学工作流	Shiny框架可快速构建交互式Web应用，可视化程度高
社区驱动的开发模式确保新功能和补丁能够快速合并，透明度高、无厂商锁定	内置从基本描述统计到高级回归、方差分析、聚类、主成分分析等上百种方法	ggplot2 实现“语法图形学”，通过分层语法快速生成美观的统计图	列表 (list) 可嵌套任意类型对象，适合组织和管理多层次、结构化的复杂分析结果	社区活跃，定期举办 R 会议和用户组 (UseR!、R-Ladies) 分享心得	RStudio IDE 集成代码补全、调试、可视化预览和 Git 支持，极大提升开发效率
拥有全球活跃的用户社区和贡献者网络，新手和专家都可在论坛、邮件列表和Stack Overflow 上获得支持	强大的扩展包如 forecast、survival 和 bayesm 提供专业领域的深度统计工具，满足复杂科研需求	plotly、leaflet 等包支持交互式 Web 可视化，无缝嵌入 Shiny 应用或 R Markdown	数据框 (data.frame) 和 tibble 支持异质列，方便进行分组、汇总和管道式数据处理	CRAN 上超过 2 万个包，涵盖数据清洗 (dplyr)、可视化 (ggplot2)、机器学习 (caret、mlr3) 等	支持与 Python、C++、SQL 等多语言互操作，利用 reticulate、Rcpp 等包扩展性能和功能

R语言的官方网站：
<http://www.r-project.org/>

R语言推荐中文教程：
https://www.math.pku.edu.cn/teachers/lidf/docs/Rbook/html/_Rbook/index.html

R扩展软件包的安装与管理

R包管理涵盖 CRAN/Bioconductor 安装、GitHub 私有源获取、renv/packrat 依赖隔离与命名空间加载，保障科研流程中扩展生态的高效可重现

从CRAN安装与卸载

01

1. 安装最新稳定版本

- `install.packages("包名")`

2. 指定镜像源 -> 加速国内下载 (清华镜像)

- `install.packages("包名", repos = "https://mirrors.tuna.tsinghua.edu.cn/CRAN/")`

3. 卸载包

- `remove.packages("包名")`

4. 查看已安装包

- `installed.packages()` -> 返回所有包和版本信息
- `sessionInfo()$otherPkgs` -> 当前会话已加载的包

5. 更新包

- `update.packages(ask=FALSE)` -> 更新所有过时包

Bioconductor平台

02

1. 安装BiocManager (仅需一次)

- `install.packages("BiocManager")`

2. 通过BiocManager安装/更新

(1) 安装单个包

- `BiocManager::install("DESeq2")`

(2) 批量安装多个包

- `BiocManager::install(c("edgeR", "limma"))`

(3) 更新已安装的所有的Bioconductor包

- `BiocManager::install()`

3. 常用生信包示例

- `GenomicRanges`: 基因组区间处理
- `AnnotationDbi`: 注释数据库接口

GitHub平台

03

1. 使用remotes包

- `install.packages("remotes")`
- `remotes::install_github("用户名/仓库名")`

2. 使用devtools包

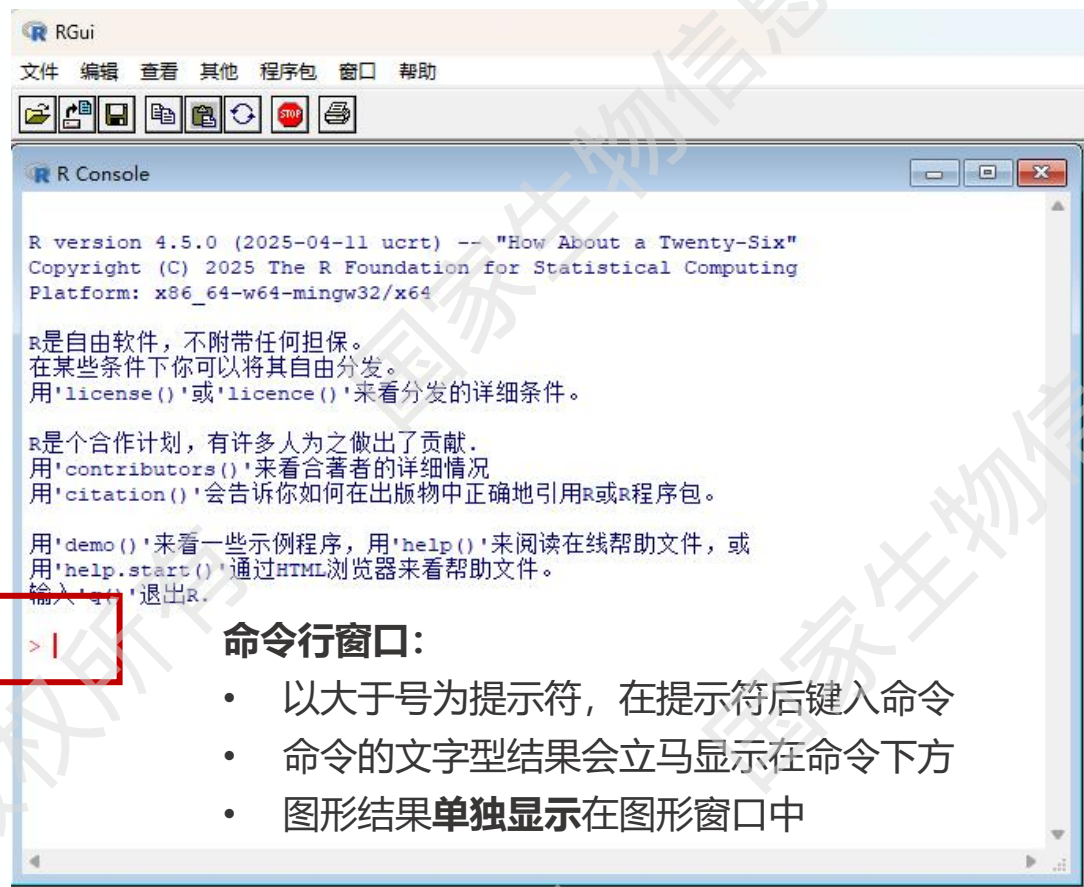
- `install.packages("devtools")`
- `devtools::install_github("用户名/仓库名", ref = "branch_or_tag")`

3. 安装私有仓库或子目录

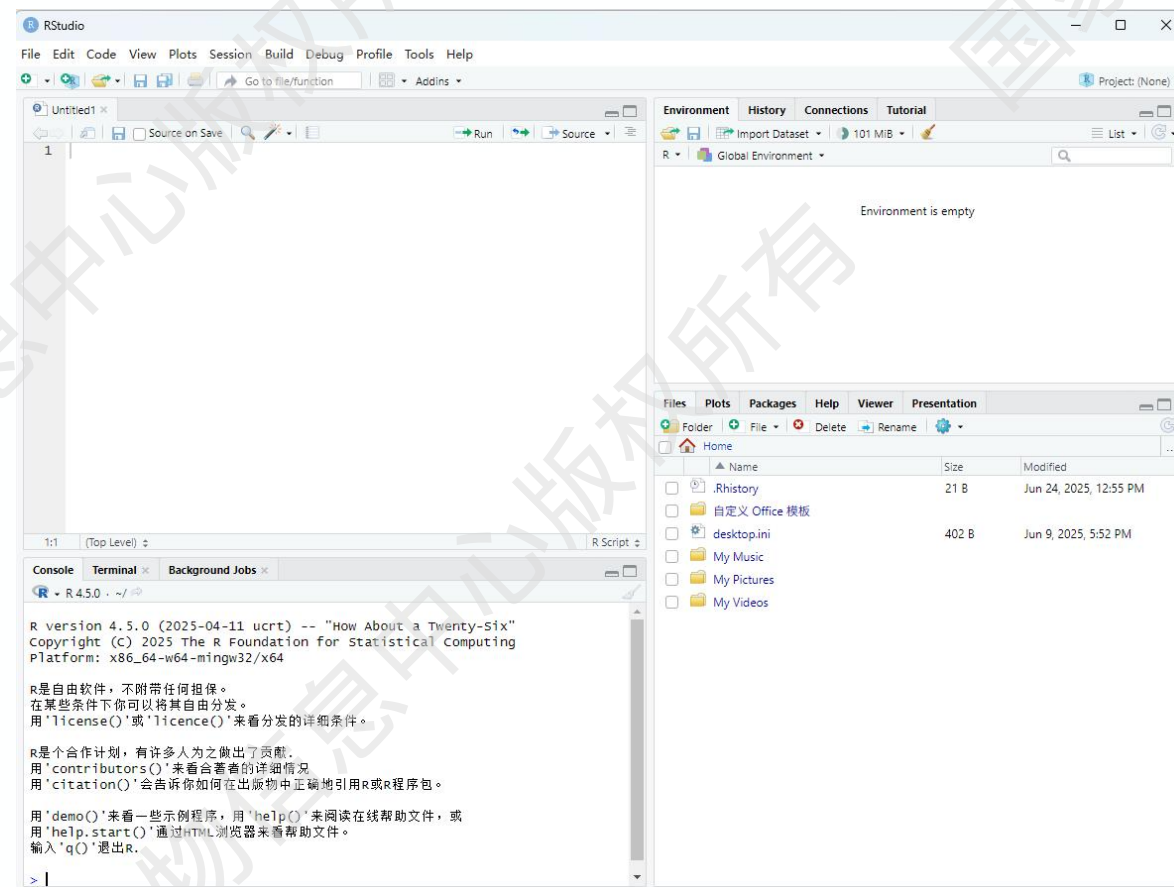
- `remotes::install_git("git@github.com:org/privatepkg.git")`
- `remotes::install_github("org/repo/subdir/pkg")`

为什么选择R studio?

RStudio提供了集成的开发环境，将代码编辑、控制台、变量/文件浏览和可视化输出无缝整合，并配备智能补全、调试工具与版本控制接口，大幅提升 R 语言脚本的编写、调试和项目管理效率



图形窗口模式的R软件 (R GUI)



R Studio软件 (软件的应用界面与增强系统)

R studio界面四大窗口

通过脚本编辑、控制台与终端、环境与历史以及文件/绘图/包/帮助/查看这四个互联窗口，R Studio 将脚本编辑、命令交互、环境管理和可视化展示无缝集成，大幅提升了 R 语言开发与分析的效率与体验

The screenshot displays the RStudio interface with four main panes:

- Source Editor (Top Left):** Contains R code for analyzing TCR data. A red box highlights the "Run" button. A red annotation states: "R代码脚本区，可以用鼠标选中，并一次性执行多行代码" (R code script area, can be selected with the mouse, and execute multiple lines of code at once).
- Environment/History (Top Right):** Shows the current environment with variables like "clonalQuant" and "scRepertoire". A red box highlights the "Help" button. A red annotation states: "帮助文档区" (Help documentation area).
- Console/Terminal (Bottom Left):** Shows the execution of R commands and their output. A red annotation states: "R语言交互区，类似单独的R语言打开的界面，一次性只能执行一行代码" (R language interaction area, similar to a separate R language interface, can only execute one line of code at a time).
- Plots/Presentation (Bottom Right):** Displays a bar chart titled "Quantify the unique clones by group or sample". The chart shows the percentage of unique clones for various samples. A red box highlights the "Zoom" and "Export" buttons. A red annotation states: "绘图区，可放大查看，可导出成png文件，或者用代码直接保存高分辨率的图片文件。" (Plotting area, can be zoomed in for viewing, can be exported as a png file, or can be saved as a high-resolution image file using code).

The bar chart data is as follows:

Samples	Percent of Unique Clones
P17B	~25
P17L	~75
P18B	~95
P18L	~95
P19B	~80
P19L	~60
P20B	~70
P20L	~95

05

使用data.table 进行数据处理

数据导入与导出

文本文件



- read.table()
- read.csv() -> csv专用
- read.delim() -> 分隔符

快速读入大数据



- data.table::fread()
- data.table包配备的fread函数具备自动检测分隔符、并行读取的功能，对大型文件的读取非常好。

Excel文件与R原生格式



- readxl::read_excel() -> excel格式
- readRDS() -> 由saveRDS保存的单个R对象

数据
导入

数据
导出

文本文件



- write.table()
- write.csv() -> csv专用

快速写入大数据



- data.table::fwrite()
- 备注：高速、支持压缩写入 (.csv.gz)

Excel文件与R原生格式



- openxlsx::write.xlsx()
- saveRDS() -> 将任意R对象序列化为.rds文件

data.table用于快速处理大型数据框（即二维表格）

data.table快速入门教程：

<https://youngspring1.github.io/categories/data-table%E6%95%99%E7%A8%8B/>

b) 形式 - data.table增强了什么

和data.frame相反，我们能做的可不仅仅局限于subset行或者select列。首先介绍下data.table的语法，如下所示：

```
DT[i, j, by]

## R:      i      j      by
## SQL: where select | update group by
```

如果你有SQL语句的基础，那么你应该能马上明白data.table的语法。

语法是：

对于DT这个data.table，使用 i 来subset行，然后计算 j，最后用 by 分组。

```
library(data.table)
```

```
# 读取
```

```
mydata = fread("volcano.txt")
```

```
# 使用data.table对二维表格进行取子集、排序、删除某一列、修改某一列、新增某一列
```

```
# 取子集
```

```
mydata
```

```
mydata[log2FoldChange >= 1]
```

```
mydata[padj <= 0.01]
```

```
mydata[log2FoldChange >= 1 & padj <= 0.01]
```

```
mydata[log2FoldChange >= 1][padj <= 0.01]
```

```
mydata[, .(gene_name)]
```

```
mydata[log2FoldChange >= 1, .(gene_name)]
```

```
mydata[log2FoldChange >= 1][padj <= 0.01][, .(gene_name)]
```

```
# 排序
```

```
mydata
```

```
setorder(mydata, log2FoldChange)
```

```
setorder(mydata, -log2FoldChange)
```

```
setorder(mydata, padj)
```

```
setorder(mydata, -log2FoldChange, padj)
```

```
# 删除某一列
```

```
mydata[, padj:=NULL]
```

```
# 修改某一列
```

```
mydata[, log2FoldChange:=1]
```

```
# 新增某一列
```

```
mydata[, group:="control"]
```

06

使用ggplot2 进行科研绘图

ggplot2语法结构

ggplot2 是 R 语言中最常用的可视化系统，基于“语法图形学（Grammar of Graphics）”，强调将数据映射到图形元素（图层 Layer）上，逐层构建图像

基本构图流程

1、添加基础图对象

```
p <- ggplot(data = df, aes(x = xvar,  
y = yvar))
```

2、添加图层函数 (geom_*())

```
p + geom_point() # 散点图  
p + geom_bar(stat = "identity") #  
柱状图  
p + geom_boxplot() # 箱线图
```

3、添加标签与主题

```
p + labs(title = "主标题", subtitle =  
"副标题", x = "X轴名称", y = "Y轴名  
称") + theme_minimal()
```

语法结构

核心语法结构

```
ggplot(data = <数据框>, aes(x =  
<x变量>, y = <y变量>)) +  
  <图层函数> +  
  <主题或标签设置>
```

- 1、data: 数据来源，必须是数据框；
- 2、aes(): 美学映射，如x/y轴、颜色等；
- 3、geom_*(): 图层函数，定义类型；
- 4、labs(): 设置标题、副标题、轴标签；
- 5、theme(): 美化图标风格，如字体等。

gg
plot2

构图流程

图层函数

常用图层函数介绍

- 1、geom_point(): 绘制散点图；
- 2、geom_bar(): 绘制柱状图；
- 3、geom_line(): 绘制折线图；
- 4、geom_boxplot(): 绘制箱线图；
- 5、geom_histogram(): 绘制直方图。

ggplot2详细教程，作者亲自撰写，强烈推荐精读

ggplot2的官方教程：
<https://ggplot2-book.org/>

ggplot2: Elegant Graphics for Data Analysis (3e)

Q

[Welcome](#)

[Preface to the third edition](#)

[Preface to the second edition](#)

[Getting Started](#)

1 [Introduction](#)

2 [First steps](#)

[Layers](#)

3 [Individual geoms](#)

4 [Collective geoms](#)

5 [Statistical summaries](#)

6 [Maps](#)

7 [Networks](#)

8 [Annotations](#)

9 [Arranging plots](#)

[Scales](#)

10 [Position scales and axes](#)

11 [Colour scales and legends](#)

12 [Other aesthetics](#)

[The Grammar](#)

13 [Build a plot layer by layer](#)

14 [Scales and guides](#)

15 [Coordinate systems](#)

16 [Faceting](#)

17 [Themes](#)

[Advanced topics](#)

18 [Programming with ggplot2](#)

19 [Internals of ggplot2](#)

20 [Extending ggplot2](#)

21 [A case study](#)

ggplot2: Elegant Graphics for Data Analysis (3e)

Welcome

This is the on-line version of work-in-progress **3rd edition** of “ggplot2: elegant graphics for data analysis” published by Springer. You can learn what’s changed from the 2nd edition in the [Preface](#).

While this book gives some details on the basics of ggplot2, its primary focus is explaining the Grammar of Graphics that ggplot2 uses, and describing the full details. It is not a [cookbook](#), and won’t necessarily help you create any specific graphic that you need. But it will help you understand the details of the underlying theory, giving you the power to tailor any plot specifically to your needs.

The book is written by [Hadley Wickham](#), [Danielle Navarro](#), and [Thomas Lin Pedersen](#).

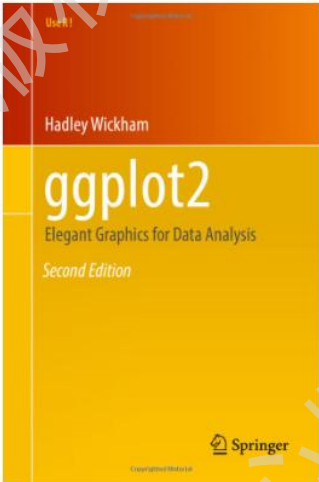


Table of contents

[Welcome](#)

Edit this page

Report an issue

Preface to the third edition →

综合案例：火山图

该示例综合运用了各种ggplot函数，演示了如何绘制差异表达基因的火山图

画出火山图

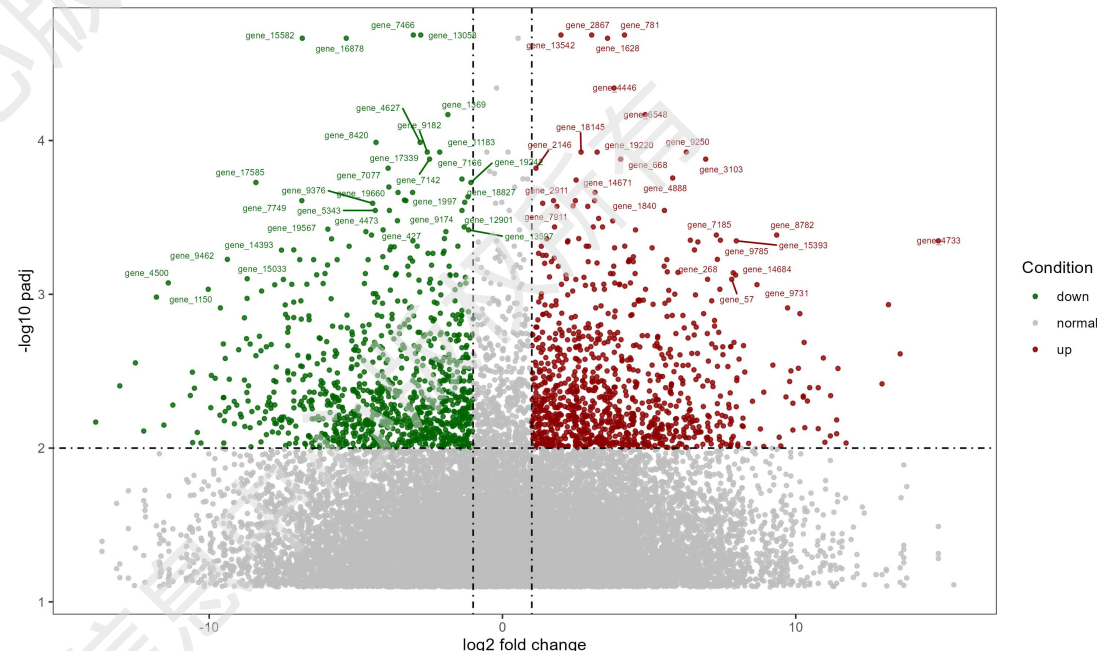
```
p = ggplot(data=mydata, aes(x=log2FoldChange, y=-log10(padj), colour=Condition)) +  
  geom_point(alpha=0.8, size=1) +  
  xlab("log2 fold change") +  
  ylab("-log10 padj") +  
  geom_hline(yintercept=-log10(0.01), linetype=4) +  
  geom_vline(xintercept=c(-1,1), linetype=4) +  
  scale_color_manual(values=c('up'='darkred', 'down'='darkgreen', 'normal'='gray')) +  
  theme_bw() +  
  theme(panel.grid = element_blank())
```

把基因名添加到火山图上

```
p = p +  
  geom_text_repel(data=mydata,aes(x=log2FoldChange,  
    y=-log10(padj),label=label),size=2)
```

保存图片

```
ggsave("./volcano.png", p, height = 6, width = 10)
```



ggplot2语法结构

ggplot2 通过 `aes()` 映射数据、`geom_*()` 构建图层、`labs()` 添加注释、`theme()` 美化风格，采用图层叠加的语法体系，实现灵活、可复用的可视化构图

美学映射 + 图层叠加

`aes(x, y, color, fill, shape, size, alpha, group)`

- color/fill: 控制颜色;
- shape/size: 控制点的形状和大小;
- alpha: 控制透明度
- group: 用于分组 (如线条分类)

组合多个几何图层

`ggplot(df, aes(x = x, y = y)) +`

`geom_point() +`

`geom_smooth(method = "lm")`

添加回归线

标签与主题美化

(1) 添加标题、标签

`+ labs(title = "图主标题", subtitle = "副标题", x = "X轴", y = "Y轴", caption = "数据来源")`

(2) 修改图形主题

`+ theme_bw()` # 黑白主题 (带有清晰网格线)

`+ theme_minimal()` # 极简主题 (无背景和坐标轴线)

`+ theme_classic()` # 经典主题 (去除灰色背景)

(3) 主题细节定制 (可选)

`+ theme(plot.title = element_text(hjust = 0.5, size = 16, face = "bold"), axis.text.x = element_text(angle = 45, hjust = 1))`

案例演示

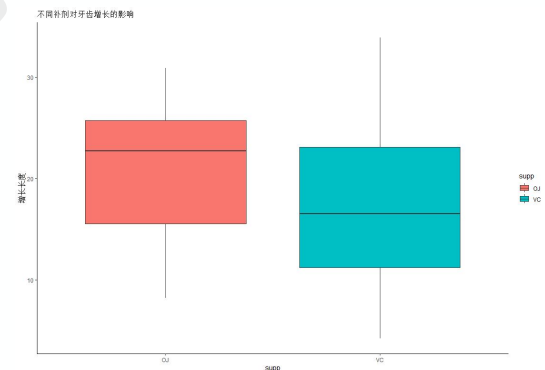
分组箱线图:

`ggplot(ToothGrowth, aes(x = supp, y = len, fill = supp))`

`+ geom_boxplot() +`

`labs(title = "不同补剂对牙齿增长的影响", y = "增长长度")`

`+ theme_classic()`



常用图层介绍: geom_point()

geom_point() 用于绘制二维平面中的散点图，每个点代表一个观测值，位置由 x 和 y 值决定，可用于查看变量之间的关系、分布、聚类等

ggplot(data = 数据框, aes(x = 变量1, y = 变量2)) + geom_point() ➡ 增加散点图层

↓

aes中的x、y分别表示坐标轴

参数详解

- 1、mapping：设置映射关系；
- 2、color：控制点的边框颜色；
- 3、fill：控制点的填充颜色；
- 4、size：控制点的大小；
- 5、alpha：控制点的透明度（0-1）；
- 6、shape：控制点的形状。

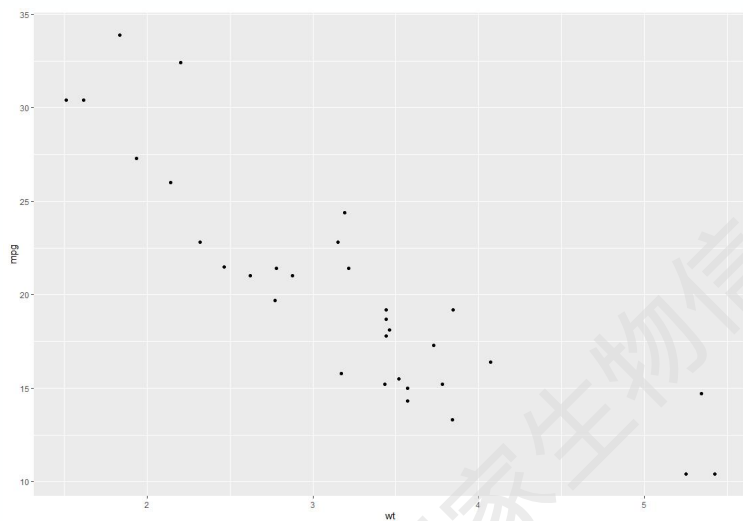
编号	描述
16	实心圆（默认）
17	实心三角形
18	实心菱形
19	实心圆+边框
21	空心圆
22	空心方形

美学映射

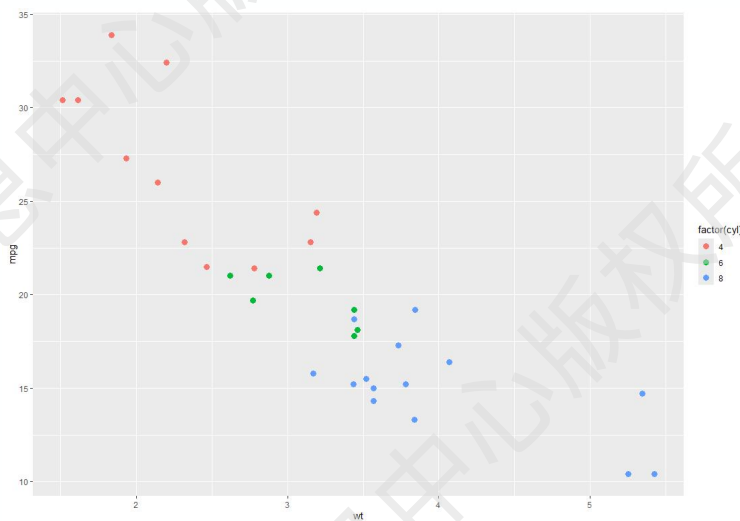
ggplot(data, aes(x = var1, y = var2, color = group, size = value)) + geom_point()

- 1、color：映射分类变量；
aes(color = group)
- 2、size：映射连续变量；
aes(size = measurement)
- 3、shape：区分组别；
aes(shape = type)
- 4、alpha：设置透明度。
aes(alpha = confidence)

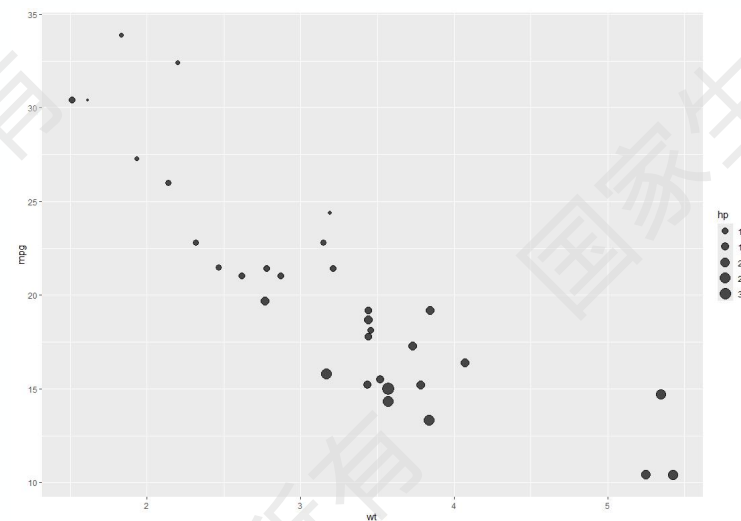
geom_point()的常见用法与案例展示



示例1



示例2



示例3

基本散点图

```
ggplot(mtcars, aes(x = wt, y = mpg)) +  
  geom_point()
```

X轴为车重 (wt) , Y轴为油耗 (mpg)

颜色按组分层

```
ggplot(mtcars, aes(x = wt, y = mpg,  
  color = factor(cyl))) + geom_point(size = 3)
```

根据气缸数 (cyl) 设置不同颜色

点大小映射连续变量

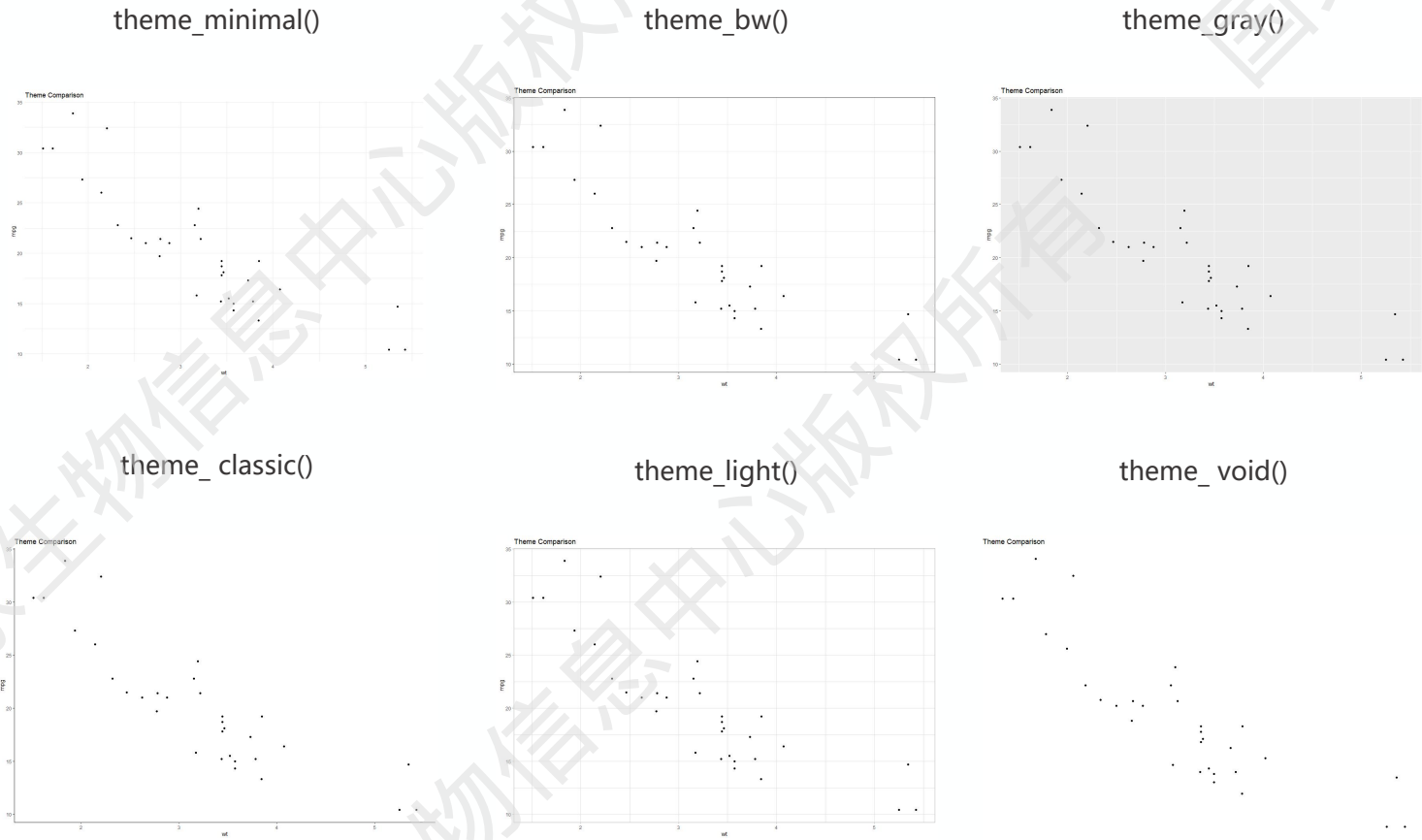
```
ggplot(mtcars, aes(x = wt, y = mpg,  
  size = hp)) + geom_point(alpha = 0.7)
```

马力 (hp) 控制点的大小, 增加可视信息

图形主题设置: theme_*

theme_*(**)** 提供预设风格模板，适用于不同的展示需求，是图表专业美化的首选工具，可与 theme(**)** 联合实现风格统一与个性调整

函数名称	风格特征	典型用途
theme_minimal()	极简设计、无边框、无背景、仅保留轴线	幻灯片展示、现代报告
theme_classic()	白底、明确的横纵坐标轴线、无背景色	学术出版、图书图示
theme_bw()	黑白强对比，带细网格线，清晰可打印	论文附件、黑白打印
theme_light()	类似 minimal，但有灰色边框和淡淡网格线	科研报告、演讲演示
theme_gray() (默认)	灰色背景+白网格线，ggplot2 默认风格	一般可视化展示
theme_void()	空白画布，无坐标轴线、无标签	自定义图像、地图展示



主题细节自定义：theme()

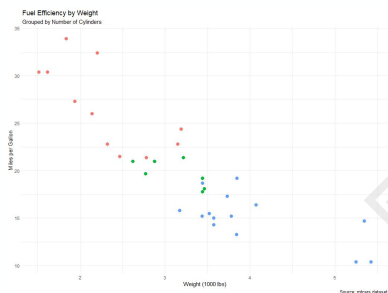
theme() 是 ggplot2 的“美工系统”，通过精细化控制图形各要素的样式，使图表在保持信息清晰的基础上达到专业与美观的统一

参数名	控制内容	常用设置示例
plot.title	主标题字体、大小、位置	element_text(hjust=0.5, size=16, face="bold")
plot.subtitle	副标题样式	element_text(size=12, hjust=0.5)
axis.title.x / .y	坐标轴标题样式	element_text(size=14, face="italic")
axis.text.x / .y	坐标轴刻度样式	element_text(angle=45, hjust=1)
legend.position	图例位置	"top", "bottom", "left", "none"
legend.title	图例标题字体样式	element_text(face = "bold")
legend.text	图例标签字体样式	element_text(size = 10)
panel.grid.major	主网格线	element_line(color = "gray90")
panel.grid.minor	次网格线	element_blank()（可隐藏）
panel.background	图形绘图区背景	element_rect(fill = "white")
plot.caption	图注样式	element_text(size = 9, color = "gray50")

主题细节自定义: theme()

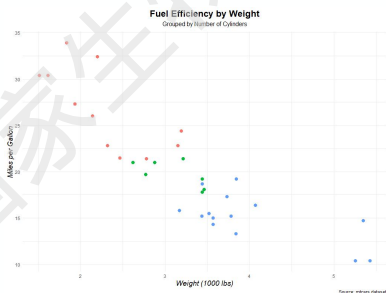
theme() 是 ggplot2 的“美工系统”，通过精细化控制图形各要素的样式，使图表在保持信息清晰的基础上达到专业与美观的统一

套用预设主题



```
ggplot(mtcars, aes(x = wt,  
y = mpg, color = factor(cyl)))  
+  
geom_point(size = 3) +  
labs(  
title = "",  
subtitle = "",  
x = "",  
y = "",  
color = "",  
caption = "")
```

调整多种标题



```
p_theme + theme(  
plot.title =  
element_text(hjust = 0.5, face  
= "bold", size = 18),  
plot.subtitle =  
element_text(hjust = 0.5, size  
= 12, margin = margin(b =  
10)),  
axis.title =  
element_text(size = 14, face =  
"italic"))
```

处理刻度文字



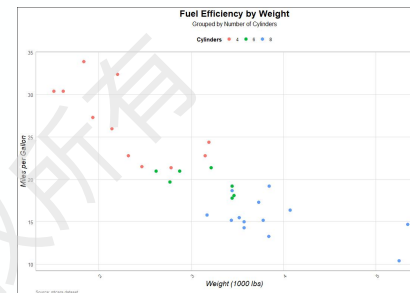
```
p_step2 + theme(  
axis.text.x =  
element_text(angle = 45, hjust  
= 1, size = 10),  
axis.text.y =  
element_text(size = 10),  
panel.grid.major =  
element_line(color = "gray85"),  
panel.grid.minor =  
element_blank())
```

调整图例图注



```
p_step3 + theme(  
legend.position = "top",  
legend.title =  
element_text(face = "bold"),  
legend.text =  
element_text(size = 9),  
plot.caption =  
element_text(size = 8, color =  
"gray50", hjust = 0)  
)
```

调整背景边框



```
p_step4 + theme(  
panel.background =  
element_rect(fill = "white"),  
panel.border =  
element_rect(color = "gray70",  
fill = NA, linewidth = 0.8),  
plot.background =  
element_rect(fill = "white")  
)
```

使用ggsave()保存单个图表

ggsave()是ggplot2包提供的一个非常方便的图表保存函数。它支持多种文件格式（如 PNG、PDF、JPEG、SVG 等），可以将绘制的图表保存到指定的路径。

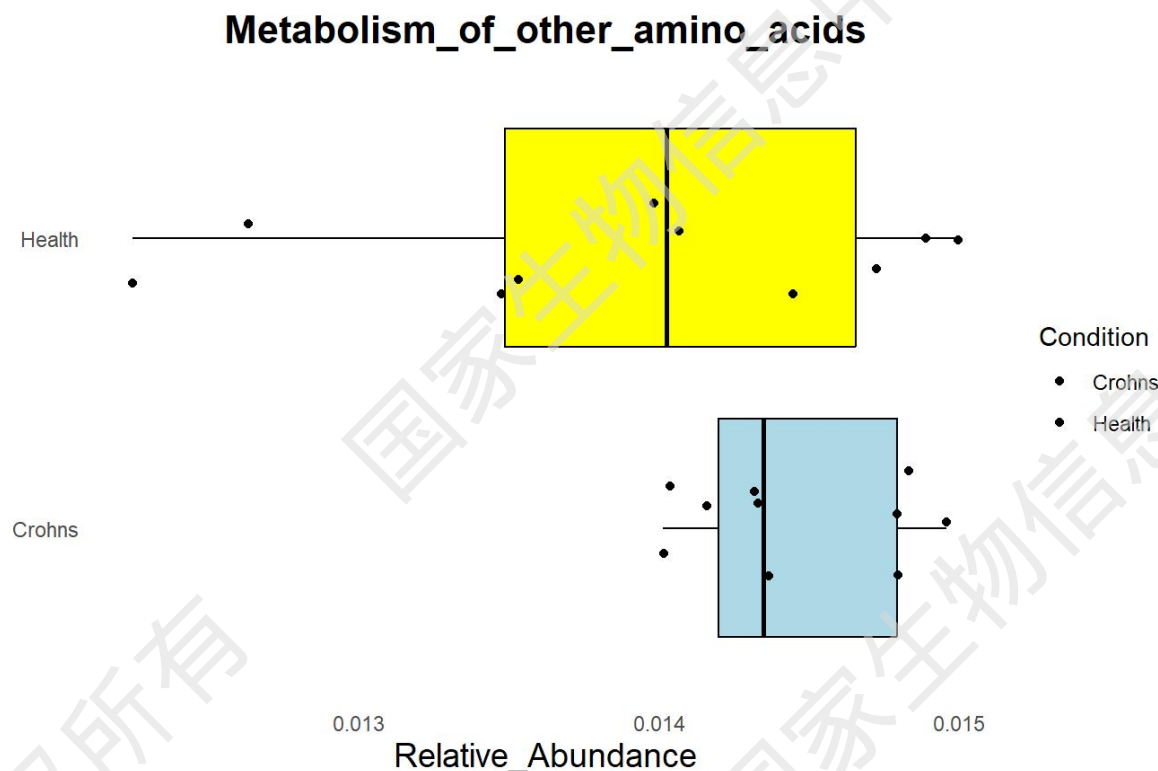
ggsave(filename, plot, width, height, units, dpi, ...)

- filename: 保存文件的路径和名称。可以指定文件格式，如 .png、.pdf、.jpeg 等。
- plot: 要保存的图表对象。通常是通过 ggplot() 绘制的图表，或者是已经绘制好的图表对象（如 p）。
- width 和 height: 指定图表的宽度和高度，单位可以是英寸、厘米或毫米。
- units: 指定单位（in 表示英寸，cm 表示厘米，mm 表示毫米）。
- dpi: 图像的分辨率（通常为 300 DPI），适用于需要打印的图表。
- ...: 其它参数，比如图表的背景颜色、格式等。

**根据文件的扩展名，ggsave()
自动识别要保存的格式，并将
图表保存为指定格式的文件**

科研常用图形案例1：水平箱线图

通过绘制箱线图比较不同组别的相对丰度（或其它数值变量），便于展示数据的分布、离群点等

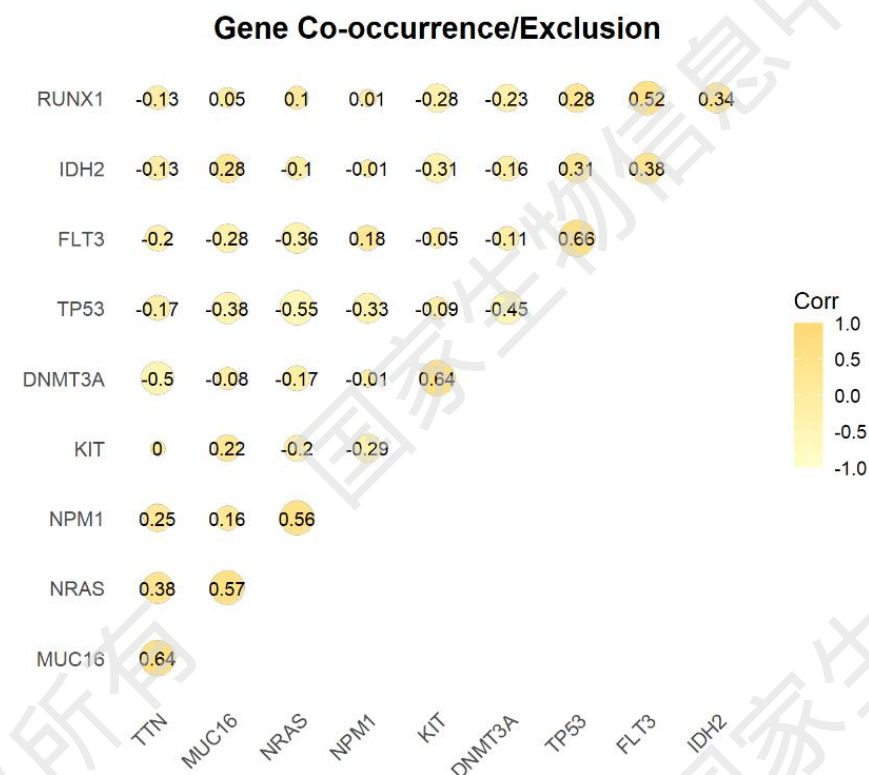


- 使用 `ggplot()` 函数映射数据，定义 x 和 y 轴，并通过 `fill` 为不同组别指定颜色。
- 通过 `geom_boxplot()` 绘制箱线图，设置颜色、边框等属性。
- 使用 `geom_jitter()` 添加散点，帮助展示个体数据。
- 自定义图表样式、坐标轴、标签等，通过 `theme_minimal()`、`theme()` 等函数实现。

基因表达数据、环境数据等比较分析

科研常用图形案例2：相关性图

绘制基因、变量或样本之间的相关性矩阵图，显示它们之间的相互关系

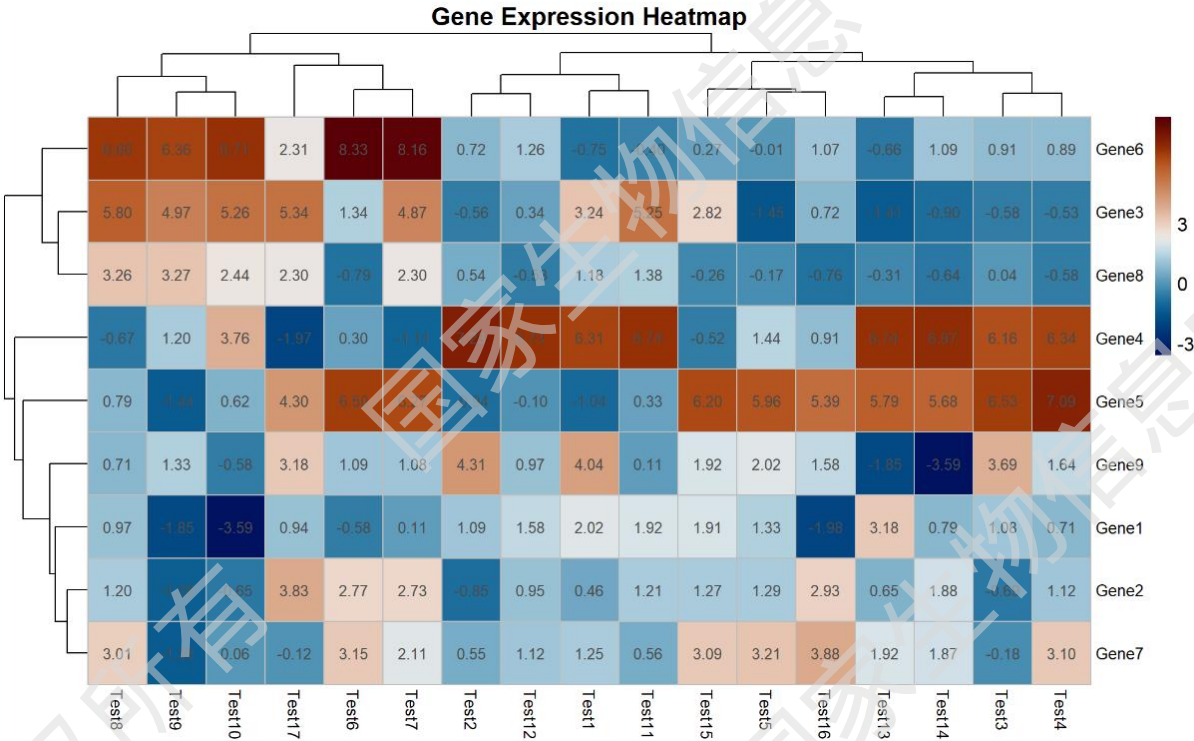


- 使用cor()函数计算数据框中的皮尔逊相关性，得到相关性矩阵。
- 使用ggcorrplot()函数绘制相关性热图，设置颜色、样式、标签等。
- 通过scale_color_manual()定义颜色梯度，确保图表信息准确传达。
- 设置标题、图例、坐标轴标签等，确保图表易读。

基因表达数据的相关性分析，样本间关系分析

科研常用图形案例3：聚类热图

通过热图展示基因、样本等之间的相似性，采用颜色渐变显示数值范围

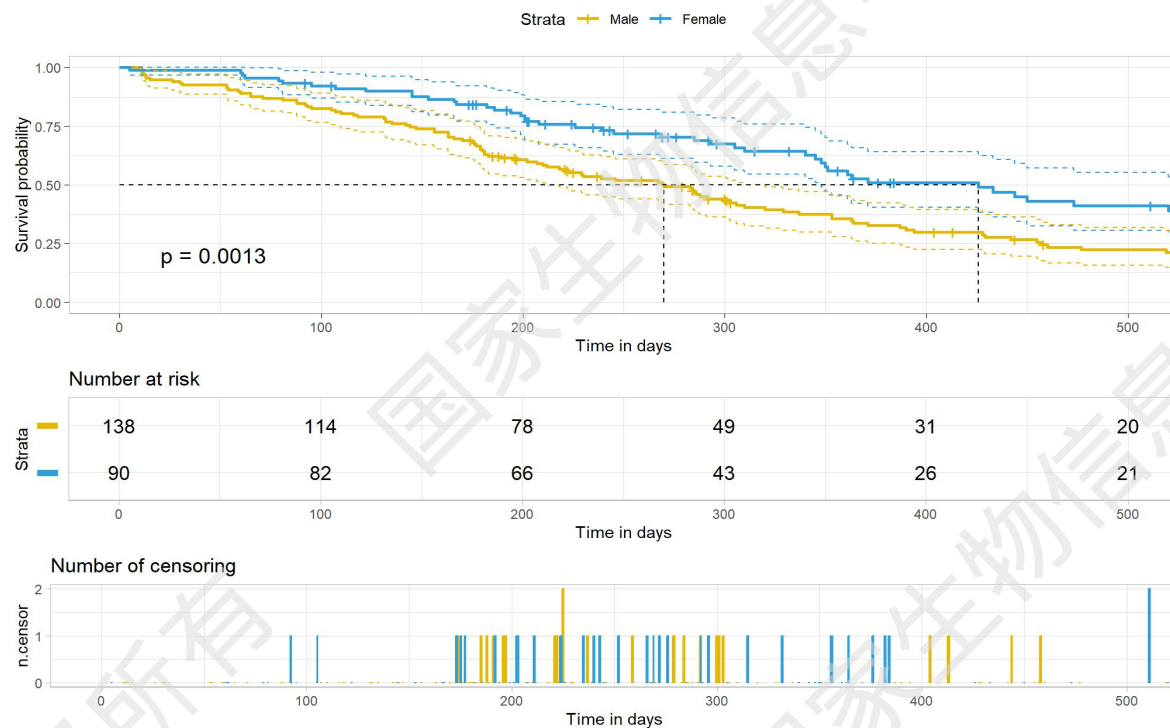


- 使用 `pheatmap()` 函数绘制热图，支持聚类、颜色渐变和数值显示。
- 通过 `cluster_rows` 和 `cluster_cols` 设置行列的聚类方式。
- 使用 `color` 设置颜色方案，通常配合 `RColorBrewer` 或 `scico` 包中的配色方案。
- 显示图例和坐标轴标签，调整字体和边框颜色。

基因表达数据、临床数据聚类分析结果的热图展示

科研常用图形案例4：生存曲线图

展示生存分析结果，通常使用 Kaplan-Meier 方法进行分析，能够比较不同组别的生存概率

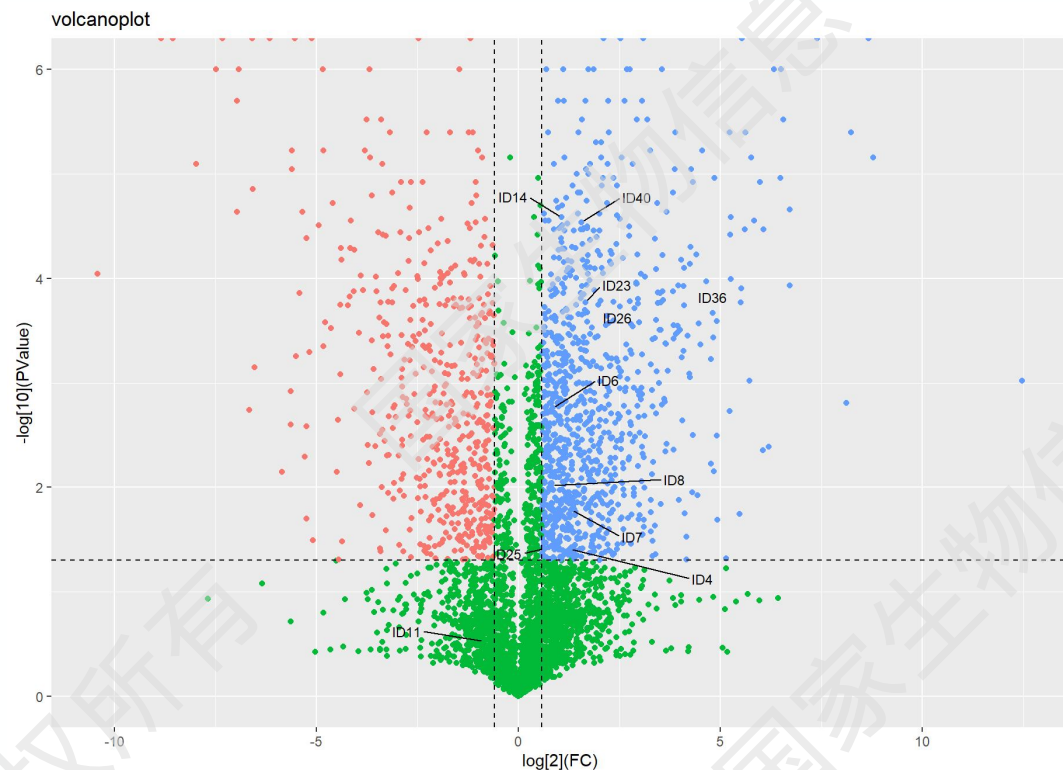


- 使用 `survfit()` 函数拟合生存模型，依据某一因素（如性别、治疗方法等）进行分组。
- 使用 `ggsurvplot()` 函数绘制 Kaplan-Meier 生存曲线图，展示每组的生存概率。
- 自定义图例、风险表、置信区间、p 值等内容。

临床数据的生存分析，药物治疗效果评估

科研常用图形案例5：火山图

通过火山图展示基因表达分析中上调和下调基因的显著性，通常结合 $\log_2FC$ 和 p 值

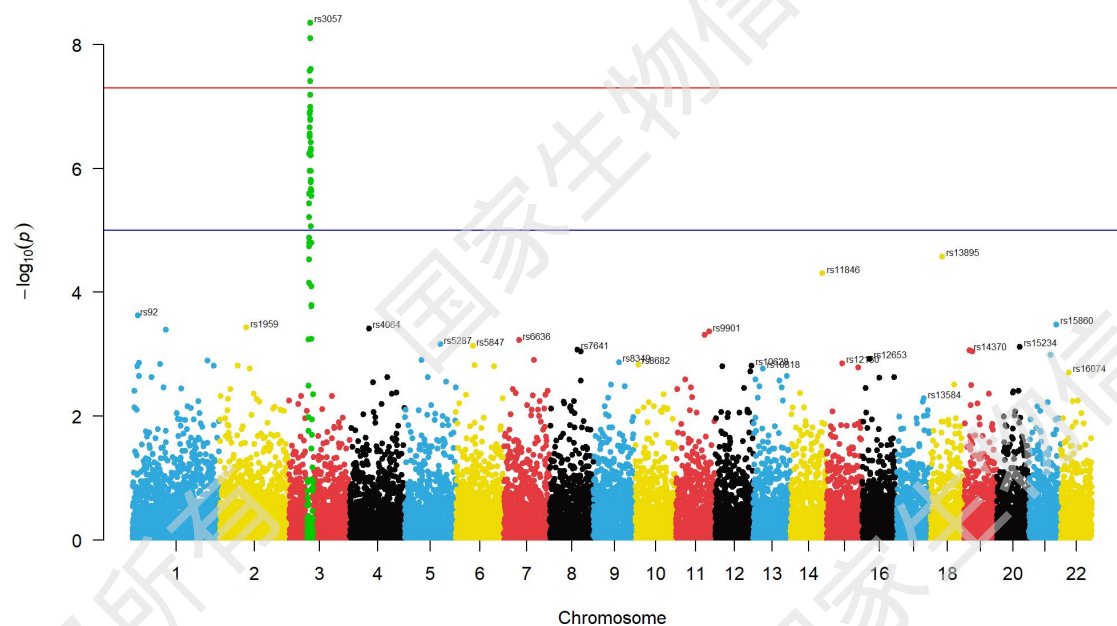


- 使用 ggplot2 绘制火山图，x 轴为 $\log_2(FC)$ ，y 轴为 $-\log_{10}(P\text{-value})$ 。
- 设置阈值：通过 `geom_vline()` 和 `geom_hline()` 添加显著性阈值（如 $\log_2FC$ 、 $P\text{-value}$ ）。
- 使用 `geom_text_repel()` 给显著的基因加上标签，避免标签重叠。

基因差异表达分析，药物靶点筛选

科研常用图形案例6：曼哈顿图

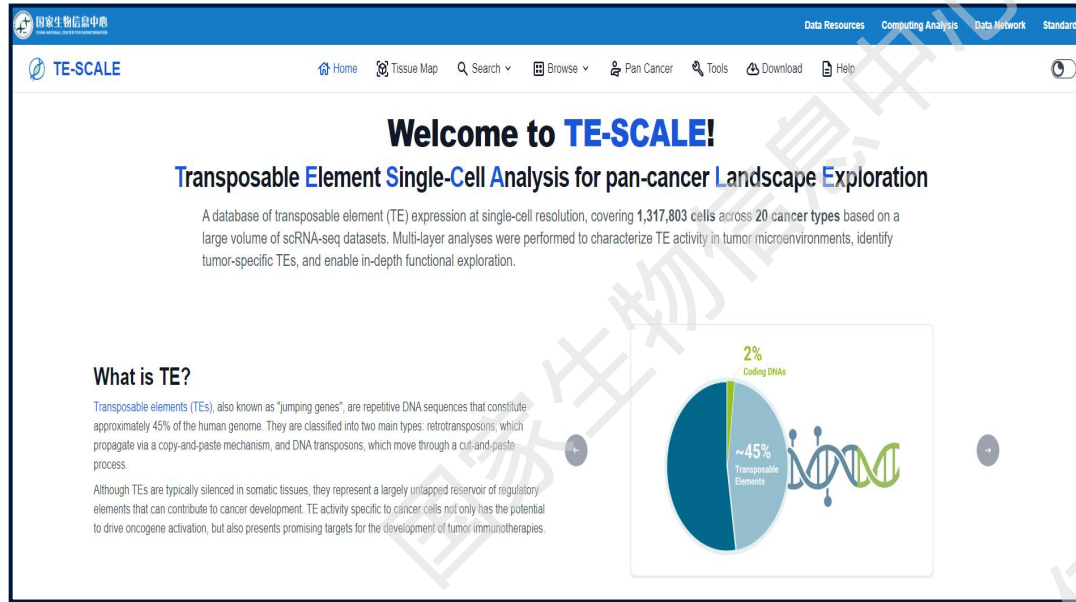
在 GWAS（基因组宽度关联研究）中用于展示 SNP 与性状之间的关联性，显示显著的 SNP



- 使用 `manhattan()` 函数绘制曼哈顿图，输入包含 SNP 名称、染色体编号、碱基位置和 p 值的数据。
- 使用 `suggestiveline` 和 `genomewideline` 添加显著性阈值的横线。
- 根据 P 值标记显著的 SNP 并进行注释。

GWAS 数据可视化，用于基因组关联分析

TE-SCALE: a comprehensive database for exploring transposable element expression across human cancers at single-cell resolution



国家生物信息中心

TE-SCALE

Home Tissue Map Search Browse Pan Cancer Tools Download Help

Welcome to TE-SCALE!

Transposable Element Single-Cell Analysis for pan-cancer Landscape Exploration

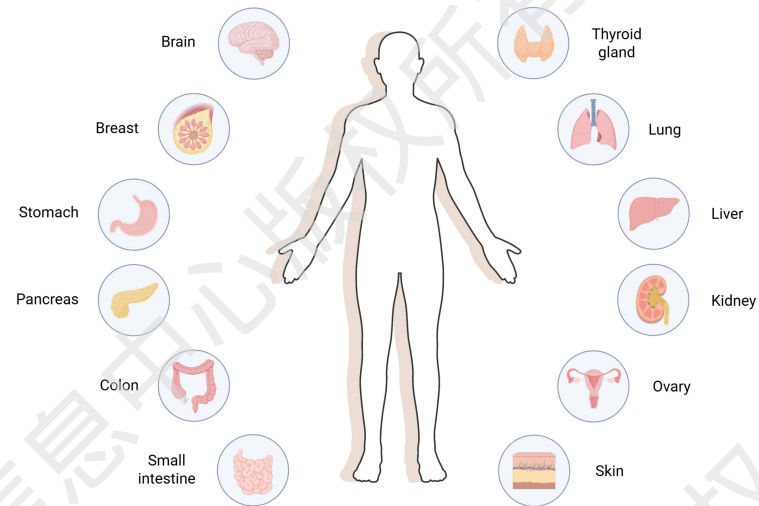
A database of transposable element (TE) expression at single-cell resolution, covering 1,317,803 cells across 20 cancer types based on a large volume of scRNA-seq datasets. Multi-layer analyses were performed to characterize TE activity in tumor microenvironments, identify tumor-specific TEs, and enable in-depth functional exploration.

What is TE?

Transposable elements (TEs), also known as "jumping genes", are repetitive DNA sequences that constitute approximately 45% of the human genome. They are classified into two main types: retrotransposons, which propagate via a copy-and-paste mechanism, and DNA transposons, which move through a cut-and-paste process.

Although TEs are typically silenced in somatic tissues, they represent a largely untapped reservoir of regulatory elements that can contribute to cancer development. TE activity specific to cancer cells not only has the potential to drive oncogene activation, but also presents promising targets for the development of tumor immunotherapies.

2% Coding DNAs
~45% Transposable Elements



Data curation

330 Datasets
1,317,803 Cells
20 Cancers
12 Tissue origins

scTEfinder

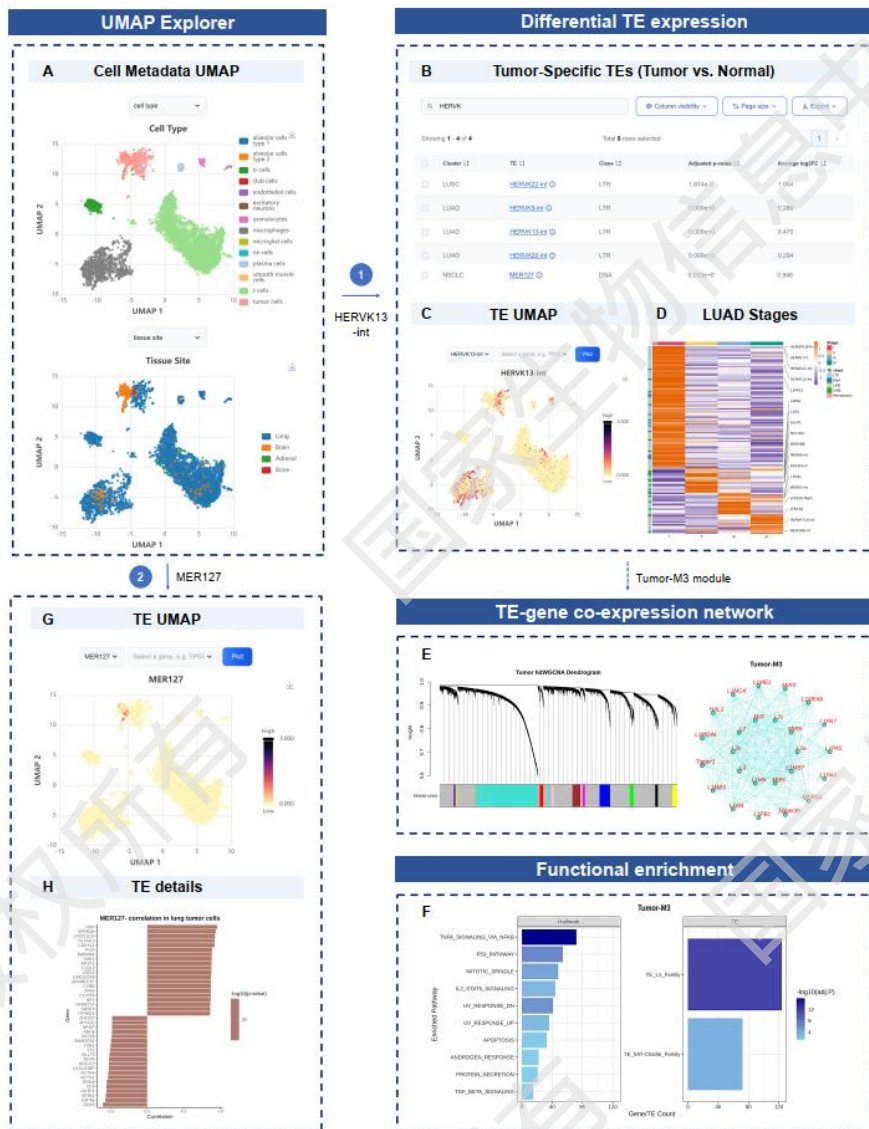
1,051 TE subfamilies

<https://ngdc.cncb.ac.cn/te-scale/>

Accepted by *Nucleic Acids Research* (IF=13.1)

- TE-SCALE: **首个**能够在**单细胞水平**上实现对人类**癌症中TE**表达进行系统性探索的综合数据库，**网站已有43278次点击**。
- 该数据库涵盖了来自20种癌症类型、12种组织来源的330个数据集中的超过130万个高质量细胞。最终得到1,051个经过人工审编的TE亚家族。
- TE-SCALE提供了交互式框架从多水平探索人类癌症中的TE表达模式：泛癌-癌组织-个体-细胞。
- TE-SCALE支持多种下游分析功能，包括：肿瘤与正常组织间TE表达的比较、不同癌症或细胞类型间的差异TE表达分析、TE-基因共表达推断以及功能探索。

Case Study: lung cancer



- 在多种肿瘤中已观察到人内源性逆转录病毒（HERVs），特别是HERV-K的转录激活。
- 以肺癌为例，TE-SCALE通过UMAP浏览器支持对TE进行深入探索，可依据细胞类型、肿瘤部位及其他相关元数据展示细胞聚类情况（图A）。
- 在“差异TE表达”模块中，分析发现HERV-K的一个亚家族HERVK13-int在肿瘤细胞中相较于正常上皮细胞显著上调，并且在I期肺腺癌（LUAD）肿瘤中表达尤为升高（图B-D）。
- 在“TE-基因共表达网络”模块中，加权基因共表达网络分析（WGCNA）将HERVK13-int归入Tumor-M3模块，该模块富集了18.9%的癌症相关基因；“功能富集”分析进一步揭示其与TNF α -NF κ B信号通路、p53通路和凋亡等关键生物学过程密切相关（图E-F）。
- 这些结果表明，HERVK13-int可能作为早期诊断标志物，并参与潜在的肿瘤-免疫相互作用。
- 上述案例凸显了TE-SCALE在识别肿瘤特异性TE及其共表达网络方面的实用价值，有助于推动相关机制研究、生物标志物的发现以及肿瘤新抗原的筛选与优先排序。

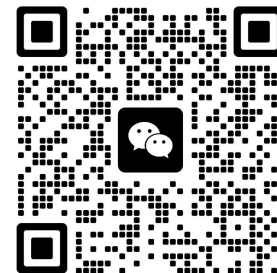
Thank you!

<http://jianglab.group>



中国科学院北京基因组研究所 (国家生物信息中心)
BEIJING INSTITUTE OF GENOMICS CHINESE ACADEMY OF SCIENCES / CHINA NATIONAL CENTER FOR BIOINFORMATION

李国超
北京 朝阳



Jiang Lab
single cell tech

[Home](#)

[News](#)

[Research](#)

[Publications](#)

[Lab Members](#)

[Team Life](#)

[About Jiang](#)

[Contact](#)

