

第三章：生物信息学编程基础

第一节 Linux操作系统

一、Linux简介

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。它能运行主要的UNIX工具软件、应用程序和网络协议。它支持32位和64位硬件。

Linux继承了Unix以网络为核心的设计思想，是一个**性能稳定的多用户网络操作系统**。

另外，linux**非常高效、稳定**：系统本身占有资源少，而且资源使用合理，很少出现死机现象。

- ◀ 生物信息学数据的分析处理，其特点是：
数据量大（一个文件几十个Gb），计算量大，计算时间长，使用的计算机配置比较高，价格贵。一般多人同时使用。或者采用集群模式，建立云计算中心来进行处理。
- ◀ 所以，生物信息学数据处理服务器一般都采用linux操作系统。

Linux操作系统诞生于1991 年10 月5 日（这是第一次正式向外公布时间）。

Linux可安装在各种计算机硬件设备中，比如手机、平板电脑、路由器、视频游戏控制台、台式计算机、大型机和超级计算机。

严格来讲，Linux这个词本身只表示Linux内核，但实际上人们已经习惯了用Linux来形容整个基于Linux内核，并且使用GNU工程各种工具和数据库的操作系统。

Linux存在着许多不同的Linux版本，但它们都使用了Linux内核。

发行版Linux：由个人，松散组织的团队，以及商业机构和志愿者组织编写。

发行版通常包括了其他的系统软件和应用软件，以及一个用来简化系统初始安装的安装工具，和让软件安装升级的集成管理器。

Redhat（小红帽），应该称为Redhat系列，包括RHEL（Redhat Enterprise Linux，也就是所谓的Redhat Advance Server，收费版本）、Fedora Core（由原来的Redhat桌面版本发展而来，免费版本）、CentOS（RHEL的社区克隆版本，免费）。

Redhat应该说是在国内使用人群最多的Linux版本。所以这个版本的特点就是使用人群数量大，资料非常多。

Redhat系列的包管理方式采用的是基于RPM包的YUM包管理方式，包分发方式是编译好的二进制文件。

稳定性方面RHEL和CentOS的稳定性非常好，适合于服务器使用，但是Fedora Core的稳定性较差，最好只用于桌面应用。

Debian，又称Debian系列，包括Debian和Ubuntu。Debian是社区类Linux的典范，是迄今为止最遵循GNU规范 的Linux系统。

Debian最具特色的是apt-get/dpkg包管理方式，其实Redhat的YUM也是在模仿Debian的APT方式，但在二进制文件发行方式中，APT应该是最好了。

Debian的资料也很丰富，有很多支持的社区，有问题求助方便。

Ubuntu是基于Debian的unstable版本加强而来，可以这么说，Ubuntu就是一个拥有Debian所有的优点，以及自己所加强的优点的近乎完美的Linux桌面系统。

根据选择的桌面系统不同，有三个版本可供选择，基于Gnome的Ubuntu，基于KDE的Kubuntu以及基于Xfce的Xubuntu。

特点是界面非常友好，容易上手，对硬件的支持非常全面，是最适合做桌面系统的Linux发行版本。

二、Linux常用命令行操作

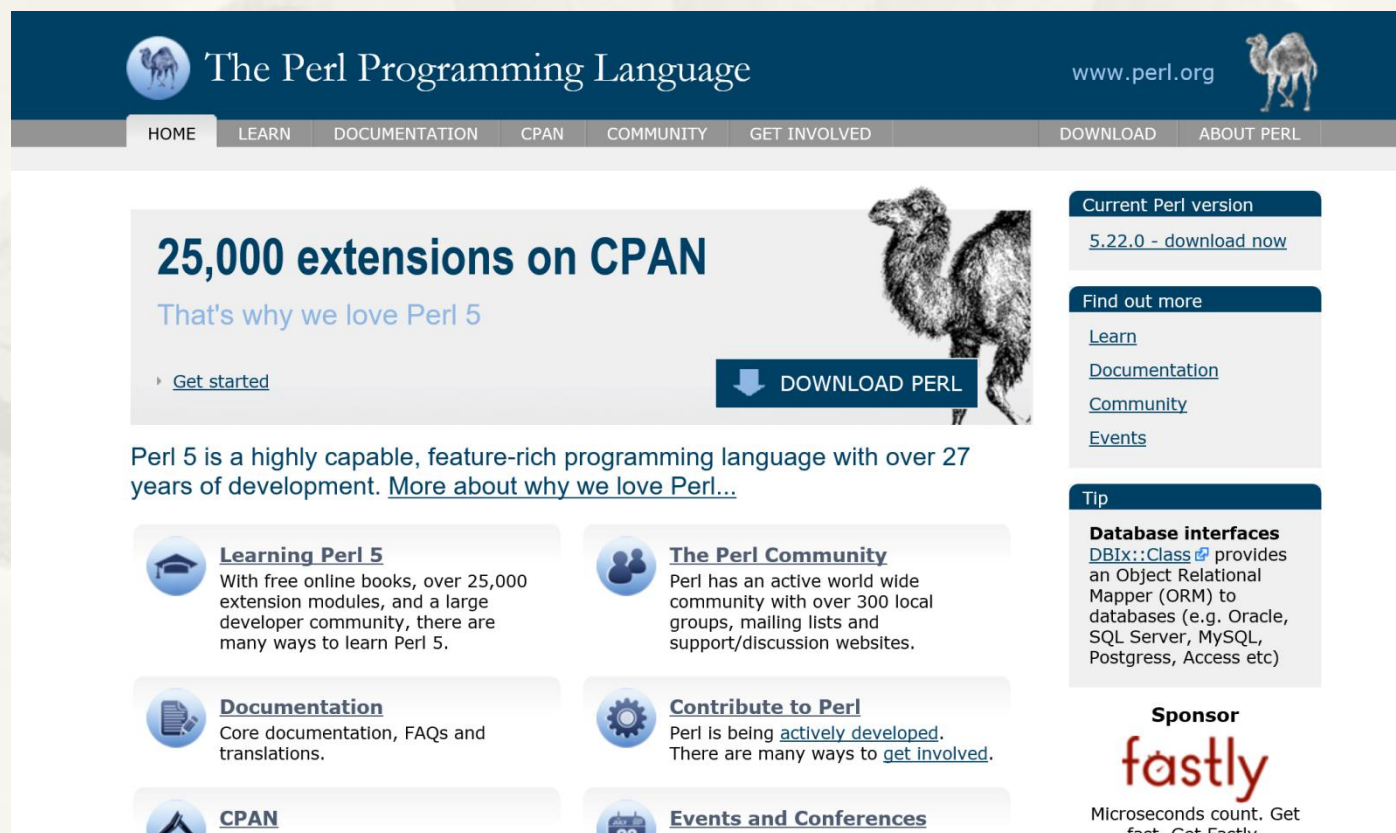
表 11-1 Linux 与 DOS 常用命令比较

Linux 命令符	DOS 命令符	功能
ls	dir	显示文件及目录
cd	cd	改变当前目录
mkdir	md	新建目录
rmdir	rd	删除目录
rm	del	删除文件
mv	move	移动文件
cp	copy	复制文件
less	type	阅读文件
echo	echo	输出字符串

第二节 生物信息学中的编程语言

一、Perl

1、Perl简介 (www.perl.org)



The Perl Programming Language

www.perl.org

HOME LEARN DOCUMENTATION CPAN COMMUNITY GET INVOLVED DOWNLOAD ABOUT PERL

25,000 extensions on CPAN

That's why we love Perl 5

[Get started](#)

[DOWNLOAD PERL](#)

Perl 5 is a highly capable, feature-rich programming language with over 27 years of development. [More about why we love Perl...](#)

Learning Perl 5

With free online books, over 25,000 extension modules, and a large developer community, there are many ways to learn Perl 5.

The Perl Community

Perl has an active world wide community with over 300 local groups, mailing lists and support/discussion websites.

Documentation

Core documentation, FAQs and translations.

Contribute to Perl

Perl is being [actively developed](#). There are many ways to [get involved](#).

Events and Conferences

[Perl Conference](#)

Current Perl version

[5.22.0 - download now](#)

Find out more

[Learn](#)
[Documentation](#)
[Community](#)
[Events](#)

Tip

Database interfaces
[DBIx::Class](#) provides an Object Relational Mapper (ORM) to databases (e.g. Oracle, SQL Server, MySQL, Postgress, Access etc)

Sponsor

fastly
Microseconds count. Get fast. Get Fastly.

2、Perl基础

数据类型：

字符串(\$string)、数组(@array)、哈希表(%hash)

正则表达式

m//;

子程序

Sub SUBNAME

{

code;

}

包

```
Package package1;
```

内置函数

abs、sqrt、exp

文件处理

```
open(FH, "path../filename");
```

模块

LWP::Simple

LWP::UserAgent

DBI

3、BioPerl概况

BioPerl是Perl的扩充，专门用于生物信息的工具与函数集。

表 11-3 BioPerl 模块说明

应用	模块
1. 序列操作	
对序列进行统计	SeqStats, SeqWord
确定限制性内切核酸酶位点	Bio::Restriction
识别氨基酸裂解位点	Sigcleave
多样性序列功能	OddCodes, SeqPattern
转换坐标系	Coordinate::Pair, RelSegment
2. 搜索相似序列	
运行远程 BLAST	using RemoteBlast, pm
分析 BLAST 和 FASTA 结果	Search, SearchIO
分析 BLAST 结果	BPlite, BPpsilite, and BPbl2seq
分析 HMM 结果	HMMER::Results, SearchIO
运行本地 BLAST	StandAloneBlast
3. 序列比对	SimpleAlign
4. 在基因组 DNA 寻找基因和其他结构	Genscan, Sim4, Grail, Genemark, ESTScan, MZEF, EPCR
5. 开发机器可读性的序列注释	
表征序列注释	SeqFeature, RichSeq, Location
表征序列注释	Annotation::Collection
表示大的序列	LargeSeq
表示改变的序列	LiveSeq

表示相关联的序列——突变体,多态性

Allele, SeqDiff

在序列注释中加入特征数据

SeqWithQuality

序列 XML 表征——生成和分析

SeqIO::game, SeqIO::bsml

使用 GFF 表示序列

Bio::DB::GFF

6. 操作序列簇

Cluster, ClusterIO

7. 在 BioPerl 中表示非序列数据:结构、树和图谱

使用 3D 结构对象并读取 PDB 文件

StructureI, Structure::IO

树对象和系统发生树

Tree::Tree, TreeIO, PAML

图谱对象操作遗传图谱

Map::MapI, MapIO

文献对象搜索文献数据库

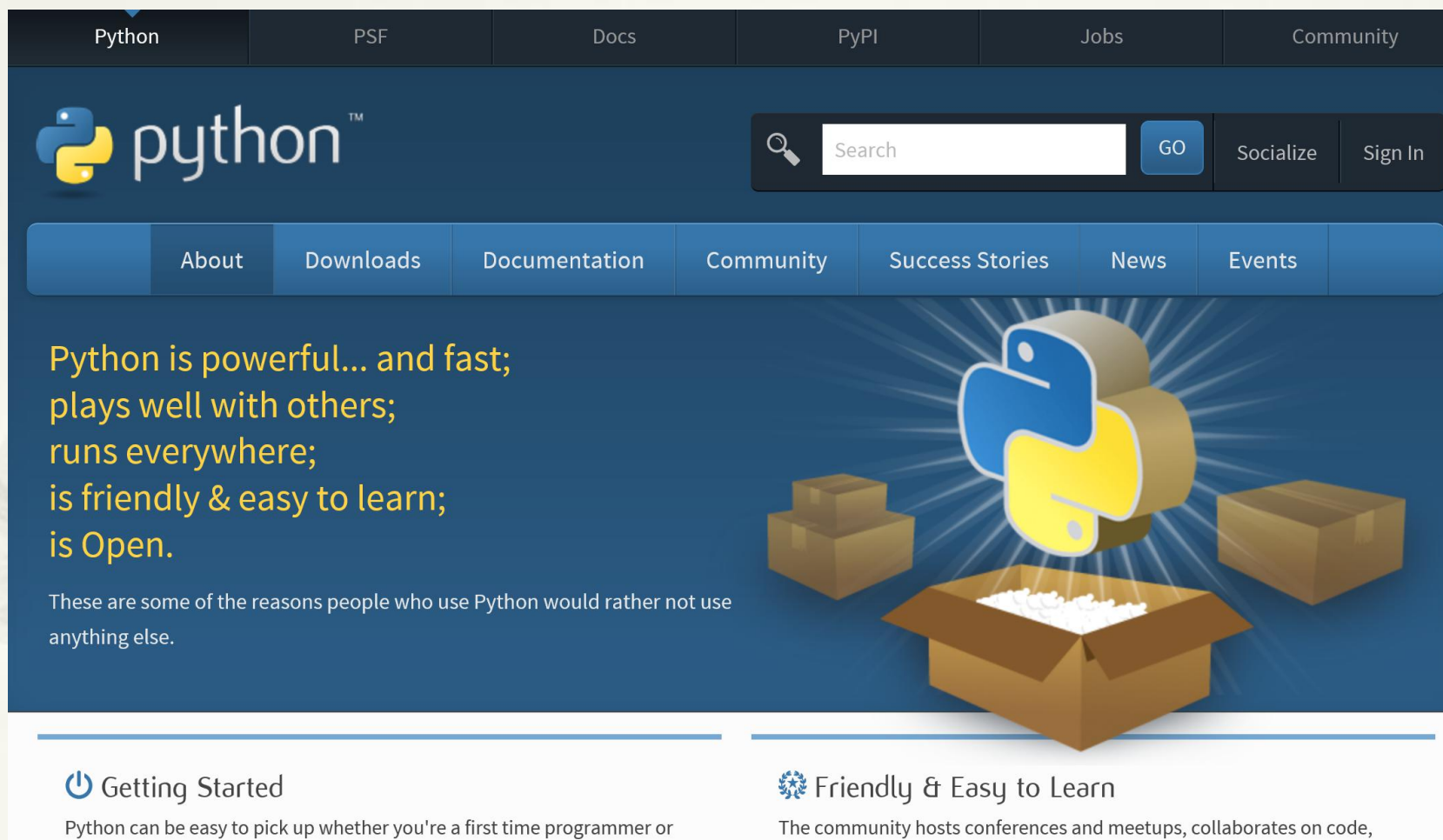
Biblio

图形对象以图形代表序列对象

Graphics

二、Python

1、Python简介 (www.python.org)



The image is a screenshot of the Python.org homepage. At the top, there is a dark blue navigation bar with links for Python, PSF, Docs, PyPI, Jobs, and Community. Below this is a large blue banner featuring the Python logo and the text "python™". To the right of the logo is a search bar with a magnifying glass icon, a "GO" button, and links for "Socialize" and "Sign In". Below the banner is a horizontal menu with links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area has a dark blue background with the text "Python is powerful... and fast; plays well with others; runs everywhere; is friendly & easy to learn; is Open." in yellow. Below this text is a 3D illustration of the Python logo emerging from an open cardboard box, with other closed boxes nearby. At the bottom, there are two white boxes: "Getting Started" with a power icon and the text "Python can be easy to pick up whether you're a first time programmer or", and "Friendly & Easy to Learn" with a gear icon and the text "The community hosts conferences and meetups, collaborates on code,".

Python

PSF

Docs

PyPI

Jobs

Community

python™

Search

GO

Socialize

Sign In

About

Downloads

Documentation

Community

Success Stories

News

Events

Python is powerful... and fast;
plays well with others;
runs everywhere;
is friendly & easy to learn;
is Open.

These are some of the reasons people who use Python would rather not use anything else.

Getting Started

Python can be easy to pick up whether you're a first time programmer or

Friendly & Easy to Learn

The community hosts conferences and meetups, collaborates on code,

Python在线学习（寒假作业）

← http://edu.51cto.com/course/course_id-527.html

51CTO学院 [首页](#) [课程](#) [直播培训](#) [路线图](#) [讲师](#) [更多](#) [登录](#) [注册](#)

[编程语言](#) > [Python](#) > Python公开课视频课程-基础篇



Python公开课视频
编程入门基础

共34课时

Python公开课视频课程-基础篇

12.12，优惠返场，名师真情回馈，课程5折起！满再送金币，多买多赠送！

价 格：**免费**

时 长：13小时17分钟

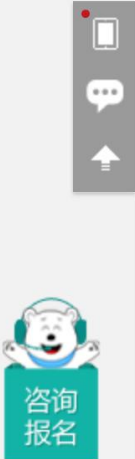
发布时间：2014-02-24

适用人群：大学生,运维工作人员

42 评价	4.9 综合评分	609 课程提问	1780 笔记
----------	-------------	-------------	------------

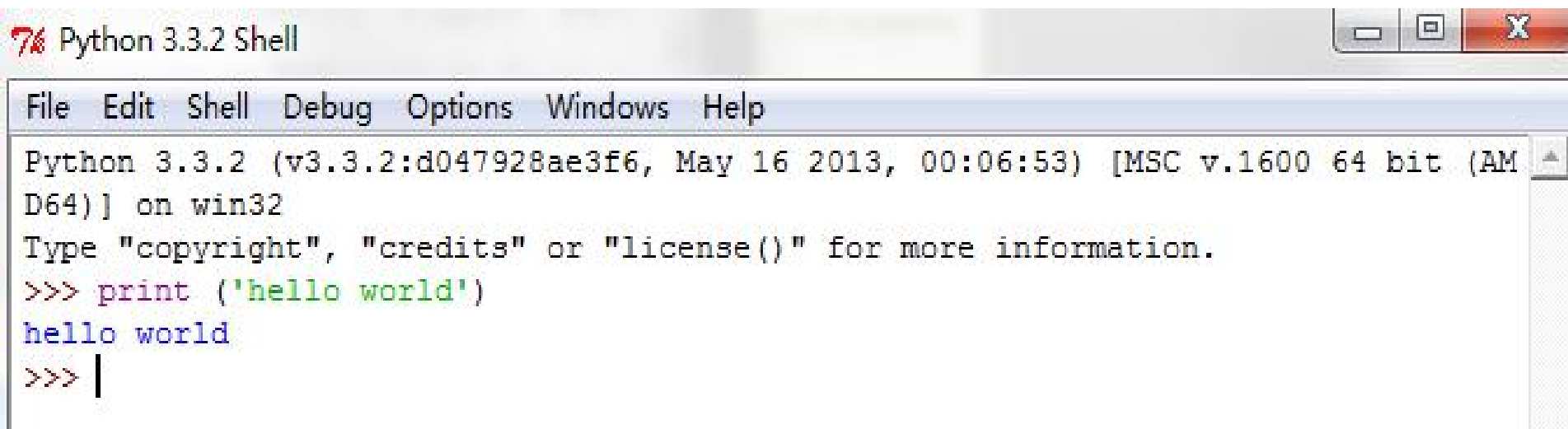
[我要学习](#)

[下载课程](#) ★ 收藏 (3082) [学习码观看](#) [分享到](#) [微信](#) [微博](#) [星星](#) [更多](#)



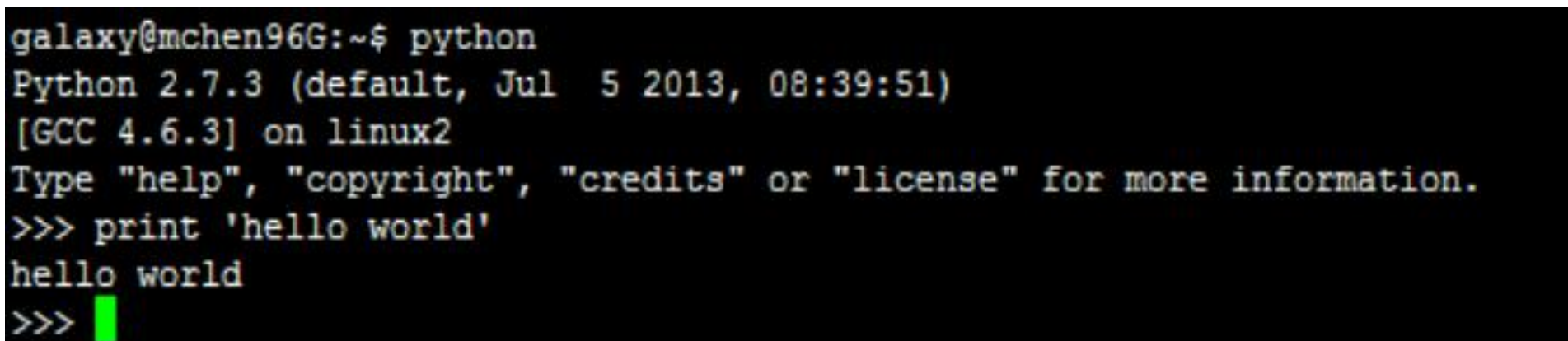
2、Python安装及开发运行环境的搭建

Windows:

A screenshot of a Windows application window titled "Python 3.3.2 Shell". The window has a standard Windows interface with minimize, maximize, and close buttons in the top right corner. Below the title bar is a menu bar with options: File, Edit, Shell, Debug, Options, Windows, and Help. The main text area displays the following content: "Python 3.3.2 (v3.3.2:d047928ae3f6, May 16 2013, 00:06:53) [MSC v.1600 64 bit (AMD64)] on win32", "Type 'copyright', 'credits' or 'license()' for more information.", and a Python prompt ">>>" followed by the command "print ('hello world')", the output "hello world", and another prompt ">>>" with a cursor. The text is color-coded: "print" is purple, "hello world" is green, and the output "hello world" is blue.

```
Python 3.3.2 Shell
File Edit Shell Debug Options Windows Help
Python 3.3.2 (v3.3.2:d047928ae3f6, May 16 2013, 00:06:53) [MSC v.1600 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print ('hello world')
hello world
>>> |
```

Linux:

A screenshot of a Linux terminal window. The prompt is "galaxy@mchen96G:~\$". The user has entered "python", and the terminal displays "Python 2.7.3 (default, Jul 5 2013, 08:39:51) [GCC 4.6.3] on linux2". Below this, it says "Type 'help', 'copyright', 'credits' or 'license' for more information." followed by a Python prompt ">>>". The user enters "print 'hello world'", and the terminal outputs "hello world". The prompt ">>>" is followed by a green cursor bar.

```
galaxy@mchen96G:~$ python
Python 2.7.3 (default, Jul 5 2013, 08:39:51)
[GCC 4.6.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> print 'hello world'
hello world
>>> █
```

Python集成开发环境简介

IDLE

Ipython

Eclipse+pydev

Ulipad

Pyscripter

Komodo和Komodo Edit

WingIDE

3、Python在生物信息学中的应用

(1) Python中生物信息学常用的模块

NumPy、SciPy、Rpy、Matplotlib

(2) 基于Python语言编写的生物信息常用程序

PyMOL、Modeller

三、Java

Java（计算机编程语言）



Java是一种可以撰写跨平台应用程序的面向对象的程序设计语言。**Java** 技术具有卓越的通用性、高效性、平台移植性和安全性，广泛应用于PC、数据中心、游戏控制台、科学超级计算机、移动电话和互联网，同时拥有全球最大的开发者专业社群。

中文名	扎哇、扎沃	现公司	Oracle
外文名	Java	推出时间	1995年5月
俗 语	咖啡	扩展名	.java
开发公司	Sun Microsystems	读 音	Java ['dʒɑ:və]



四、

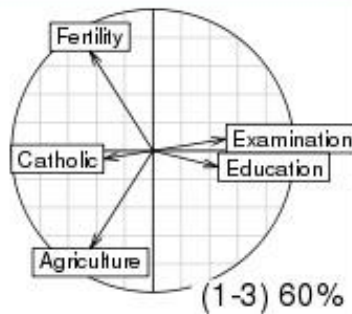


语言

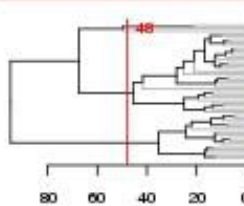
The R Project for Statistical Computing

PCA 5 vars

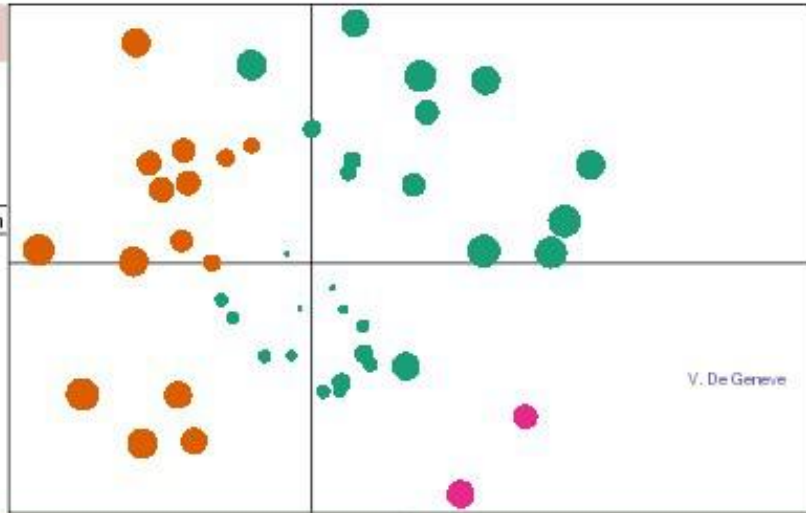
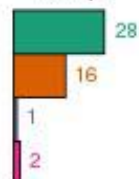
`prcomp(x = data, cor = cor)`



Clustering 4 groups

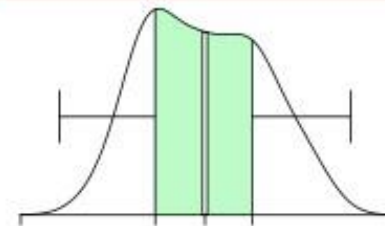
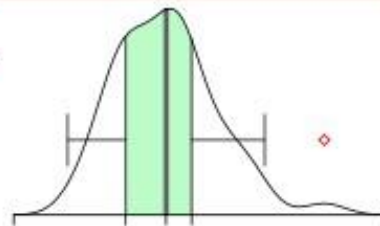


Groups



Factor 1 [41%]

Factor 3 [19%]



R语言中文网 (<http://www.r-china.net/portal.php>)



用QQ帐号登录

只需一步，快速开始

用户名

密码

☐ 自动登录

登录

首页

论坛

微博

最新动态

写技术博客

积分规则

R语言安装下载

积分充值

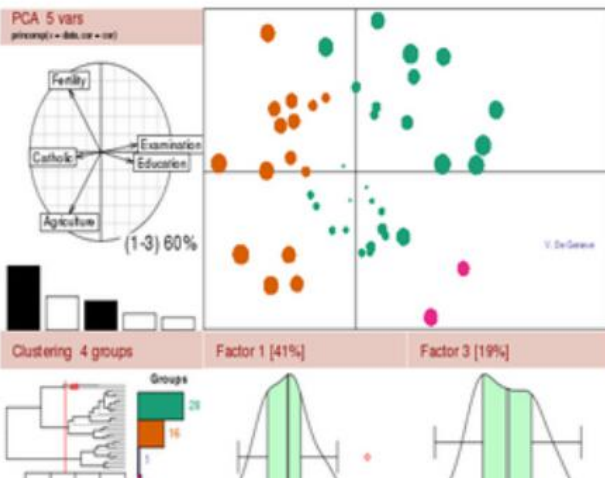
快捷导

请输入搜索内容

帖子



热搜: 自相关 apriori 验证会员 因子分析 xlsx ggplot 程序包 excel输入 splines arules



会员



走着回家



lsypku



信天翁



sunchao



472163547



还是诗雨



htl55034...



faya



官方QQ群

QQ群号码: 74076289;



R语言入门答疑VIP群号码: 313083685;



R语言新浪微博: R语言中文网官网

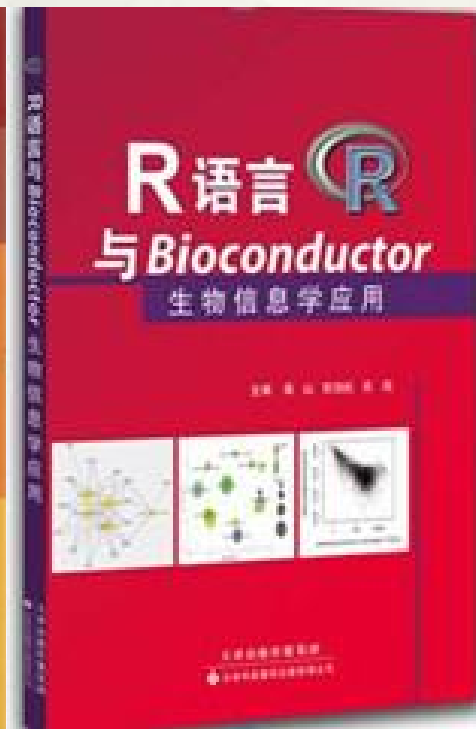
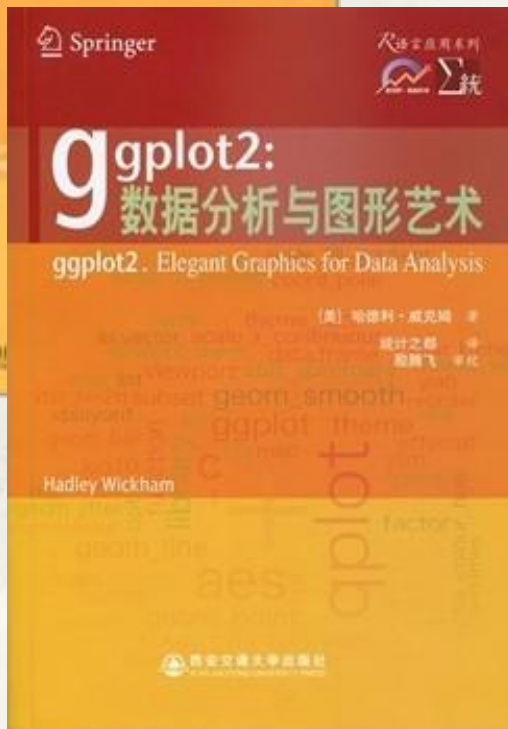
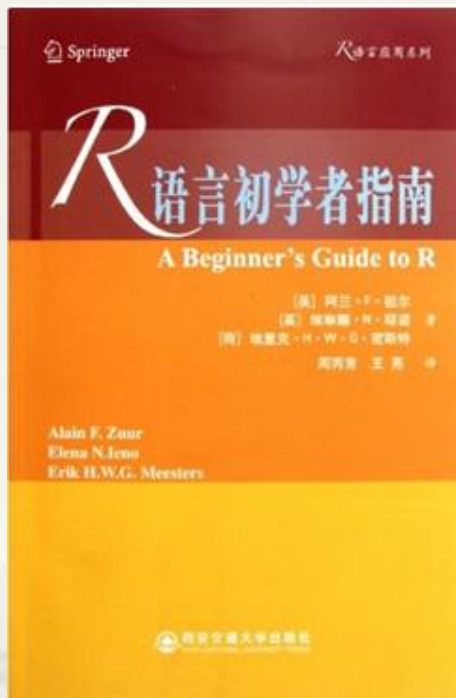
R语言微信公众平台: R语言中文网

注: 微信公众号添加方法: 微信->添加朋友->手机号->输入“R语言中文网”, 添加就可以了。

注册须知

本站为R语言专业论坛, 注册采取严格的审核制

推荐自学参考书:



<http://www.highstat.com/>

Why R?



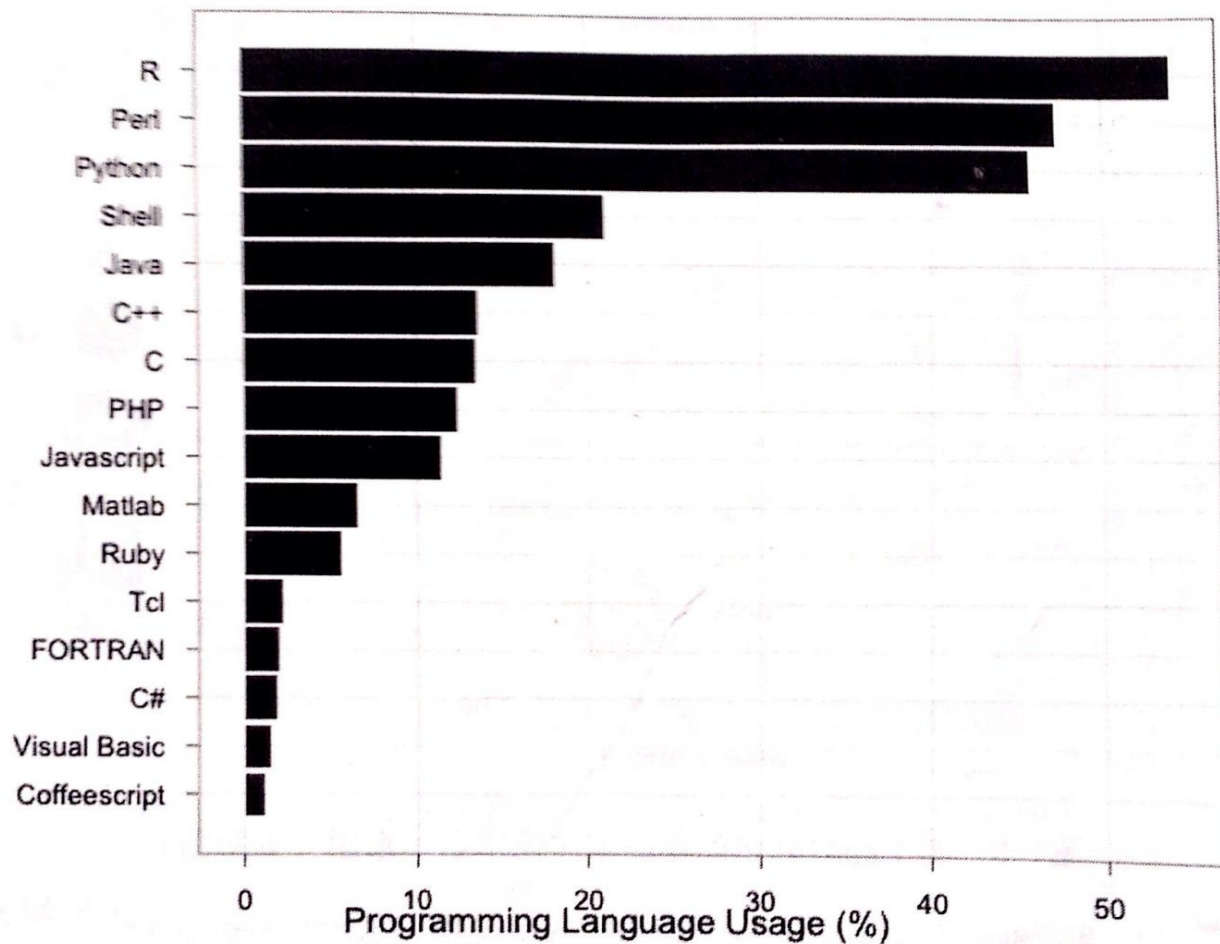
R first appeared in 1996, when the statistics professors Robert Gentleman, left, and Ross Ihaka released the code as a free software package.

R是：

- ← 一个开放的**统计编程环境**；
- ← 一种**语言**，是S语言的一种方言之一（另一种是S-plus）；
- ← 一种软件，是统计分析与图形直观显示于一体的**统计分析软件**。

- 我们希望软件能便宜一些，而R的价格是 \$ 0
- R作为解释性语言，在速度上自然会逊于C Fortran等“底层”语言，但是R可以与其它语言相互调用
- 对于统计编程来说，R具有其它软件无法比拟的便捷性（统计的数据结构、统计的函数、统计的图形）

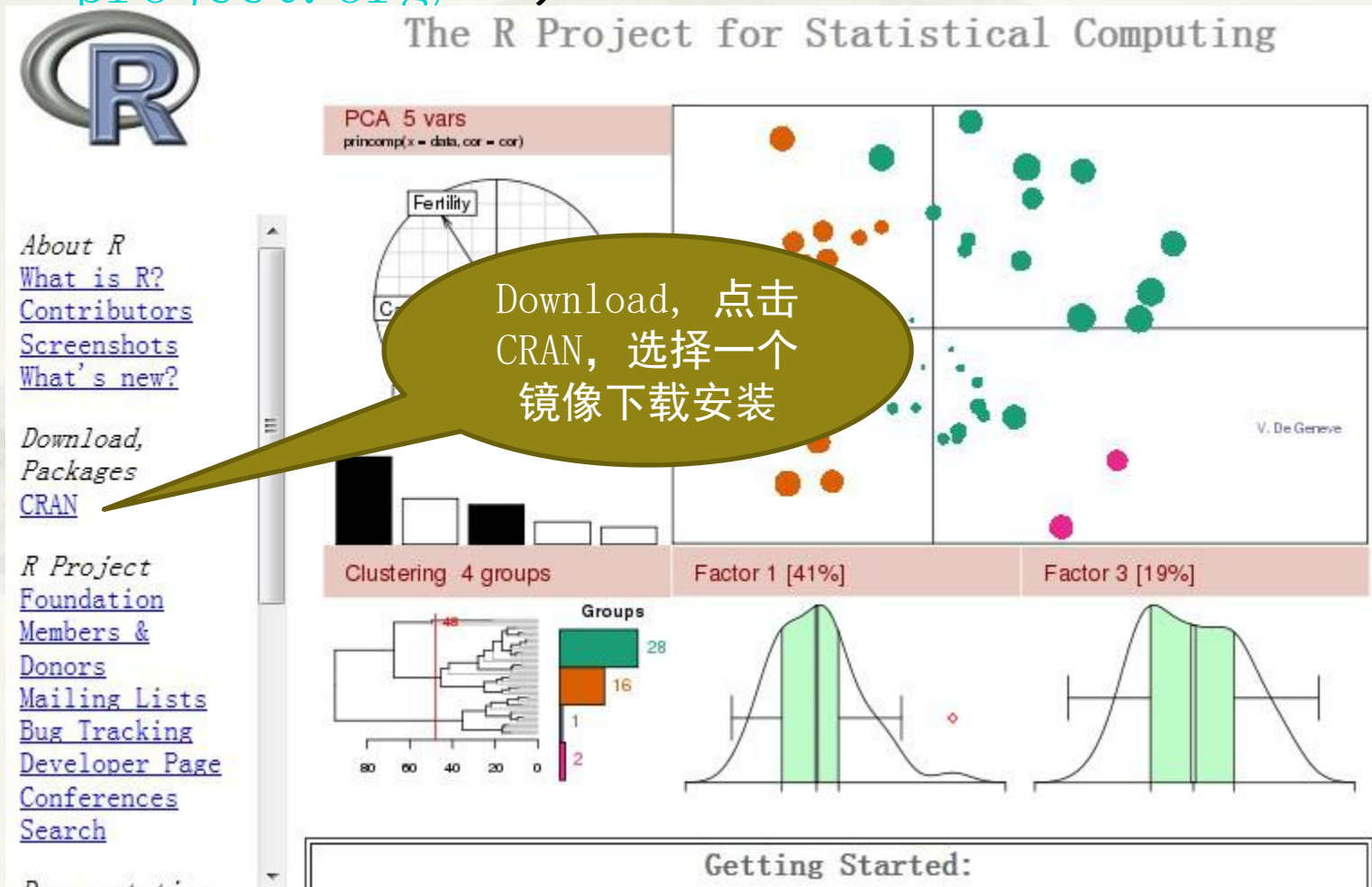
R缺点是：需要编程，不傻瓜！！！！



主流生物信息学编程语言的用户量排名
(图片来源: 高山等, R语言与Bioconductor生物信息学应用)

(一)、R语言的安装

← 主程序的下载与安装 (<http://www.r-project.org/>)





About R
[What is R?](#)
[Contributors](#)
[Screenshots](#)
[What's new?](#)

[Download, Packages CRAN](#)

[R Project Foundation](#)
[Members & Donors](#)
[Mailing Lists](#)
[Bug Tracking](#)
[Developer Page](#)
[Conferences](#)

Canada

<http://www.vps.fmvz.usp.br/CRAN/> University of Sao Paulo, Sao Paulo
<http://brieger.esalq.usp.br/CRAN/> University of Sao Paulo, Piracicaba

<http://cran.stat.sfu.ca/> Simon Fraser University, Burnaby
<http://mirror.its.dal.ca/cran/> Dalhousie University, Halifax
<http://cran.utstat.utoronto.ca/> University of Toronto
<http://cran.skazkaforyou.com/> iWeb, Montreal
<http://cran.parentingamerica.com/> iWeb, Montreal

Chile

<http://dirichlet.mat.puc.cl/> Pontificia Universidad Catolica de Chile, Santiago

China

<http://ftp.ctex.org/mirrors/CRAN/> CTEX.ORG
<http://mirror.bjtu.edu.cn/cran> Beijing Jiaotong University, Beijing

<http://mirrors.ustc.edu>

<http://mirrors.xmu.edu>

Colombia

<http://www.laqee.unal>

<http://www.icesi.edu.co>

Czech Republic

[CRAN](#)
[Mirrors](#)
[What's new?](#)
[Task Views](#)
[Search](#)

[About R](#)
[R Homepage](#)
[The R Journal](#)

[Software](#)
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Other](#)

[Documentation](#)
[Manuals](#)
[FAQs](#)
[Contributed](#)



The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

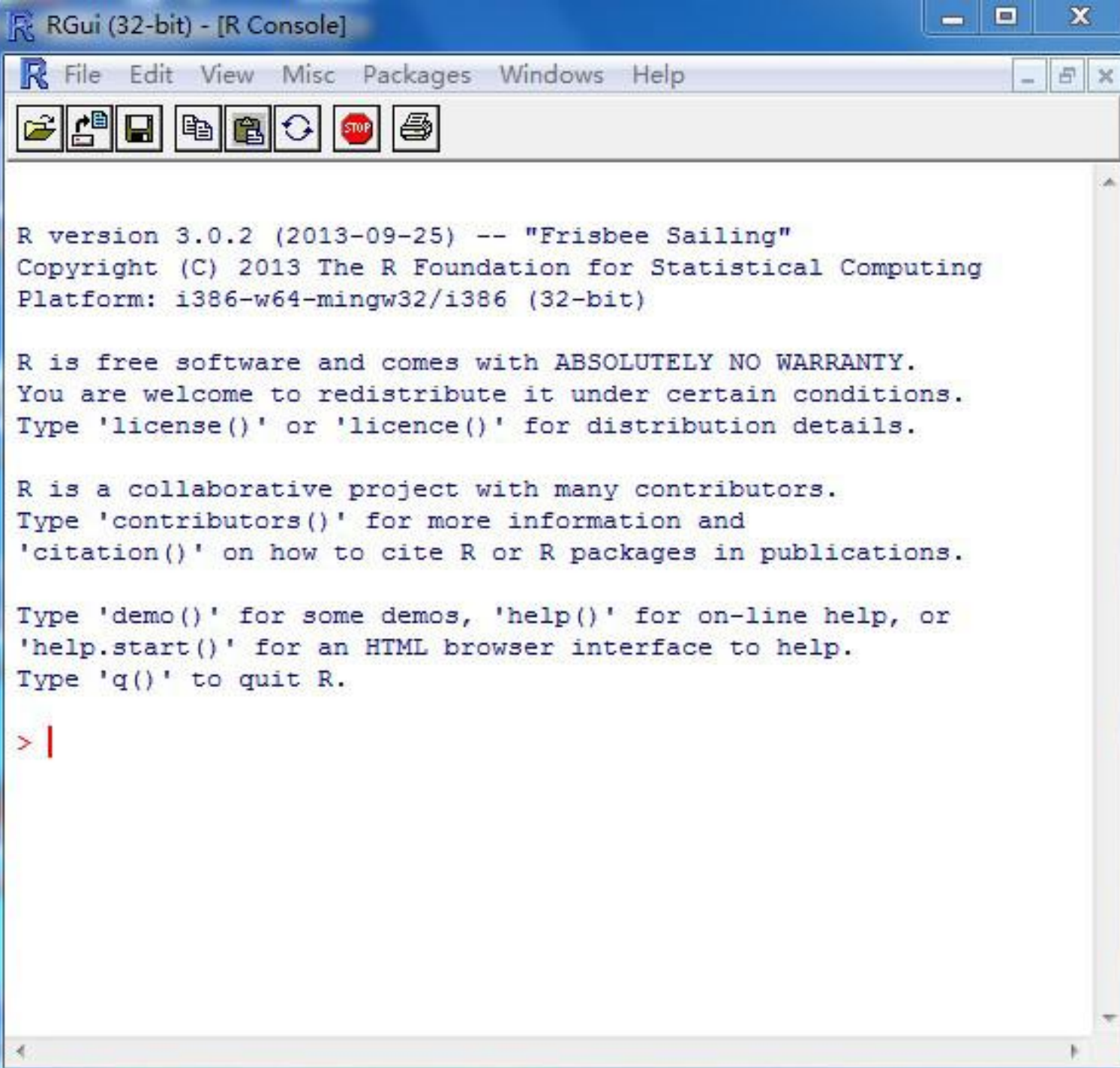
- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2014-07-10, Sock it to Me) [R-3.1.1.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are



扩展包的下载与安装

- ← R语言一个突出特点就是大量的扩展包存在，能够实现更为复杂的统计绘图、工程计算以及与其它语言的相互调用等功能。
- ← 扩展包主要有两个来源，一部分来自R的官方网站CRAN (The Comprehensive R Archive Network, <http://cran.r-project.org/>) ;另一部分是第三方 (CRAN) 管理的，如生物数据分析包Bioconductor (<http://www.bioconductor.org>) 。

← 在线安装（以下均以Windows平台为例）：在菜单栏选择『Packages』→『Install package(s)…』，出现“CRAN mirror”界面，选择中国境内的CRAN，在“Packages”选择要安装的程序包，点击『确定』，下载完成后自动安装，完成后进入Rgui窗口。

更一般的是使用命令安装：

```
options(CRAN= “http://cran.r-project.org” ); #指定镜像网站
```

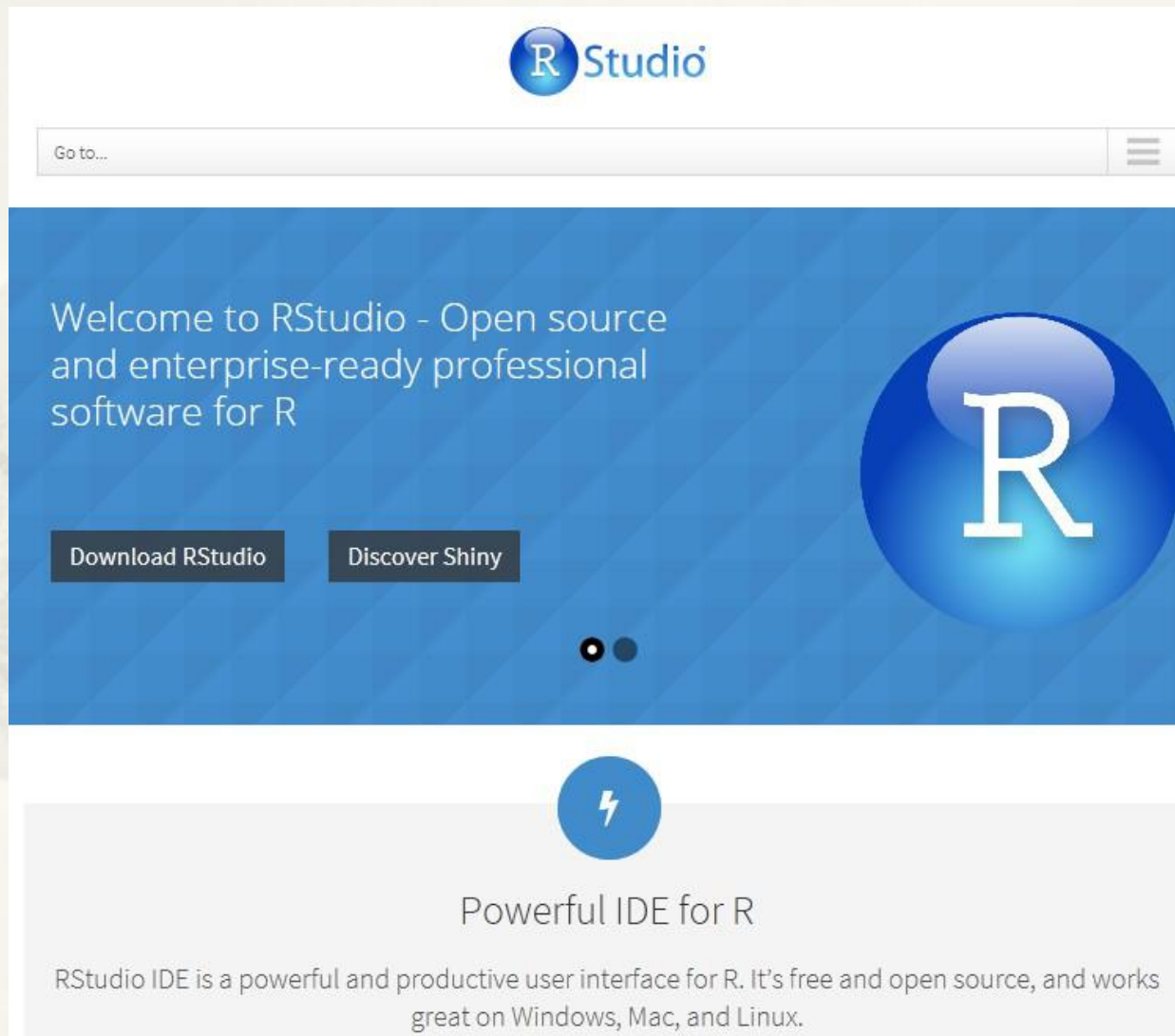
```
install.packages( “ggplot2” ); #安装很强大的一个作图工具包
```

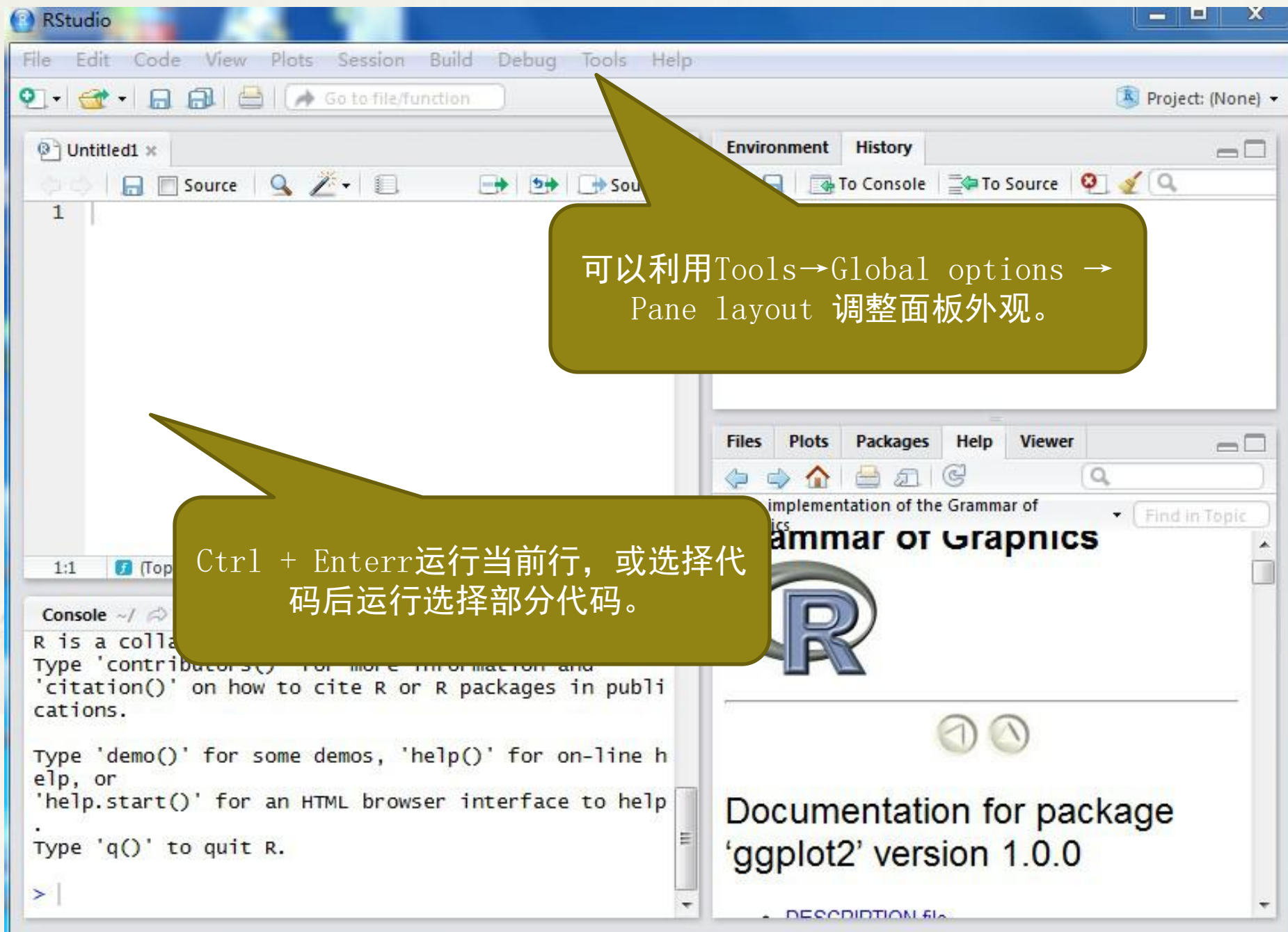
- ← 下载到本地安装：在菜单栏选择『Packages』→『Install package(s) from local zip files…』，找到下载的包，点击确定。
- ← 使用第三方提供的脚本在线安装（以Bioconductor为例）：

```
source( "http://bioconductor.org/biocLite.R" )  
#下载安装脚本biocLite到环境中  
biocLite() #利用biocLite函数安装所有  
Bioconductor核心包，其它Bioconductor程序包依赖于这  
些核心包，所以必须先执行该命令，待安装完成后再安装  
其它包，否则某些包将无法使用library命令加载。安装  
所需的特定包时，需传入包的名称，如  
biocLite( "limma" )。
```

R语言的集成开发环境IDE: RStudio

← <http://www.rstudio.com/>





交互式使用R

- 一个R程序需要你输入命令时，默认的提示符是 `>`
- 退出R程序的命令是`>q()`
- 此时R会话会问你是否需要保存数据：

```
> q()  
Save workspace image to ~/.RData? [y/n/c]: |
```


Help

- R也有类似man的内嵌帮助工具
- 比如得到solve函数的帮助:
- `>help(solve)`

此时会发出请求, 得到html格式的帮助
或者? solve 作用一样的

(二)、R语言的对象

- ← 向量 (vector) : 基本类型, 可以存放数值、逻辑和字符等类型数据;
- ← 数组 (array) : 多维向量 (数值类型必须一致) ;
- ← 矩阵 (matrix)
- ← 数据框 (data frame)
- ← 列表 (list)

```
1  c(1,2,3,4)
2  c('hello woild','I am a R user')
3
4  c(1,2,3,4)+c(3,4,5,6)
5  c(1,2,3,4)+c(1,2)
6
7  1:6
8  6:1
9
10 c(1,2,3,4) > c(1,2,1,2)
11 |
```

← 向量之间可以进行加减乘除在内的各种运算；

```
> x <- c(1,2,3,4)
> x
[1] 1 2 3 4
> y <- c('hello woild','I am a R user')
> y
[1] "hello woild"    "I am a R user"
> |
```

- ← 变量 (variable) : 变量对应计算机的内存或寄存器地址, 用于保存数据; R语言的变量常常是一个向量。
- ← “<-” 赋值运算符 (也可以使用 “=”) , 在Rstudio中快捷键是Alt + “-” !

← 如果从变量 x 中取出某一元素，可以用方括号加索引号，如：

```
> x <- c(1,2,3,4)
> x
[1] 1 2 3 4
> x[1]
[1] 1
> x[1:3]
[1] 1 2 3
> x[-4]
[1] 1 2 3
> |
```

函数

R是一种解释性语言，不需要先编译成**.exe**文件，输入后可直接运行。

函数形式

function (对象, 选项=)

每一个函数执行特定的功能，后面紧跟括号，例如：

平均值 **mean()**

求和 **sum()**

绘图 **plot()**

排序 **sort()**

← 在R语言中，所有的操作和运算都是由函数来完成的。

```
> exp(1)
[1] 2.718282
> exp(c(1,2,3,4))
[1] 2.718282 7.389056 20.085537 54.598150
> |
```

- 在Rstudio中可以使用函数提示，如输入“ex”后按“Tab”键。

← 数组（array）：多维向量（数值类型必须一致）；

```
> x <- 1:12
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12
> a <- array(data=x,dim=c(3,4))
> a
      [,1] [,2] [,3] [,4]
[1,]    1    4    7   10
[2,]    2    5    8   11
[3,]    3    6    9   12
> a[2,2]
[1] 5
> |
```

- ← 数据框（data frame）是一种类似Excel表格的数据结构。数据框非常适合用来进行分析，它的每一列可以代表数据的每个变量或属性，每一行可以代表一个样

```
> city <- c('beijing','shanghai','shanghai','chongqing')
> age <- c(30,23,45,22)
> sex <- c('F','M','M','F')
> people <- data.frame(city,age,sex)
> print(people)
```

	city	age	sex
1	beijing	30	F
2	shanghai	23	M
3	shanghai	45	M
4	chongqing	22	F

```
> |
```

数据可以是不同类项，
但长度必须一致！

```
> city <- c('beijing','shanghai','shanghai','chongqing')
> age <- c(30,23,45,22)
> sex <- c('F','M','M','F')
> people <- data.frame(city,age,sex)
> print(people)
      city age sex
1  beijing  30  F
2  shanghai 23  M
3  shanghai 45  M
4 chongqing 22  F
> people
      city age sex
1  beijing  30  F
2  shanghai 23  M
3  shanghai 45  M
4 chongqing 22  F
> people[2,3]
[1] M
Levels: F M
> people$age
[1] 30 23 45 22
> people$age>30
[1] FALSE FALSE  TRUE FALSE
```

```
> people[people$age>30,]
      city age sex
3  shanghai 45  M
> |
```

- ← 列表（list），是最为灵活的数据结构，它的不同元素可以是不同类型，也可以是不同长度。

```
> city <- c('beijing','shanghai','shanghai','chongqing')
> age <- c(30,23,45,22)
> sex <- c('F','M','M','F')
```

```
> peoplelist <- list(age=age,city=city,sex=sex)
> print(peoplelist)
$age
[1] 30 23 45 22

$city
[1] "beijing"    "shanghai"   "shanghai"   "chongqing"

$sex
[1] "F" "M" "M" "F"

> |
```

```
> peoplelist$age
[1] 30 23 45 22
> peoplelist[1]
$age
[1] 30 23 45 22

> peoplelist[[1]]
[1] 30 23 45 22
> |
```


对象和类

- ← R是一种基于对象的语言，所以你在R语言中接触到的每一样东西都是一个对象，对象中包含我们需要的数据，同时对象也具有很多 属性（Attribute）。其中一种重要的属性就是它的类（Class），R语言中最为基本的类包括数值、逻辑、字符，在此基础上构成了一些复合型的类，包括矩阵、数组、数据框和列表。

- ← 对于任何一个对象，我们可以用`class()`函数来观察它的类，同时可以用`attributes()`函数来观察其属性，用`str()`函数了解一个对象的

```
> class(people)
[1] "data.frame"
> attributes(people)
$names
[1] "city" "age"  "sex"

$row.names
[1] 1 2 3 4

$class
[1] "data.frame"

> str(people)
'data.frame': 4 obs. of 3 variables:
 $ city: Factor w/ 3 levels "beijing","chongqing",...
: 1 3 3 2
 $ age : num 30 23 45 22
 $ sex : Factor w/ 2 levels "F","M": 1 2 2 1
> |
```

(三)、R语言中的可视化函数

← graphics 基本软件包，图形朴素，参数调整复杂（也可画出比较漂亮的图形）

← lattice

← ggplot2 (<http://ggplot2.org>)



The screenshot shows the ggplot2 website. At the top, the title 'ggplot2' is displayed in a large, bold font. Below it, a paragraph describes ggplot2 as a plotting system for R based on the grammar of graphics. A section titled 'Documentation' provides a link to 'docs.ggplot2.org'. Another section titled 'Mailing list' invites users to ask questions on R-help or sign up for a mailing list. At the bottom, there is a form for email subscription with a 'Subscribe' button.

ggplot2

ggplot2 is a plotting system for R, based on the grammar of graphics, which tries to take the good parts of base and lattice graphics and none of the bad parts. It takes care of many of the fiddly details that make plotting a hassle (like drawing legends) as well as providing a powerful model of graphics that makes it easy to produce complex multi-layered graphics.

Documentation

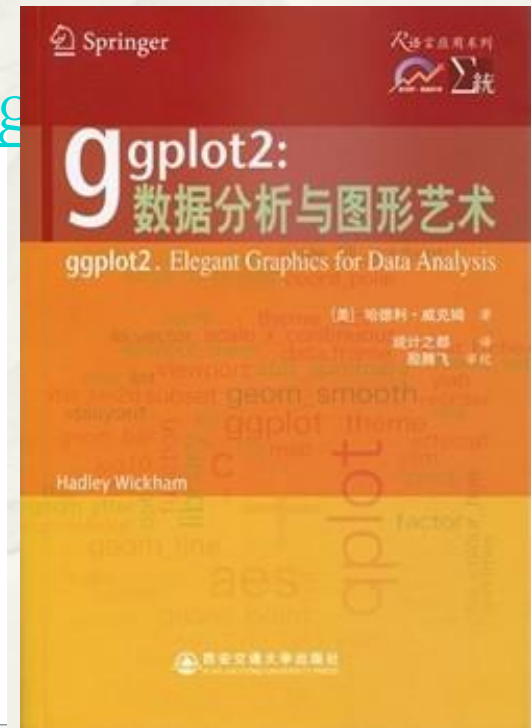
ggplot2 documentation is now available at docs.ggplot2.org.

Mailing list

You are welcome to ask ggplot2 questions on R-help, but if you'd like to participate in a more focussed mailing list, please sign up for the [ggplot2 mailing list](#):

Your email address:

You must be a member to post messages, but anyone can read the archived discussions.



Why ggplot2 ?

- 用户能在更抽象层面上控制图形，使创造性绘图更容易；
- 采用图层的设计方式，有利于结构化思维；
- 图形美观，同时避免繁琐细节。

- ← Hadley Wickham读博士期间作品(2005)，现在Rice University当教授；
- ← Hadley Wickham说ggplot2是一个强大的作图工具，它可以让你不受现有图形类型的限制，创造出任何有助于解决你所遇到🤔🤔🤔图形。（**一点也不谦虚。**）
- ← H. W. 还说了另外一句话，“学习ggplot2你得忘记一些东西”，所以也有人说ggplot2是作图软件中的太极功。



```
← install.packages("ggplot2") #安装  
ggplot2包
```

```
← qplot() 函数
```

Hadley Wickham很善解人意，知道我们接受一种新事物不会太容易，所以设计了个qplot函数。qplot 即“快速作图”

(quick plot)，顾名思义，能快速对数据进行可视化分析。它的用法和R base包的plot函数很相似，主要作用是让读者/用户在不知不觉中洗脑。


```
← qplot(x, y = NULL, ..., data,  
  facets = NULL, margins = FALSE,  
  geom = "auto", stat = list(NULL),  
  position = list(NULL), xlim = c(NA,  
  NA), ylim = c(NA, NA), log = "",  
  main = NULL, xlab =  
  deparse(substitute(x)), ylab  
  =deparse(substitute(y)), asp = NA)
```

- ← `x, y`: 意义明确, 不用说了
- ← `data`: 这个可以有, 为数据框 (`data.frame`) 类型; 如果有这个参数, 那么 `x, y` 的名称必需对应数据框中某列变量的名称。
- ← `facets`: 图形/数据的分面。这是 `ggplot2` 作图比较特殊的一个概念, 它把数据按某种规则进行分类, 每一类数据做一个图形, 所以最终效果就是一页多图。
- ← `margins`: 是否显示边界。
- ← `geom`: 图形的几何类型 (`geometry`), 这又是 `ggplot2` 的作图概念。 `ggplot2` 用几何类型表示图形类别, 比如 `point` 表示散点图、`line` 表示曲线图、`bar` 表示柱形图等。
- ← `stat`: 统计类型 (`statistics`), 这个更加特殊。直接将数据统计和图形结合, 这是 `ggplot2` 强大和受欢迎的原因之一。
- ← `position`: 图形或者数据的位置调整, 这不算太特殊, 但对于图形外观很重要。
- ← `xlim, ylim, xlab, ylab, asp`: 初步可以按照 `plot` 函数的相应参数来理解。

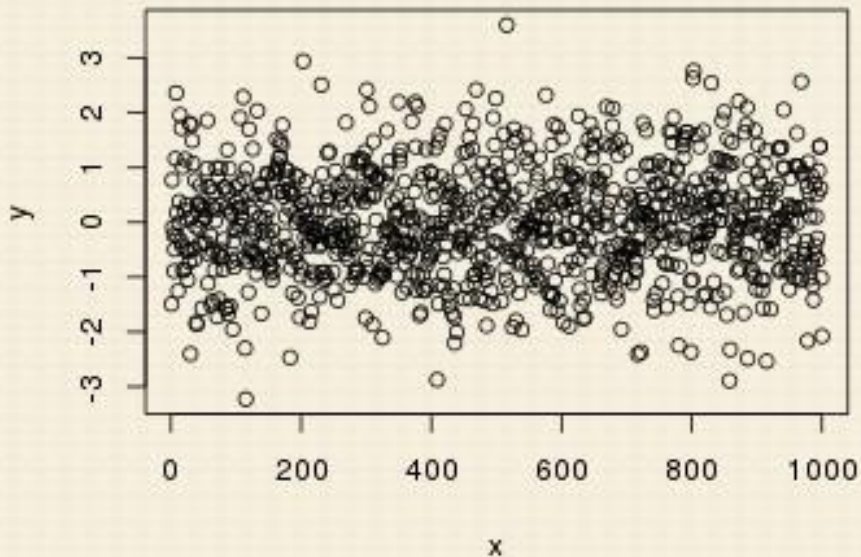
qplot做散点图

```
> library(ggplot2);  
> x <- 1:1000;  
> y <- rnorm(1000);  
> plot(x, y, main="Scatter plot by plot()");  
> qplot(x,y, main="Scatter plot by qplot()");
```

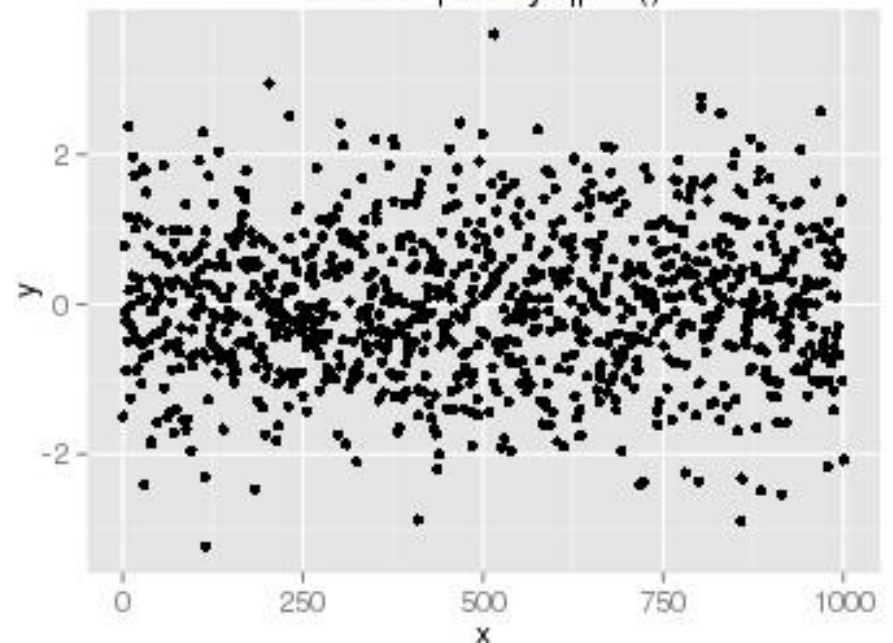
- ← **使用向量数据**和plot函数一样，如果不指定图形的类型，qplot默认做出散点图。对于给定的x和y向量做散点图，qplot用法也和plot函数差不多。

```
> library(ggplot2);  
> x <- 1:1000;  
> y <- rnorm(1000);  
> plot(x, y, main="Scatter plot by plot()");  
> qplot(x,y, main="Scatter plot by qplot()");
```

Scatter plot by plot()



Scatter plot by qplot()



- ← **使用数据框数据：**数据框可以用来存储数值、字符串、因子等不同类型等数据；把数据放在同一个R数据框对象中可以避免使用过程中数据关系的混乱；数据外观的整理和转换方便。
- ← ggplot2中使用数据框作图的最直接的一个效果就是：你可以直接用数据的分类特性（数据框中的列变量）来决定图形元素的外观，这个过程在ggplot2中称为映射（mapping），是自动的。
- ← 在演示使用数据框作图的好处之前我们先了解以下ggplot2提供的一组有关钻石的示范数据 diamonds（测试数据集是ggplot2中的data(diamonds)，具体可以用?diamonds查看）。

```
> ?diamonds
> str(diamonds)
'data.frame': 53940 obs. of 10 variables:
 $ carat : num 0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 0.23 ...
 $ cut : Ord.factor w/ 5 levels "Fair"<"Good"<...: 5 4 2 4 2 3 3 3 1 3 ...
 $ color : Ord.factor w/ 7 levels "D"<"E"<"F"<"G"<...: 2 2 2 6 7 7 6 5 2 5 ...
 $ clarity: Ord.factor w/ 8 levels "I1"<"SI2"<"SI1"<...: 2 3 5 4 2 6 7 3 4 5 ...
 $ depth : num 61.5 59.8 56.9 62.4 63.3 62.8 62.3 61.9 65.1 59.4 ...
 $ table : num 55 61 65 58 58 57 57 55 61 61 ...
 $ price : int 326 326 327 334 335 336 336 337 337 338 ...
 $ x : num 3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ...
 $ y : num 3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 ...
 $ z : num 2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39 ...
> |
```

- ← 可以看到这是数据框（data.frame）类型，有10个变量（列），每个变量有53940个测量值（行）。第一列为钻石的克拉数（carat），为数字型数据；第二列为钻石的切工好坏（cut），为因子类型数据，有5个水平；第三列为钻石颜色（color），为7水平的因子；后面还有其他数据。由于数据太多，我们只取前7列的100个随机观测值。


```
> set.seed(1000)#设置随机种子，使随机取样具有可重复性
>
> datax <- diamonds[sample(53940,100),seq(1,7)]
> head(datax,4)
```

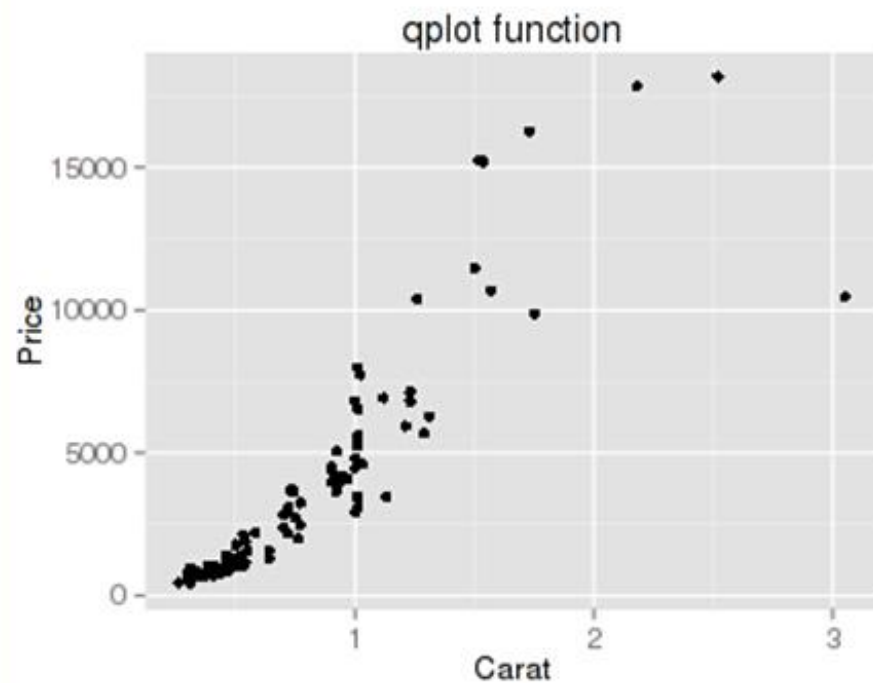
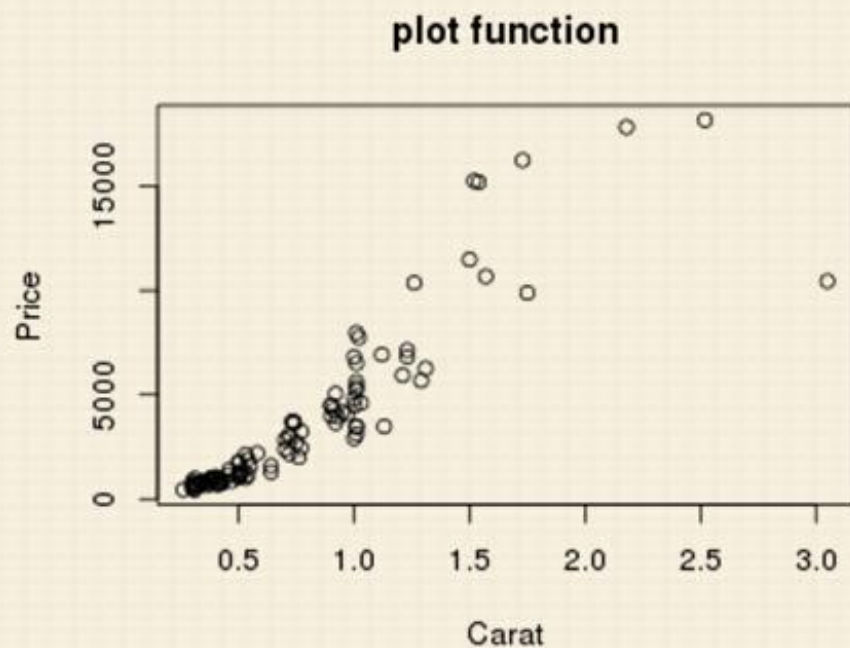
	carat	cut	color	clarity	depth	table	price
17686	1.23	Ideal	H	VS2	62.2	55	7130
40932	0.30	Ideal	E	SI1	61.7	58	499
6146	0.90	Good	H	VS2	61.9	58	3989
37258	0.31	Ideal	G	VVS1	62.8	57	977

```
> |
```

如果要做钻石克拉和价格关系的曲线图，用plot和qplot函数都差不多：

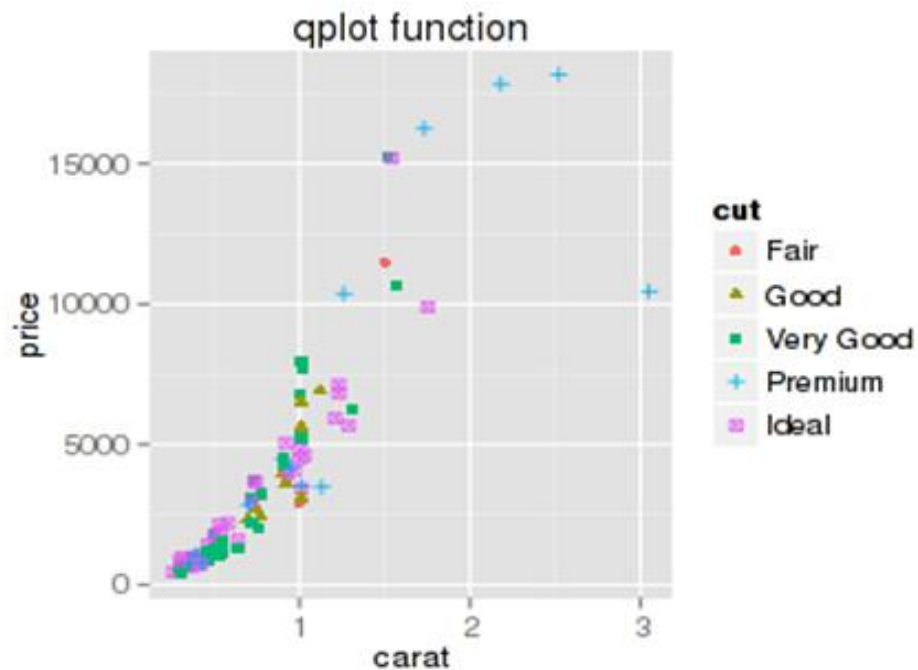
```
> plot(x=datax$carat, y=datax$price, xlab="Carat", ylab="Price", main="plot function");
```

```
> qplot(x=carat, y=price, data=datax, xlab="Carat", ylab="Price", main="qplot function");
```



← 但如果要按切工进行分类作图，`plot`函数的处理就复杂了，你首先得将数据进行分类提取，然后再一个个作图。虽然可以用循环完成，但作图后图标的添加还得非常小心，你得自己保证数据和图形外观之间的对应关系。但用`ggplot2`作图你需要考虑数据分类和图形元素方面的问题就很少，你只要告诉它用做分类的数据就可以了：

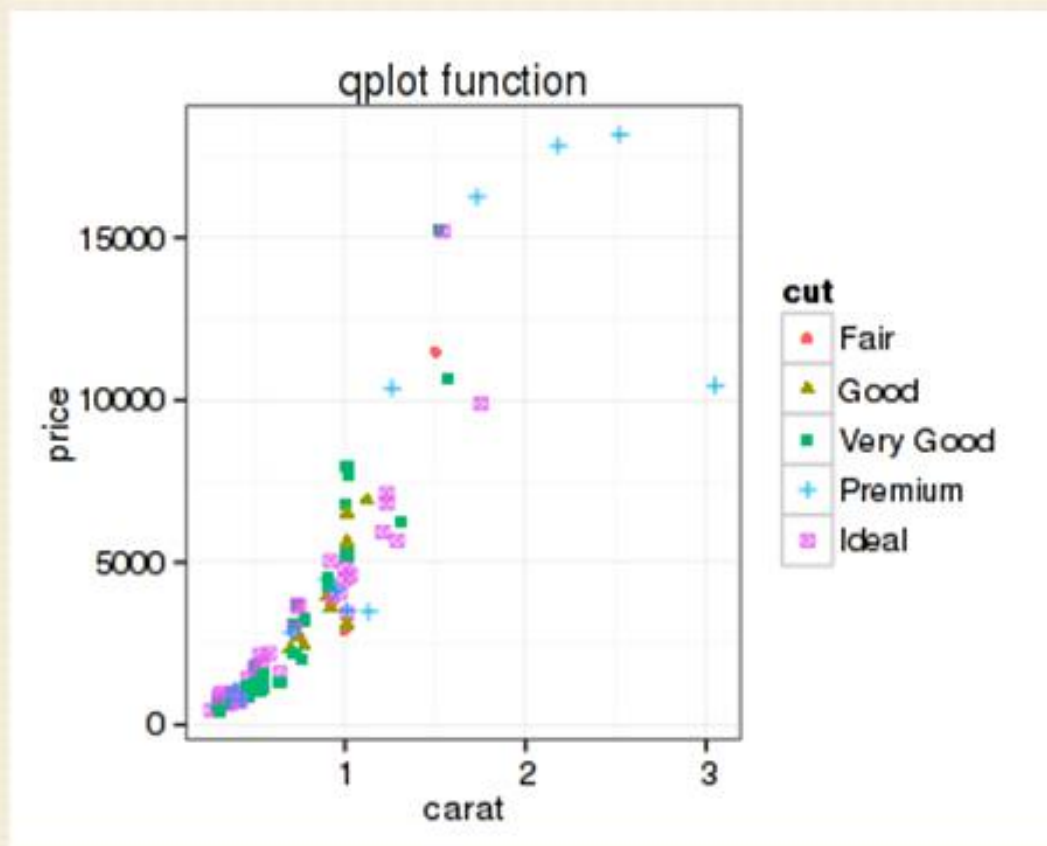
```
> qplot(x=carat, y=price, data=datax, color=cut, shape=cut, main="qplot function")
```



← 如果不喜欢它默认的图形背景，要改变也相当简单，
`ggplot2`预置了几个模板，这些内容我们在后面再详细
说：

```
> theme_set(theme_bw());
```

```
> qplot(x=carat, y=price, data=datax, color=cut, shape=cut, main="qplot function");
```



← 数据框可以存储不同的数据，而这些数据是有类型差别的。*ggplot2*作图对各类数据的要求也非常严格，用于分类的数据必需是因子类型，否则就出错，例如下面的语句就会出错：

```
qplot(x=carat, y=price, data=datax, shape=depth)
```

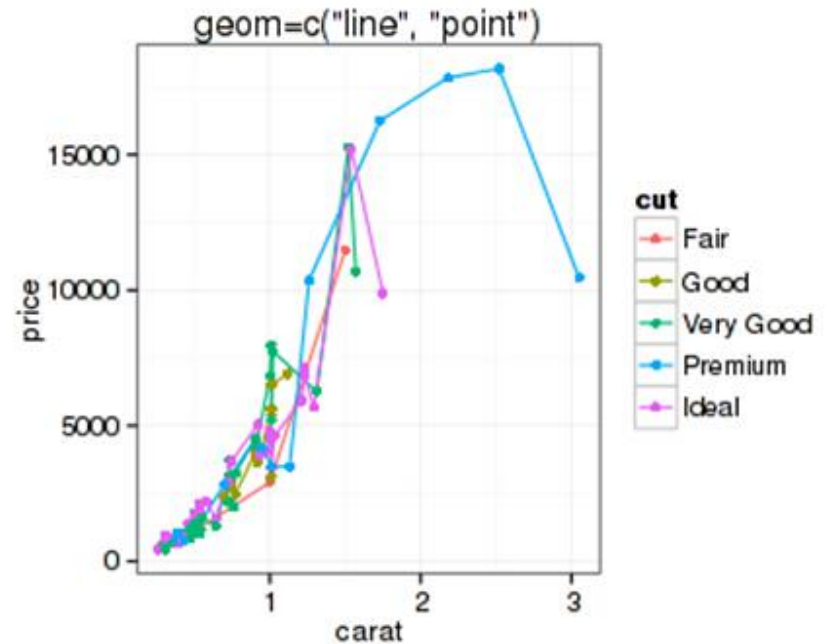
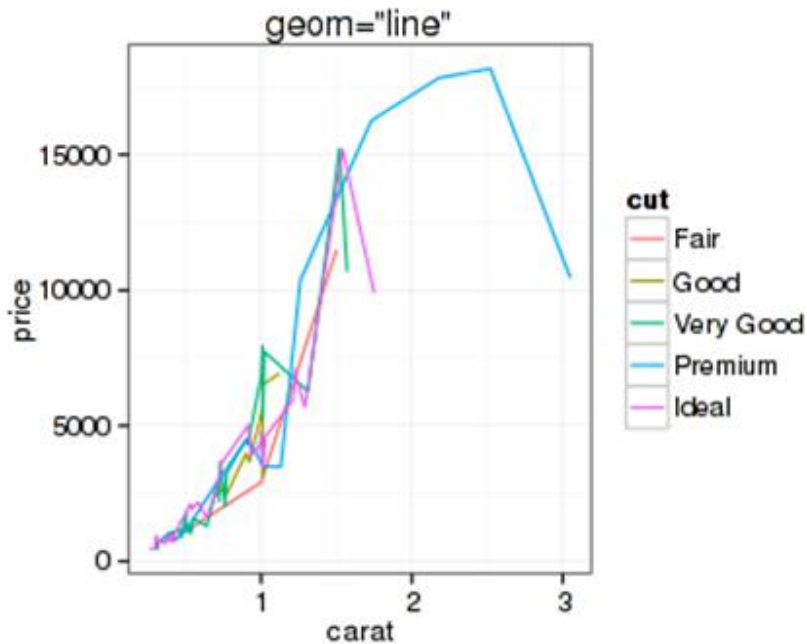
```
## Error: A continuous variable can not be mapped to shape
```

qplot做曲线图

← 和`plot`函数一样，`qplot`也可以通过设置合适的参数产生曲线图，这个参数就是`geom`（几何类型）。图形的组合非常直接，组合表示几何类型的向量即

```
> qplot(x=carat, y=price, data=datax, color=cut, geom="line", main="geom=\"line\"");
```

```
> qplot(x=carat, y=price, data=datax, color=cut, geom=c("line", "point"), main="geom=c(\"line\", \"point\")")
```



qplot做统计图

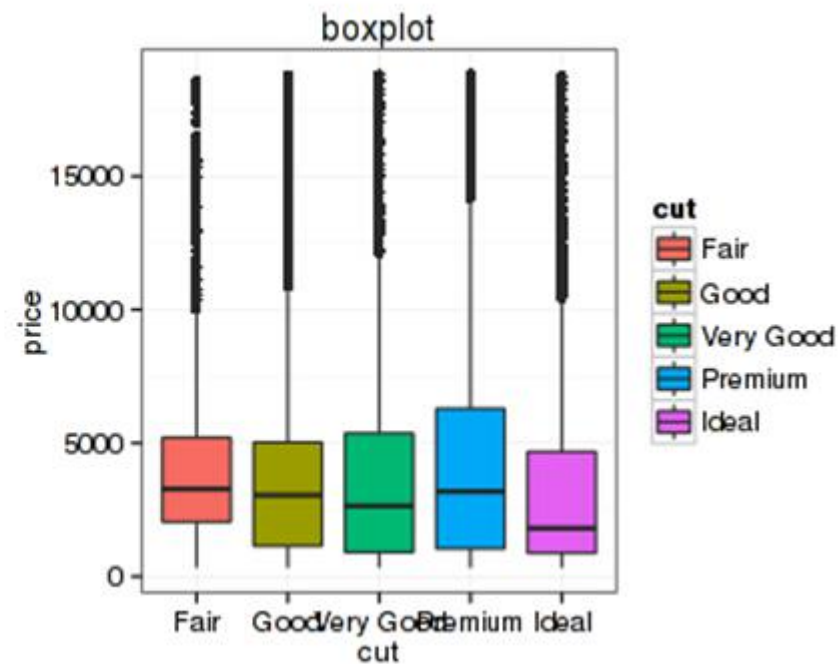
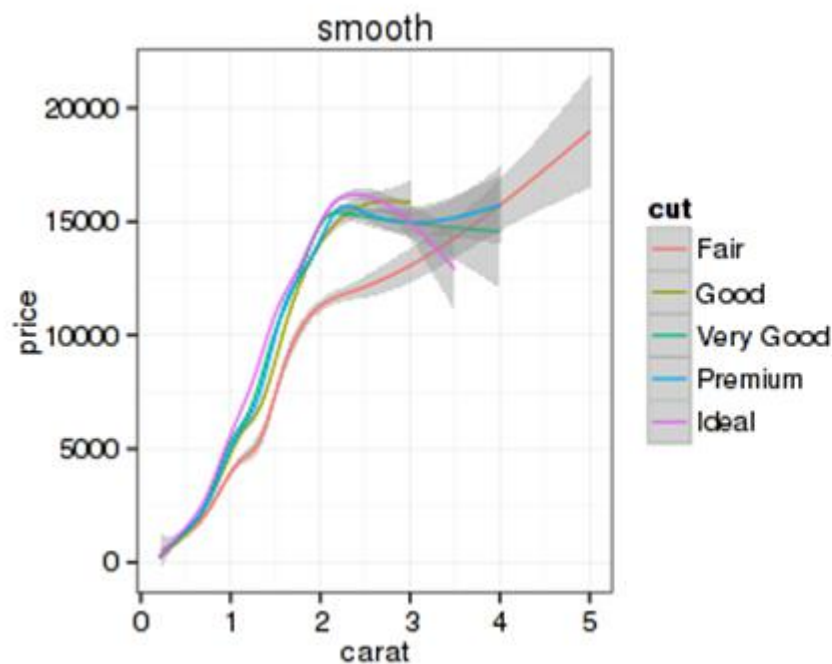
- ← *qplot*是名副其实的*qplot* (*quick plot*) 函数, 通过改变几何类型*geom*参数的值你可以获得各种图形。*geom*参数可以设置的值和意义是:
- ← *point*: 散点图
- ← *line*: 曲线图
- ← *smooth*: 平滑曲线
- ← *jitter*: 另一种散点图
- ← *boxplot*: 箱线图
- ← *histogram*: 直方图
- ← *density*: 密度分布图
- ← *bar*: 柱状图

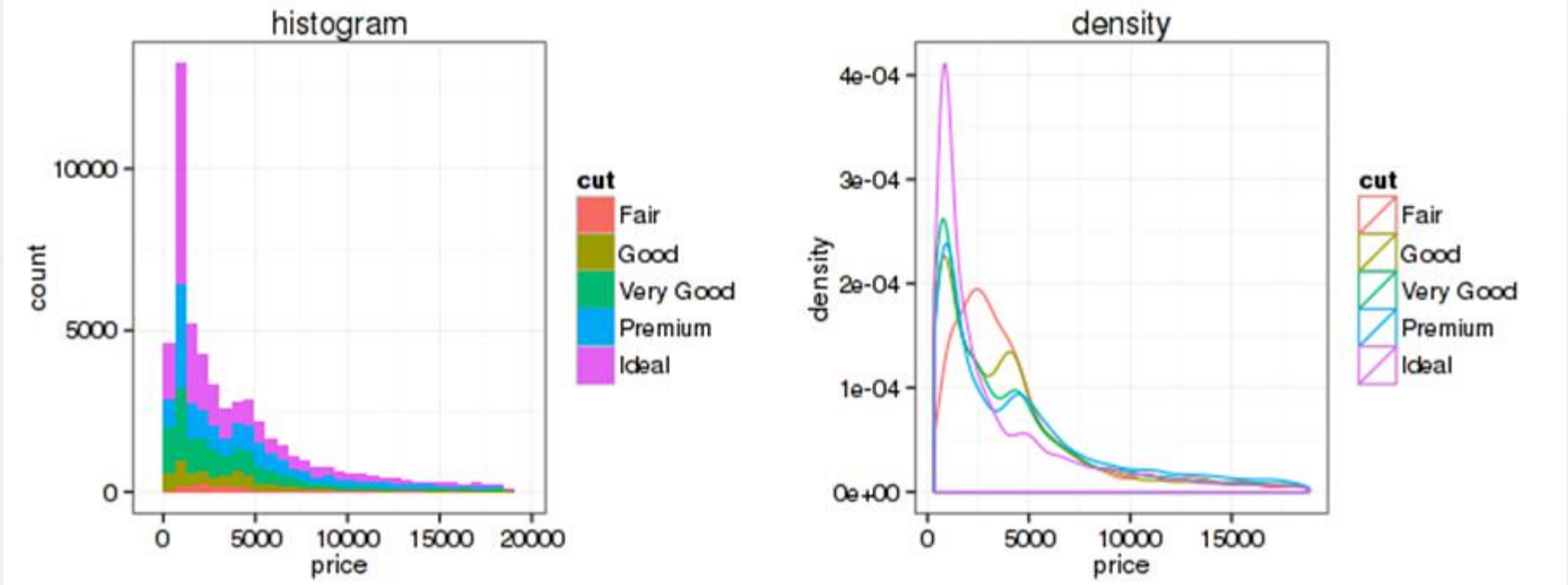
```
> qplot(carat, price, data = diamonds, color=cut, geom = "smooth", main = "smooth");
```

```
> qplot(cut, price, data = diamonds, fill=cut, geom = "boxplot", main = "boxplot");
```

```
> qplot(price, data = diamonds, fill=cut, geom = "histogram", main = "histogram");
```

```
> qplot(price, data = diamonds, color=cut, geom = "density", main = "density");
```





← 能做什么样的图形取决于数据，这点我们都很清楚，所以不同类型的图使用的数据有所不同，参数也有变化。前面我们说 `ggplot2` 可以整合不同类型的图形到一个图中，但很重要的一个前提是要组合的这些形状要能共享一组数据和参数。道理很简单，如果某人绞尽脑汁把散点图和密度分布图融合在一个图中展示，不出一周他就要住进精神病院。

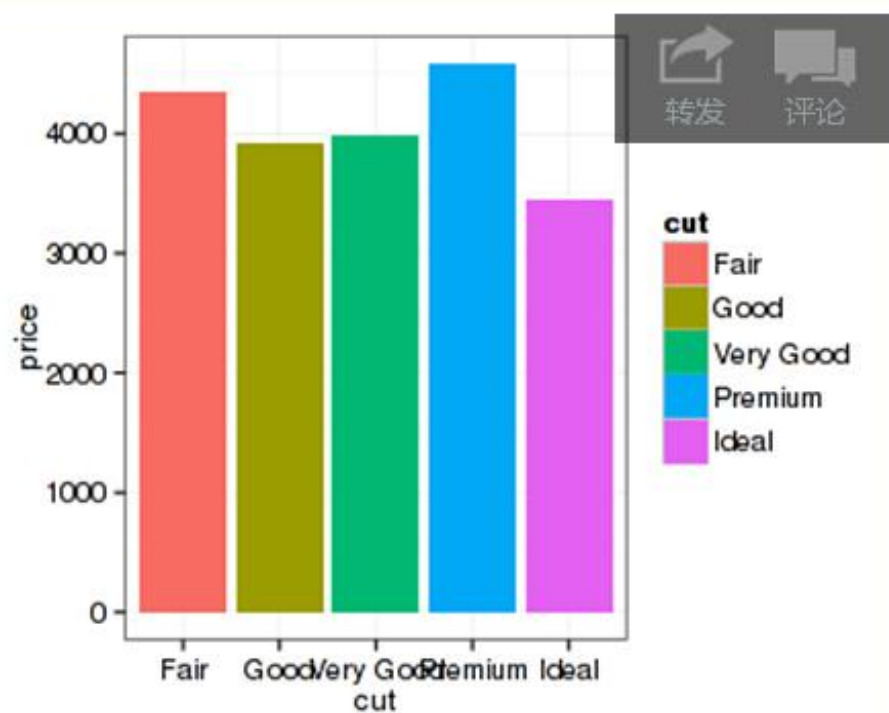
```
> qplot(price, data = diamonds, color=cut, geom = c("point", "density"))
```

```
## Error: geom_point requires the following missing aesthetics: y
```

qplot做柱形图

- ← 做柱形图很少直接用原始数据，一般都要通过计算变换如求平均值后再做。这其实是一个统计过程，所以多数柱形图应该也是统计类型的图。`ggplot2`对柱形图的处理体现了这一思想：柱形图是一种特殊的直方图。所以`ggplot2`可以直接用原始数据做出柱形图，这是它的优点之一。下面按钻石切工对价格求平均值后做柱形图：

```
> qplot(x=cut, y=price, data = diamonds, fill=cut, geom = "histogram", stat="summary", fun.y="mean")
```



← *stat*参数表示统计的类型，而*fun.y*则表示应用于统计的函数。把*geom*参数值换成*bar*得到相同的图形：

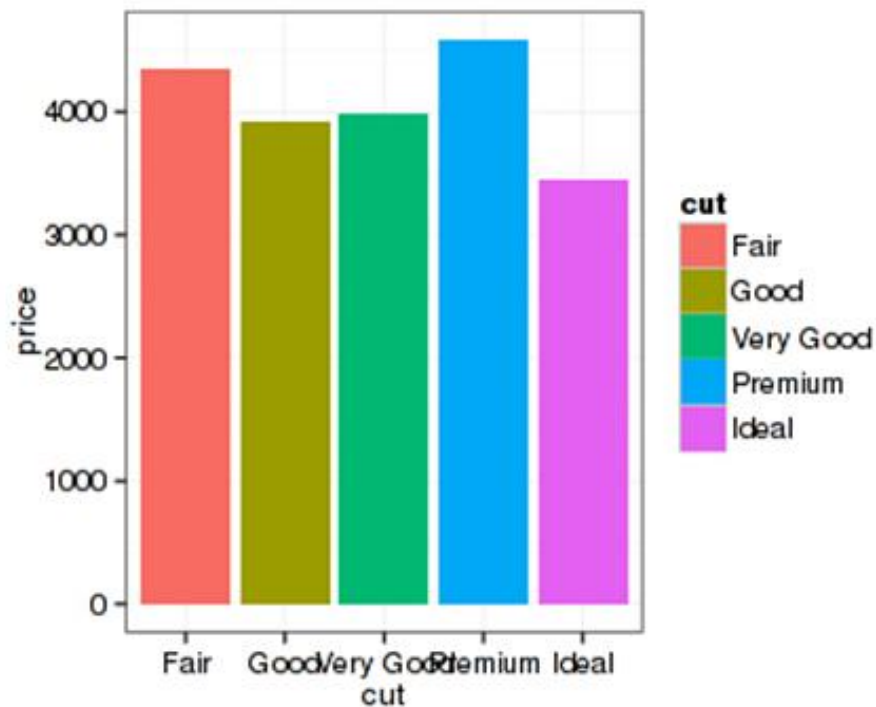
```
> qplot(x=cut, y=price, data =  
diamonds, fill=cut, geom = "bar",  
stat="summary", fun.y="mean")
```

如果不嫌麻烦，可以先计算出平均值再柱形图也没什么问题。引物
*bar*图形的本质是统计图形，所以得设置*stat*参数为*identity*，即
不做统计：


```
> mean.price <- with(diamonds, aggregate(price~cut, FUN=mean))
> mean.price
```

	cut	price
1	Fair	4358.758
2	Good	3928.864
3	Very Good	3981.760
4	Premium	4584.258
5	Ideal	3457.542

```
> qplot(x=cut, y=price, data=mean.price, fill=cut, geom="bar", stat="identity")
```



ggplot2总结:几个基本概念

- 数据 (Data) 和映射 (Mapping)
- 标度 (Scale)
- 几何对象 (Geometric)
- 统计变换 (Statistics)
- 坐标系 (Coordinate)
- 图层 (Layer)
- 分面 (Facet)

数据 (Data) 和映射 (Mapping)

将数据中的变量映射到图形属性。映射控制了二者之间的关系。

length	width	depth	trt
2	3	4	a
1	2	1	a
4	5	15	b
9	10	80	b



x	y	colour
2	3	a
1	2	a
4	5	b
9	10	b

标度 (Scale)

标度负责控制映射后图形属性的显示方式。具体形式上来看是图例和坐标刻度。Scale和Mapping是紧密相关的概念。

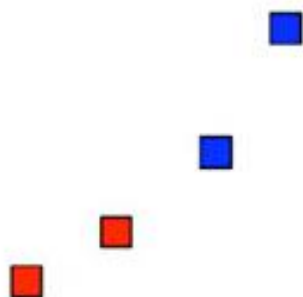
x	y	colour
2	3	a
1	2	a
4	5	b
9	10	b



x	y	colour
25	11	red
0	0	red
75	53	blue
200	300	blue

几何对象 (Geometric)

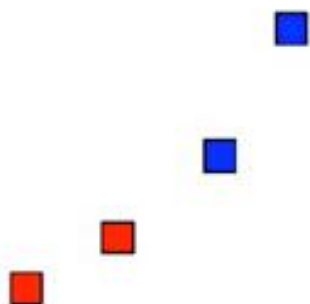
几何对象代表我们在图中实际看到的图形元素，如点、线、多边形等。



Geoms

统计变换 (statistics)

对原始数据进行某种计算，例如对二元散点图加上一条回归线。



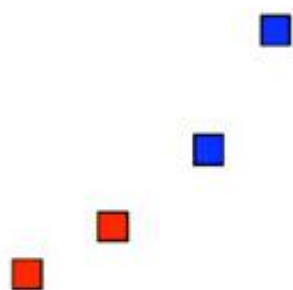
Geoms



Stat

坐标系 (Coordinate)

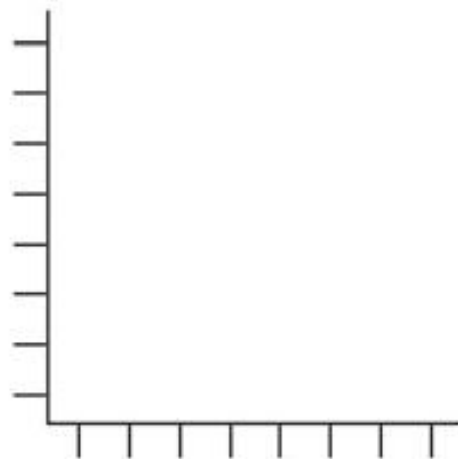
坐标系控制坐标轴并影响所有图形元素，坐标轴可以进行变换以满足不同的需要。



Geoms



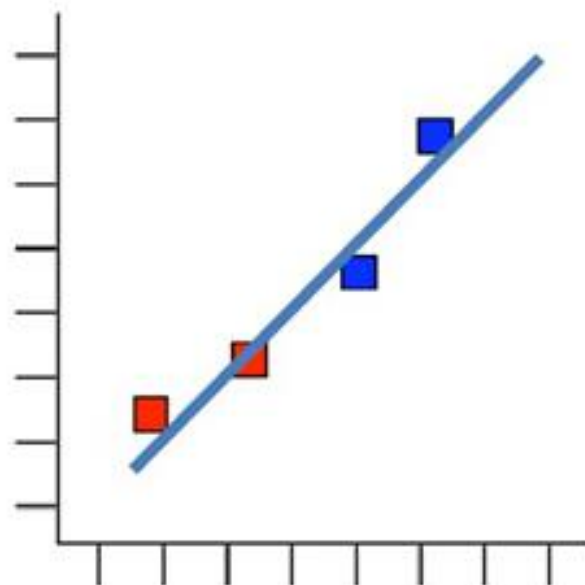
Stat



Coord

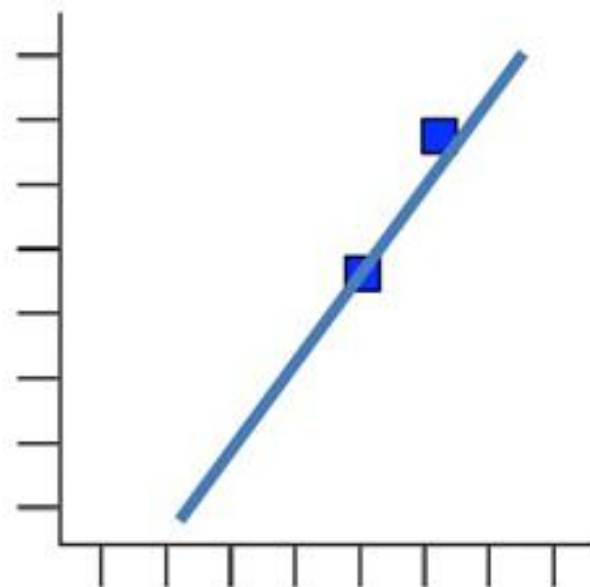
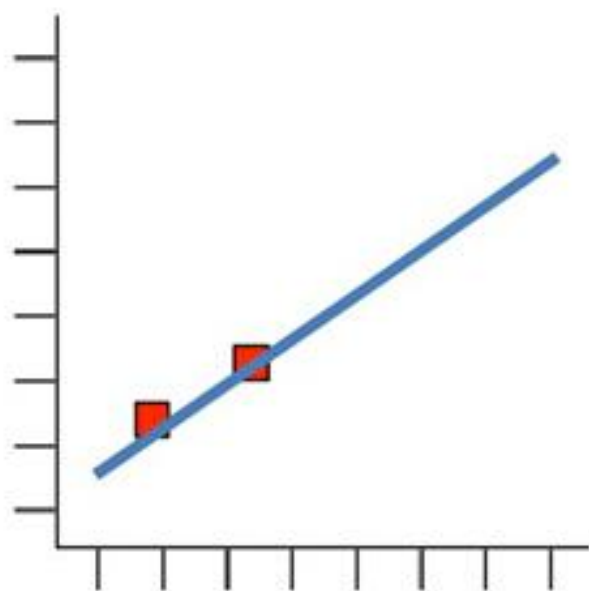
图层 (Layer)

数据、映射、几何对象、统计变换等构成一个图层。
图层可以允许用户一步步的构建图形，方便单独对图层进行修改。



分面 (Facet)

条件绘图，将数据按某种方式分组，然后分别绘图。
分面就是控制分组绘图的方法和排列形式。



(四)、文档的读写

- ← 本地文件（主要介绍，其它2种方式自学）
- ← 数据库文件
- ← 网页文件

载入数据是进行数据分析的第一步！学习一个新程序包中最困难的就是如何载入数据了！！！！

控制台的输入

- ← 用户可以通过控制台进行交互输入。
Readline函数可以输入单个字符串数据，
而scan函数可以输入多个数值数据。

```
> x <- readline()  
user  
> y <- scan()  
1: 3.1  
2: 2.3  
3: 4.5  
4:  
Read 3 items  
> y  
[1] 3.1 2.3 4.5  
> |
```

本地文件输出

- ← 使用file函数建立一个文件连接。使用cat函数可以直接将数据对象输出到文件连接中，如果文件已经有内容，可以在cat函数中设置append参数为真，即表示新增文件

```
> output <- file('output.txt')
> class(output)
[1] "file"          "connection"
> cat(1:100)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 2
1 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38
39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 5
6 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73
74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 9
1 92 93 94 95 96 97 98 99 100
> cat(1:100,sep='\t',file=output)
> getwd()
[1] "C:/Users/THINKPAD/Documents"
> close(output)
> |
```

本地文件输入

- ← 要注意scan函数读入的内容应该是一致的类型，即不能同时读入字符和数值。

```
> output <- file('output.txt')
> input <- scan(file=output)
Read 100 items
> close(output)
> head(input)
[1] 1 2 3 4 5 6
> input
 [1] 1 2 3 4 5 6 7 8 9 10 11
[12] 12 13 14 15 16 17 18 19 20 21 22
[23] 23 24 25 26 27 28 29 30 31 32 33
[34] 34 35 36 37 38 39 40 41 42 43 44
[45] 45 46 47 48 49 50 51 52 53 54 55
[56] 56 57 58 59 60 61 62 63 64 65 66
[67] 67 68 69 70 71 72 73 74 75 76 77
[78] 78 79 80 81 82 83 84 85 86 87 88
[89] 89 90 91 92 93 94 95 96 97 98 99
[100] 100
> |
```


字符串的输入输出

- ← 如果只需要处理字符串的输入输出，可以考虑readLines和writeLines这一对函数。

```
> output <- file('output.txt')
> writelines(as.character(1:12),con=output)
Error: could not find function "writelines"
> writeLines(as.character(1:12),con=output)
> input <- readLines(output)
> input
[1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"
[10] "10" "11" "12"
> |
```

数据表的读写

- ← 可以使用read.table和write.table函数；
- ← 另一种更方便的导入方法是利用Rstudio的功能，在workspace菜单中选现” import dataset”。

```
> head(iris) # iris为R内置的数据
  Sepal.Length Sepal.Width Petal.Length
1          5.1           3.5          1.4
2          4.9           3.0          1.4
3          4.7           3.2          1.3
4          4.6           3.1          1.5
5          5.0           3.6          1.4
6          5.4           3.9          1.7
  Petal.Width Species
1         0.2  setosa
2         0.2  setosa
3         0.2  setosa
4         0.2  setosa
5         0.2  setosa
6         0.4  setosa
> |
```

```
> write.table(iris,file='iris.csv',sep=',')
> data <- read.table(file='iris.csv',sep=',')
> head(data)
```

	Sepal.Length	Sepal.width	Petal.Length	Petal.width
1	5.1	3.5	1.4	0.2
2	4.9	3.0	1.4	0.2
3	4.7	3.2	1.3	0.2
4	4.6	3.1	1.5	0.2
5	5.0	3.6	1.4	0.2
6	5.4	3.9	1.7	0.4

	Species
1	setosa
2	setosa
3	setosa
4	setosa
5	setosa
6	setosa

Excel中数据的载人

两种方式：

- ← 第一种：1) 准备Excel数据；2) 将其提取到制表符分隔的ascii文件中；3) 关闭Excel；4) 使用read.table函数将数据载入到R中。
- ← 第二种：下载一个专门的R程序包，RODBC，它可以访问Excel中选定的行和列。

第三节 SQL及数据库编程

一、数据库和数据库技术

1、数据库

2、生物数据库

NCBI、KEGG、PDB...

3、数据库技术支持

技术要素：

RDBMS数据库程序（如MS Access、SQL Server、MySQL）

服务器端脚本语言（如PHP或ASP）

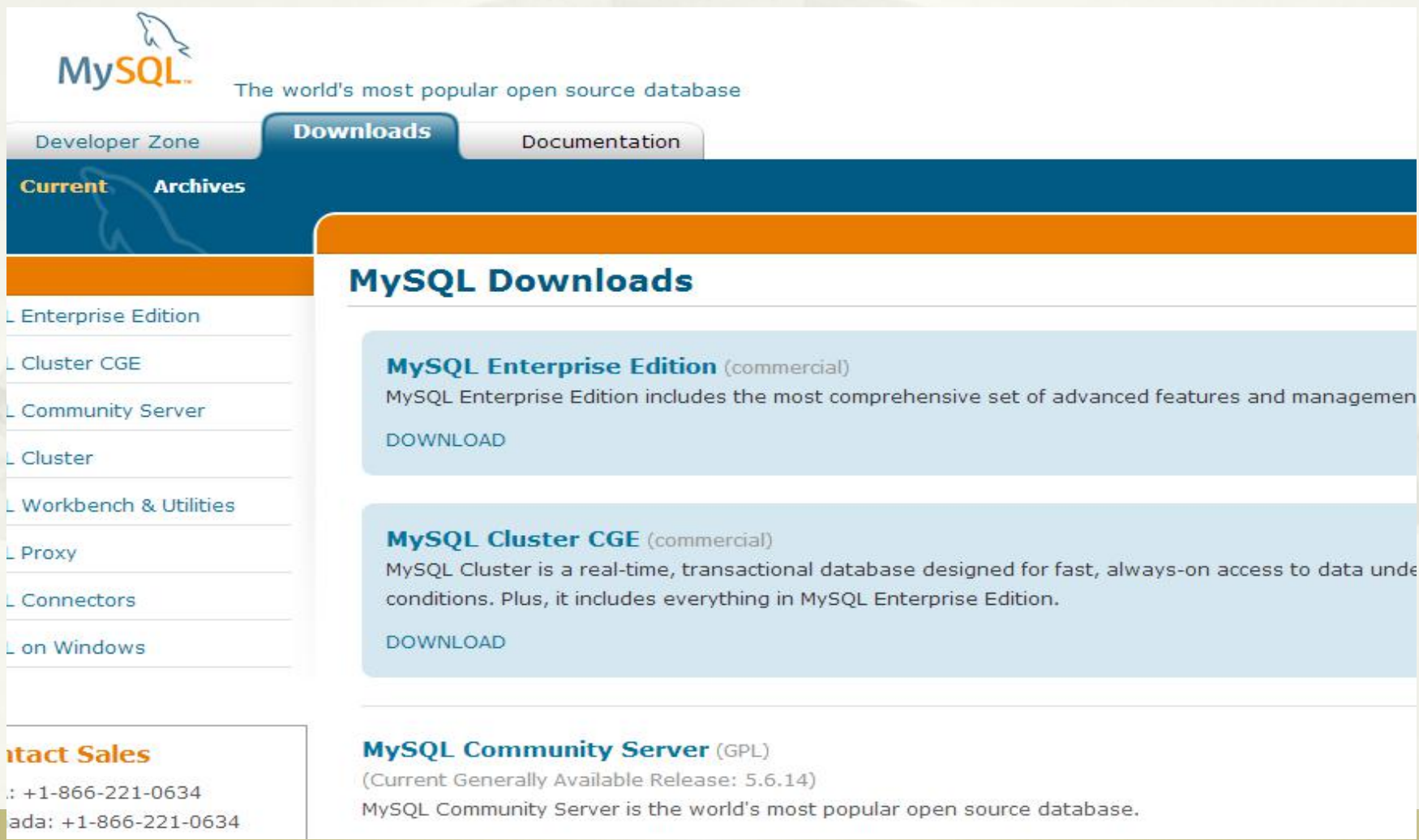
SQL

HTML/CSS

二、MySQL

1、MySQL安装和客户端

开源 (<http://www.mysql.com/>)



The screenshot shows the MySQL website's 'Downloads' section. The header includes the MySQL logo and the tagline 'The world's most popular open source database'. Navigation tabs for 'Developer Zone', 'Downloads', and 'Documentation' are present, with 'Downloads' being the active tab. Below this, there are sub-tabs for 'Current' and 'Archives'. A left sidebar lists various MySQL products and utilities, including Enterprise Edition, Cluster CGE, Community Server, Cluster, Workbench & Utilities, Proxy, Connectors, and on Windows. The main content area, titled 'MySQL Downloads', features three product cards: 'MySQL Enterprise Edition (commercial)', 'MySQL Cluster CGE (commercial)', and 'MySQL Community Server (GPL)'. Each card provides a brief description and a 'DOWNLOAD' link. The 'MySQL Community Server' card also mentions the 'Current Generally Available Release: 5.6.14' and states it is the world's most popular open source database. A 'Contact Sales' section is visible at the bottom left of the page.

MySQL
The world's most popular open source database

Developer Zone Downloads Documentation

Current Archives

MySQL Downloads

MySQL Enterprise Edition (commercial)
MySQL Enterprise Edition includes the most comprehensive set of advanced features and management tools.
DOWNLOAD

MySQL Cluster CGE (commercial)
MySQL Cluster is a real-time, transactional database designed for fast, always-on access to data under the most demanding conditions. Plus, it includes everything in MySQL Enterprise Edition.
DOWNLOAD

MySQL Community Server (GPL)
(Current Generally Available Release: 5.6.14)
MySQL Community Server is the world's most popular open source database.

Contact Sales
Tel: +1-866-221-0634
Canada: +1-866-221-0634

管理员: C:\Windows\system32\cmd.exe - mysql -uroot -p

C:\Users\Junjie>mysql -uroot -p

Enter password:

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 12

Server version: 5.1.37-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> _

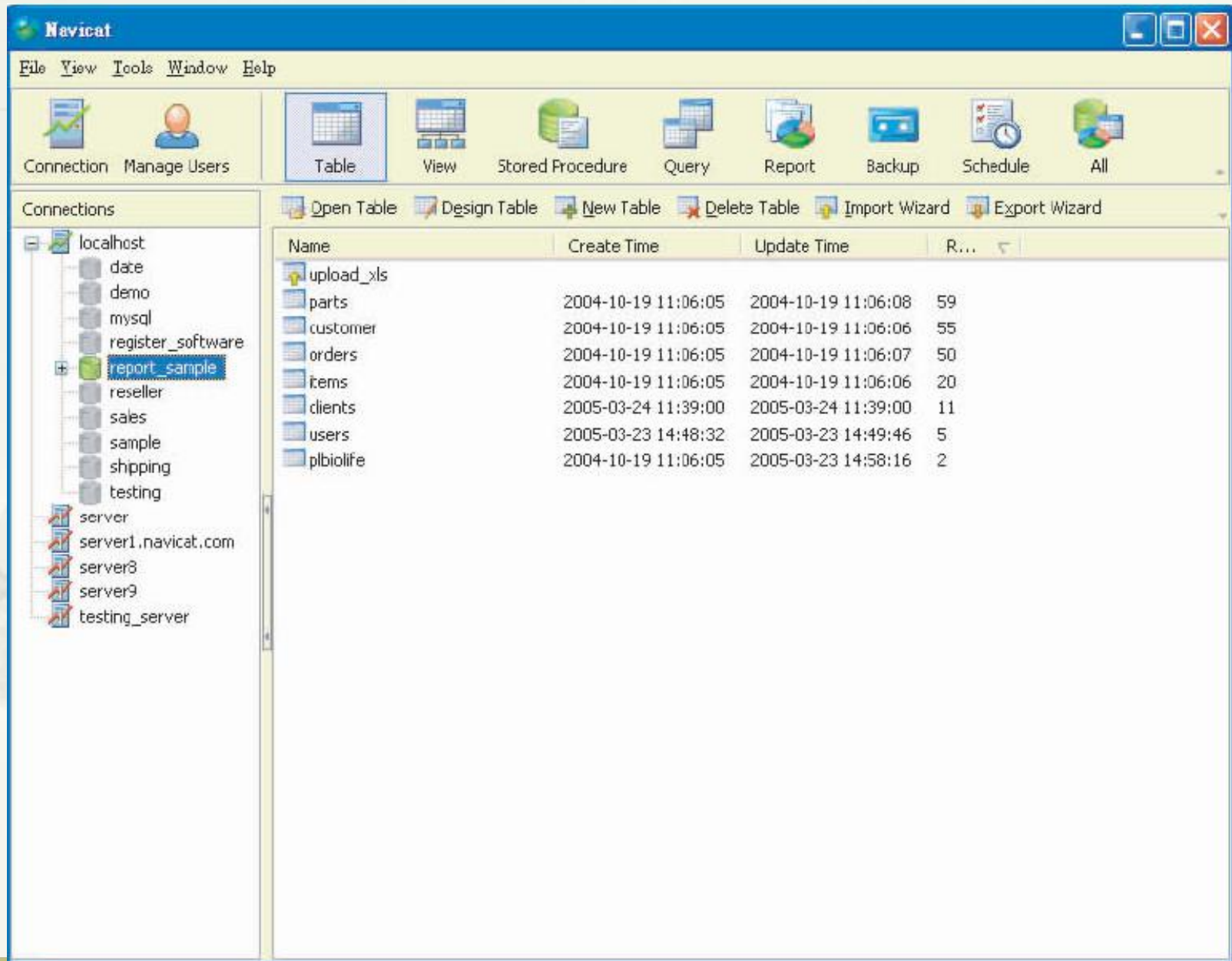
半:

(2) phpMyAdmin

The screenshot shows the phpMyAdmin 2.10.3 web interface. The browser window title is "localhost / localhost | phpMyAdmin 2.10.3 - Mozilla Firefox". The address bar shows "http://localhost/phpmyadmin/". The interface is divided into several sections:

- Left Sidebar:** Contains the phpMyAdmin logo, navigation icons (Home, Servers, Databases, SQL, Search), a "Database" section with a dropdown menu showing "(Databases) ...", and the text "Please select a database".
- localhost Section:** Displays server information for "localhost":
 - Server version: 5.0.51b-community-nt-log
 - Protocol version: 10
 - Server: localhost via TCP/IP
 - User: root@localhost
 - MySQL charset: **UTF-8 Unicode (utf8)**
 - MySQL connection collation: utf8_unicode_ci
 - Create new database: [Text input field] Collation: [Dropdown menu] [Create button]
 - Links: Show MySQL runtime information, Show MySQL system variables, Processes, Character Sets and Collations, Storage Engines, Reload privileges, Privileges, Databases, Export, Import, Log out.
- phpMyAdmin - 2.10.3 Section:** Displays client and configuration information:
 - MySQL client version: 5.0.51a
 - Used PHP extensions: mysql
 - Language: English (dropdown menu)
 - Theme / Style: Original (dropdown menu)
 - Font size: 100% (dropdown menu)
 - Links: phpMyAdmin documentation, phpMyAdmin wiki, Official phpMyAdmin Homepage, [ChangeLog] [Subversion] [Lists]
- Footer:** The phpMyAdmin logo is displayed in the bottom right corner.

(3) Navicat



2、基本概念

(1) 表、行、列与列类型

(2) 键

键是特殊的列，通过它可以实现实体间的链接。主键（primary key）是某些规则必须遵守的唯一标识符。

(3) 表关系

one-to-one、one-to-many、many-to-many

3、数据库设计与范式

数据库设计，就是建立数据库的结构，又称数据建模（data modeling）。

规范化，就是在数据库设计当中要遵循某些规则，这些规则被称为范式（normal form）。

1NF、2NF、3NF

4、数据库操作

Create

Insert

Select

Drop

Delete

Update

Alter

```
管理员: C:\Windows\system32\cmd.exe - mysql -uroot -p

mysql> CREATE DATABASE db_paper;
Query OK, 1 row affected (0.06 sec)

mysql> USE db_paper;
Database changed

mysql> CREATE TABLE journal (
  -> id INT UNSIGNED NOT NULL AUTO_INCREMENT,
  -> journal_title VARCHAR(50) NOT NULL,
  -> issn VARCHAR(20) NOT NULL,
  -> total_cites MEDIUMINT NOT NULL,
  -> impact_factor FLOAT NOT NULL,
  -> PRIMARY KEY (id)
  -> );
Query OK, 0 rows affected (0.47 sec)

mysql> CREATE TABLE paper(
  -> id INT UNSIGNED NOT NULL AUTO_INCREMENT,
  -> title VARCHAR(100) NOT NULL,
  -> journal_id INT UNSIGNED NOT NULL,
  -> PRIMARY KEY (id)
  -> );
Query OK, 0 rows affected (0.12 sec)

mysql> SHOW TABLES;
+-----+
| Tables_in_db_paper |
+-----+
| journal             |
| paper               |
+-----+
2 rows in set (0.02 sec)

mysql> EXPLAIN journal;
+-----+-----+-----+-----+-----+-----+
| Field      | Type                | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | int(10) unsigned   | NO   | PRI | NULL    | auto_increment |
| journal_title | varchar(50)        | NO   |     | NULL    |                |
| issn       | varchar(20)        | NO   |     | NULL    |                |
| total_cites | mediumint(9)       | NO   |     | NULL    |                |
| impact_factor | float              | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.06 sec)

mysql> _ 半:
```

三、PHP

1、PHP简介

实用性、可选择性、开源与低成本



[downloads](#) | [documentation](#) |

What is PHP?

PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. If you are new to PHP and want to get some idea of how it works, try the [introductory tutorial](#). After that, check out the online [manual](#).

Ever wondered how popular PHP is? See the [Netcraft Survey](#).

Thanks To

[easyDNS](#)
[Directi](#)
[pair Networks](#)
[Server Central](#)

Upcoming conferences: [Ski PHP 2014](#) [Madison PHP Conference](#) [CodeConnexx 2013](#) [International PHP Conference](#)
Calling for papers: [Ski PHP 2014](#)

PHP 5.4.21 Released

17-Oct-2013 The PHP development team announces the immediate availability of PHP 5.4.21. About 10 bugs fixed in this version.

For source downloads of PHP 5.4.21 please visit our [downloads page](#), Windows binaries can be found on [Windows ChangeLog](#).

PHP 5.5.5 has been released

16-Oct-2013 The PHP development team announces the immediate availability of PHP 5.5.5. This release fixes a number of bugs in the build system. All PHP users are encouraged to upgrade to this new version.

For source downloads of PHP 5.5.5 please visit our [downloads page](#), Windows binaries can be found on [Windows ChangeLog](#).

PHP 5.4.20 released

第四节 并行计算

并行计算（parallel computing）是一种许多计算同时进行的计算形式，其基于大问题常可以分割成较小的问题并同时（并行）解决的原理。

一、概述

二、硬件



图 11-9 Cray-1

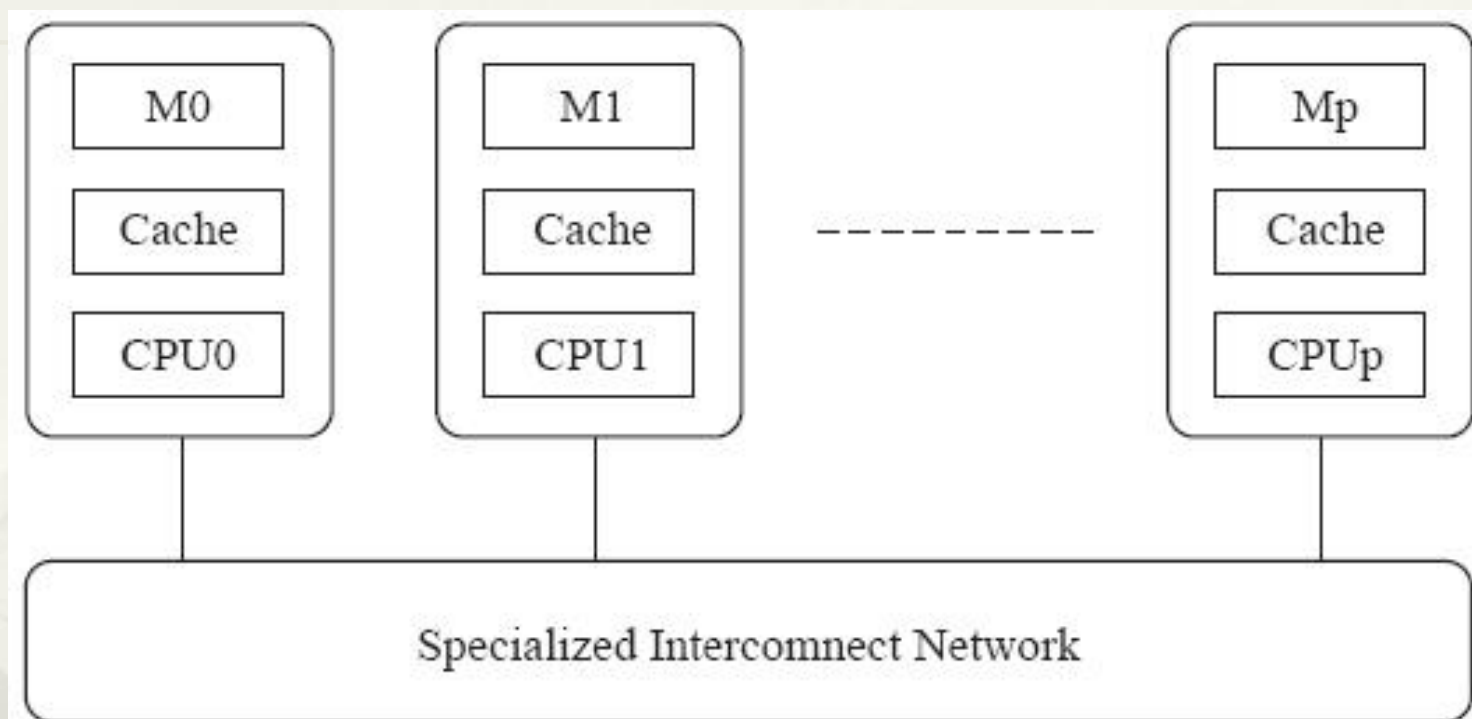


图 11-10 大规模并行处理机的系统结构

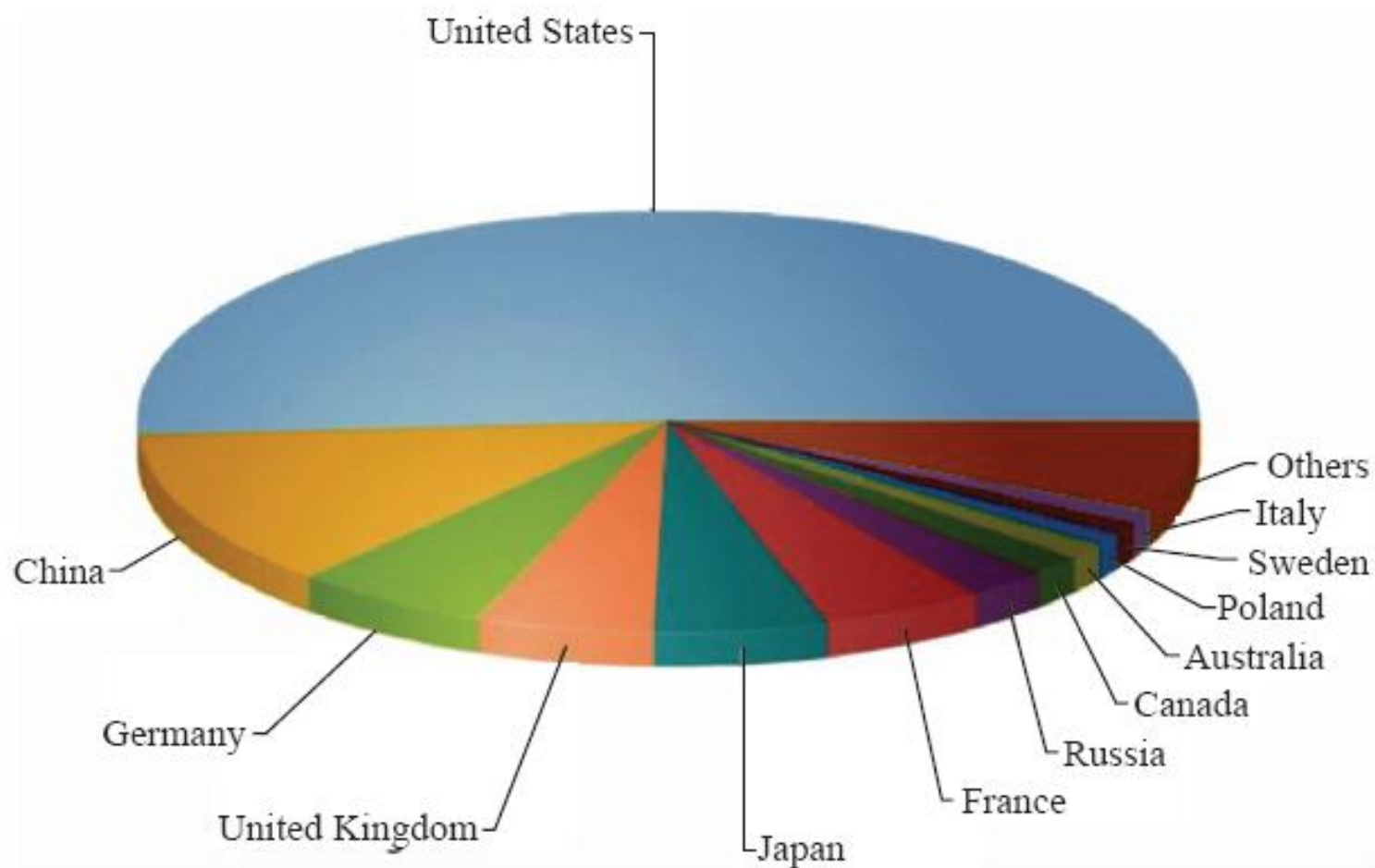


图 11-11 Top 500 按国家分布(资料来源: www.top500.org)



图 11-12 NBCR 集群

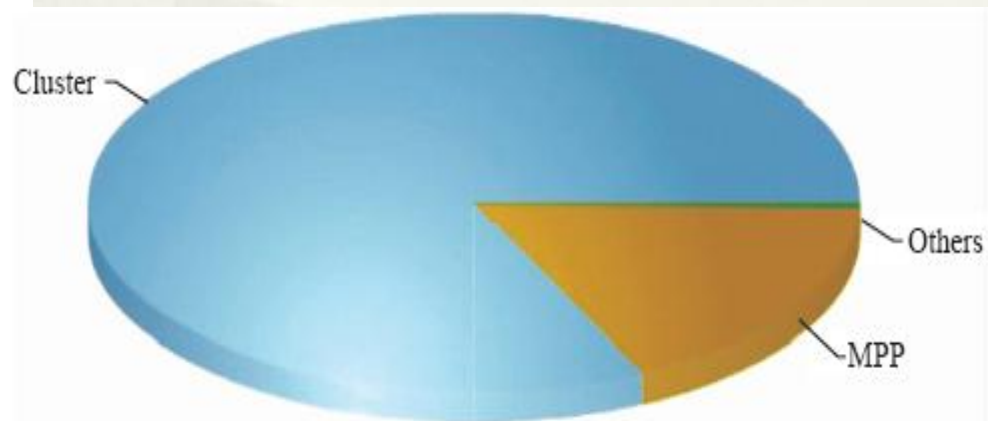


图 11-13 Top500 按体系结构分布(资料来源:www.top500.org)



三、软件

共享存储、分布式存储、共享分布式存储

四、并行算法设计

1、基本概念

粒度、加速比、效率、性能

2、基本方法

3、并行计算的实现

五、云计算

云计算是分布式计算技术的一种，是指通过网络将庞大的计算处理程序自动分拆成无数个较小的子程序，再交由多部服务器所组成的庞大系统经搜寻、计算分析之后将处理结果回传给用户。

特点：

超大规模、虚拟化、高可靠性、通用性、高可扩展性、按需服务、及其廉价。