

## R

Gang Chen  
chengang@genomics.cn

November 7, 2015

# Outline

- 1 Object-Oriented Programming
- 2 Data Visualization
- 3 R Package
- 4 Bioconductor
- 5 Reproducible Research in R

# Next

- 1 Object-Oriented Programming
  - History
    - S3
    - S4
- 2 Data Visualization
- 3 R Package
- 4 Bioconductor
- 5 Reproducible Research in R

# S4 Classes and methods

## History

- 1976, Rick Becker and John Chambers, S on Honeywell OS
- Ported to UNIX, S2
- Around 1986, functional programming and object self-description, S3
- 1992, concept of classes and methods, S4
- 2010, Reference Classes (RC), R 2.12

appendix in Software for Data Analysis by Chambers

# S4 Classes and methods

## OO Systems in R

- S3
- S4
- RC
- Base Types

Best Reference: <http://adv-r.had.co.nz/OO-essentials.html>

## S3

# S4 Classes and methods

## S4 in R

```
library(stats4)
library(pryr)
y <- c(26, 17, 13, 12, 20, 5, 9, 8, 5, 4, 8)
nLL <- function(lambda) -sum(dpois(y, lambda, log = TRUE))
fit <- mle(nLL, start = list(lambda = 5), nobs = length(y))
isS4(fit)

## [1] TRUE

otype(fit)

## [1] "S4"

isS4(nobs)

## [1] TRUE

ftype(nobs)
```

# S4 Classes and methods

## Defining classes and creating objects

```
setClass("Person",  
  slots = list(name = "character", age = "numeric"))  
setClass("Employee",  
  slots = list(boss = "Person"),  
  contains = "Person")  
  
alice <- new("Person", name = "Alice", age = 40)  
john <- new("Employee", name = "John", age = 20, boss = alice)
```



# S4 Classes and methods

access slots of an S4 object

```
alice@age  
slot(john, "boss")
```

# S4 Classes and methods

## Creating new methods and generics

```
setGeneric("union")
setMethod("union",
  c(x = "data.frame", y = "data.frame"),
  function(x, y) {
    unique(rbind(x, y))
  }
)
setGeneric("myGeneric", function(x) {
  standardGeneric("myGeneric")
})
```

# Next

## 1 Object-Oriented Programming

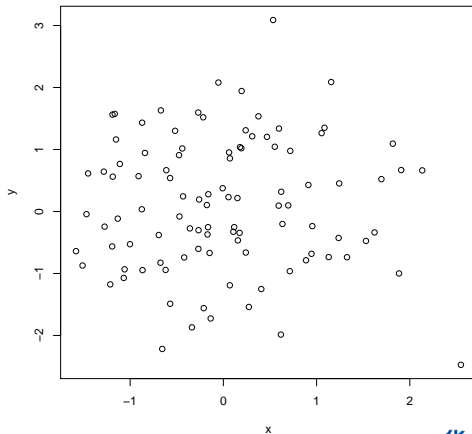
## 2 Data Visualization

- Scatter Plot
- 时间序列
- 柱状图
- 饼图
- 分类数据绘图
- 绘制分布
- 箱线图

## 3 R Package

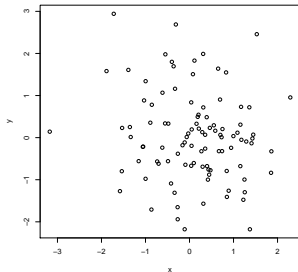
## 4 Bioconductor

## plot



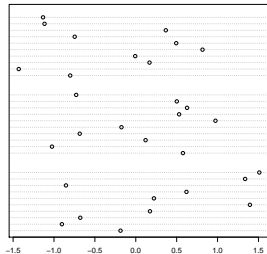
# plot

```
x = rnorm(100)  
y = rnorm(100)  
plot(x, y)
```



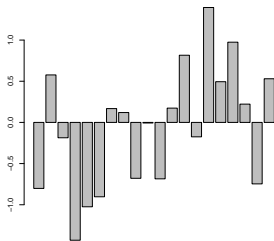
# dotchart

```
x = rnorm(30)
dotchart(x, groups = rep(1:3,10))
```



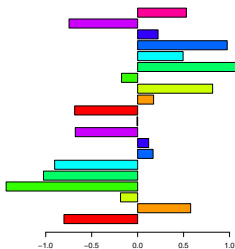
# barplot

```
barplot(x[1:20])
```



# barplot

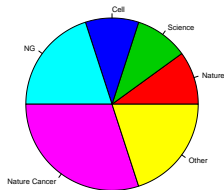
```
barplot(x[1:20], width=2, horiz=T, col=rainbow(10))
```





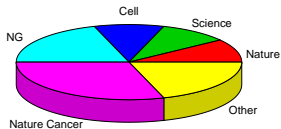
## pie

```
pie(c(10,10,10,20,30,20), c("Nature","Science","Cell","NG",  
Cancer","Other"),col=2:7)
```



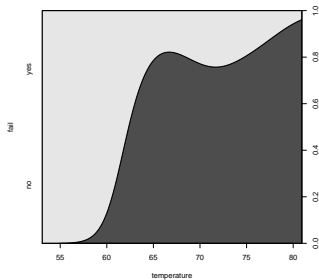
## pie

```
library(plotrix)
pie3D(c(10,10,10,20,30,20), labels=c("Nature", "Science", "Cell", "Nature Cancer", "Other", "NG"), col=2:7)
```



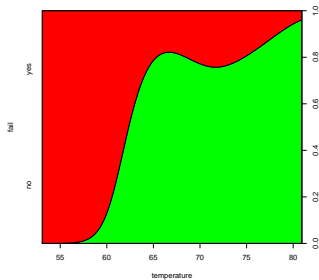
# cdplot

```
cdplot(temperature, fail)
```



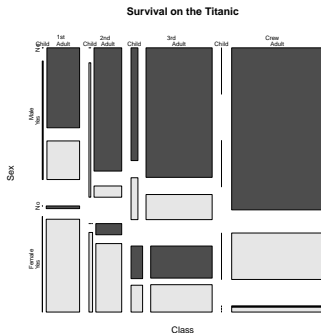
# cdplot

```
cdplot(temperature, fail, col=c("green", "red"))
```



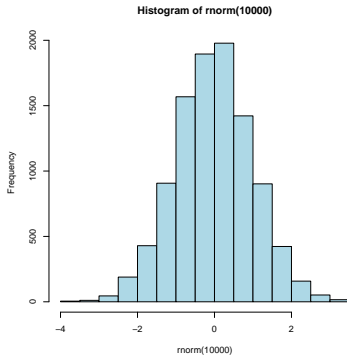
# mosaicplot

```
require(stats)
mosaicplot(Titanic, main = "Survival on the Titanic",
color = TRUE)
```



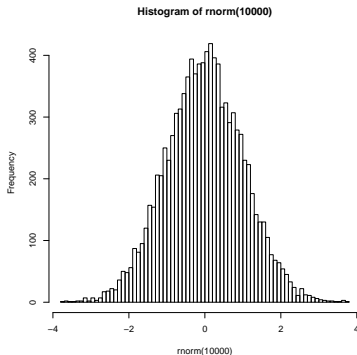
# hist

```
hist(rnorm(10000), col="lightblue")
```



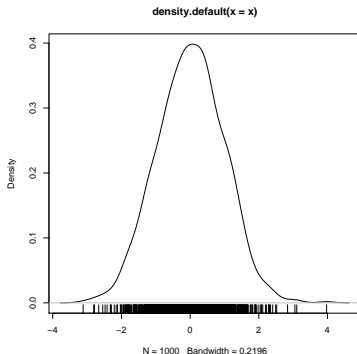
# hist

```
hist(rnorm(10000), breaks=100)
```



# density + rug

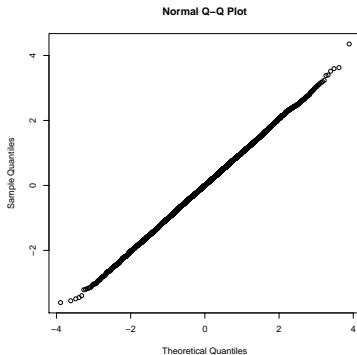
```
x = rnorm(1000)
plot(density(x))
rug(x)
```





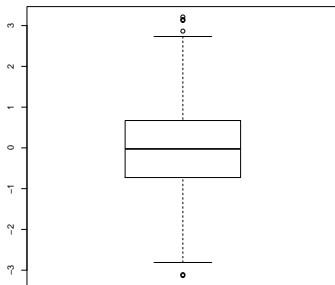
# Q-Q plot

```
qqnorm(rnorm(10000))
```



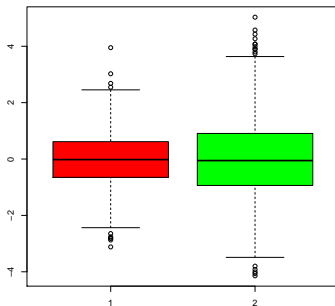
# boxplot

```
boxplot(rnorm(1000))
```



# boxplot

```
boxplot(cbind(rnorm(1000), rnorm(1000)+rnorm(1000)), col=c("red", "green"))
```



# Next

- 1 Object-Oriented Programming
- 2 Data Visualization
- 3 **R Package**
  - R Package Development
  - devtools
- 4 Bioconductor
- 5 Reproducible Research in R

# R Package

- Hadley: In R, the fundamental unit of shareable code is the package.
- Hilary Parker: Seriously, it doesn't have to be about sharing your code (although that is an added benefit!). It is about saving yourself time.

# References

- Writing R Extensions:  
<http://cran.r-project.org/manuals.html>
- R Packages from Hadley:  
<http://r-pkgs.had.co.nz/>
- Writing an R package from scratch:  
<http://hilaryparker.com/2014/04/29/writing-an-r-package-from-scratch/>
- 开发R程序包之忍者篇:  
<http://cos.name/2011/05/write-r-packages-like-a-ninja/>

# R Package from Scratch

see cgr directory

# Why devtools?

- This book espouses my philosophy of package development:
- anything that can be automated, should be automated.
- Do as little as possible by hand.
- Do as much as possible with functions.



# Next

- 1 Object-Oriented Programming
- 2 Data Visualization
- 3 R Package
- 4 Bioconductor**
  - Overview
  - ggbio
- 5 Reproducible Research in R

# Overview

## Bioconductor

Bioconductor provides tools for the analysis and comprehension of high-throughput genomic data. Bioconductor uses the R statistical programming language, and is open source and open development. It has two releases each year, 934 software packages, and an active user community.

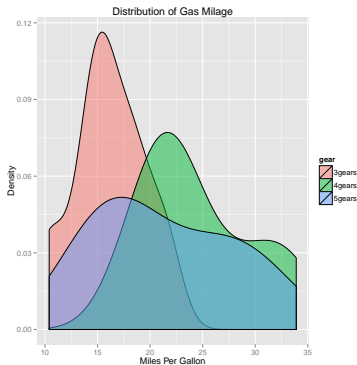
# Goals

- To provide widespread access to a broad range of powerful statistical and graphical methods for the analysis of genomic data.
- To facilitate the inclusion of biological metadata in the analysis of genomic data, e.g. literature data from PubMed, annotation data from Entrez genes.
- To provide a common software platform that enables the rapid development and deployment of extensible, scalable, and interoperable software.
- To further scientific understanding by producing high-quality documentation and reproducible research.
- To train researchers on computational and statistical methods for the analysis of genomic data.

see GNB5010-2013/4. Biological Data Analysis and Visualization in R/slides.pdf

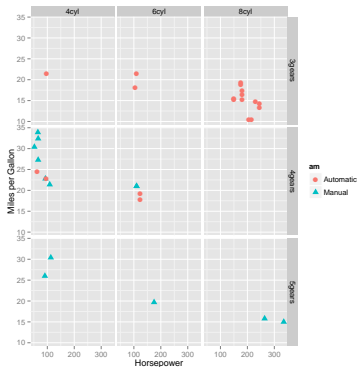
# ggplot2

```
qplot(mpg, data=mtcars, geom="density", fill=gear, alpha=I(.5), main="Distribution of Gas Milage", xlab="Miles Per Gallon", ylab="Density")
```



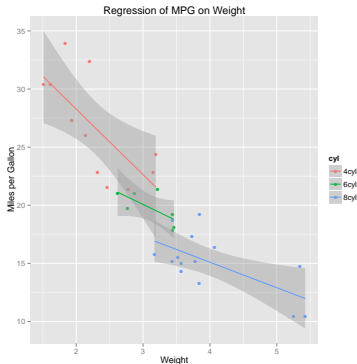
# ggplot2

```
qplot(hp, mpg, data=mtcars, shape=am, color=am, facets=gear,
size=I(3), xlab="Horsepower", ylab="Miles per Gallon")
```



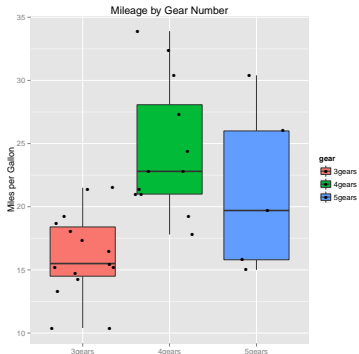
# ggplot2

```
qplot(wt, mpg, data=mtcars, geom=c("point", "smooth"),  
method="lm", formula=y ~ x, color=cyl, main="Regression  
of MPG on Weight", xlab="Weight", ylab="Miles per Gallon")
```



# ggplot2

```
qplot(gear, mpg, data=mtcars, geom=c("boxplot", "jitter"),  
fill=gear, main="Mileage by Gear Number", xlab="", ylab="Miles  
per Gallon")
```





# ggbio

- ggplot2 + bioconductor = ggbio
- Website: <http://www.tengfei.name/ggbio/>
- Author: Tengfei Yin at Seven Bridges Genomics

# ggbio Examples

```
source("http://bioconductor.org/biocLite.R")  
biocLite("ggbio")  
library(ggbio)  
example(autoplot)
```

# Next

- 1 Object-Oriented Programming
- 2 Data Visualization
- 3 R Package
- 4 Bioconductor
- 5 Reproducible Research in R**
  - knitr
  - Interactive Report and Shiny

# Overview

- Official website: <http://yihui.name/knitr/>
- Reference: Dynamic Documents with R and knitr
- Author: Yihui Xie

# Examples

- the slides of the R lectures are generated by knitr
  - Knitr
  - XeLaTeX
- see knitr directory for the Knitr example in Markdown

# Interactive Report

- Google Analytics
- 百度统计 from Baidu.com
- 数据魔方 from Taobao.com

# Shiny Overview

- A web application framework for R
- Turn your analyses into interactive web applications
- No HTML, CSS, or JavaScript knowledge required
- <http://shiny.rstudio.com/>

# Example

see shinyApp directory