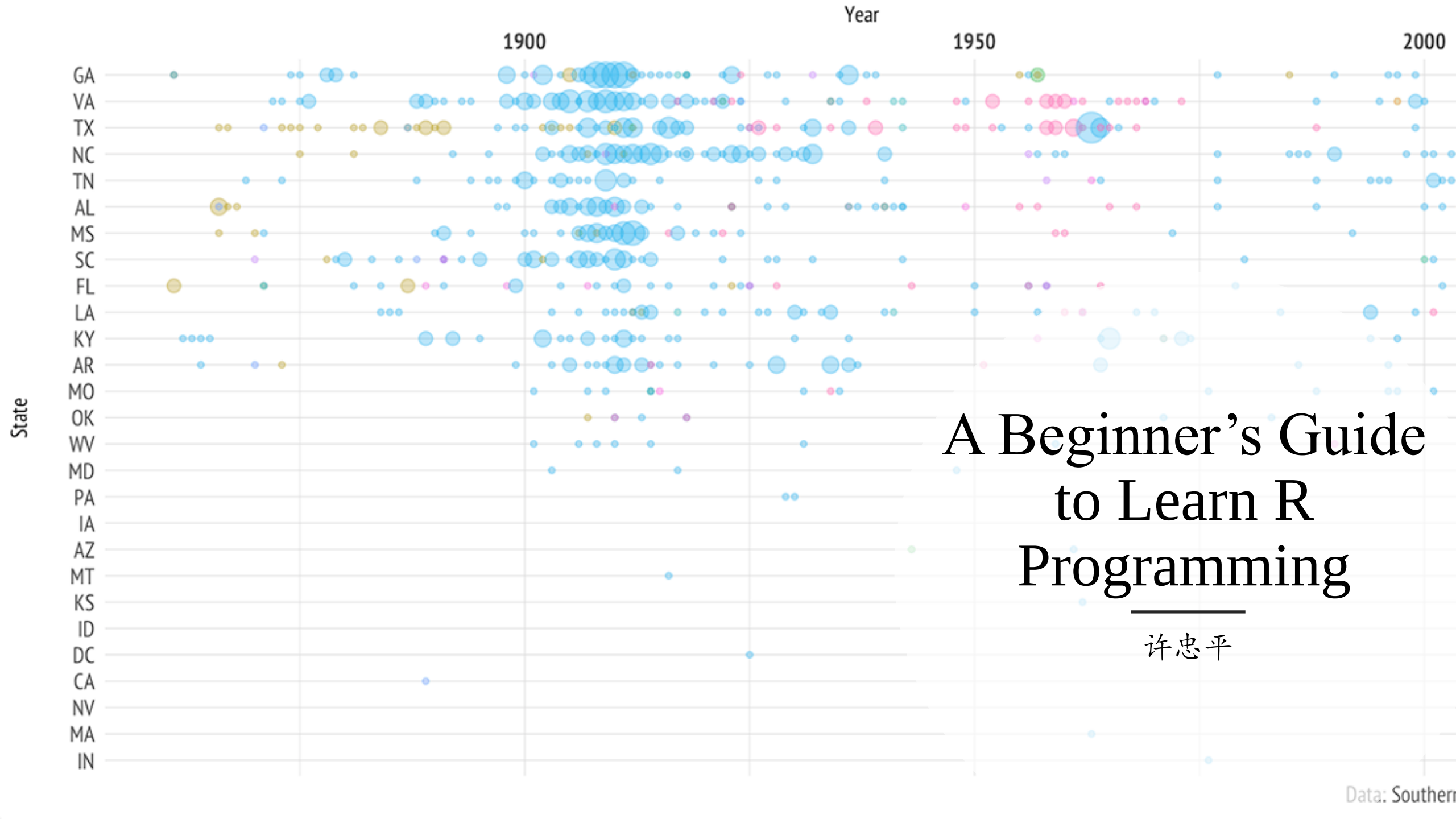
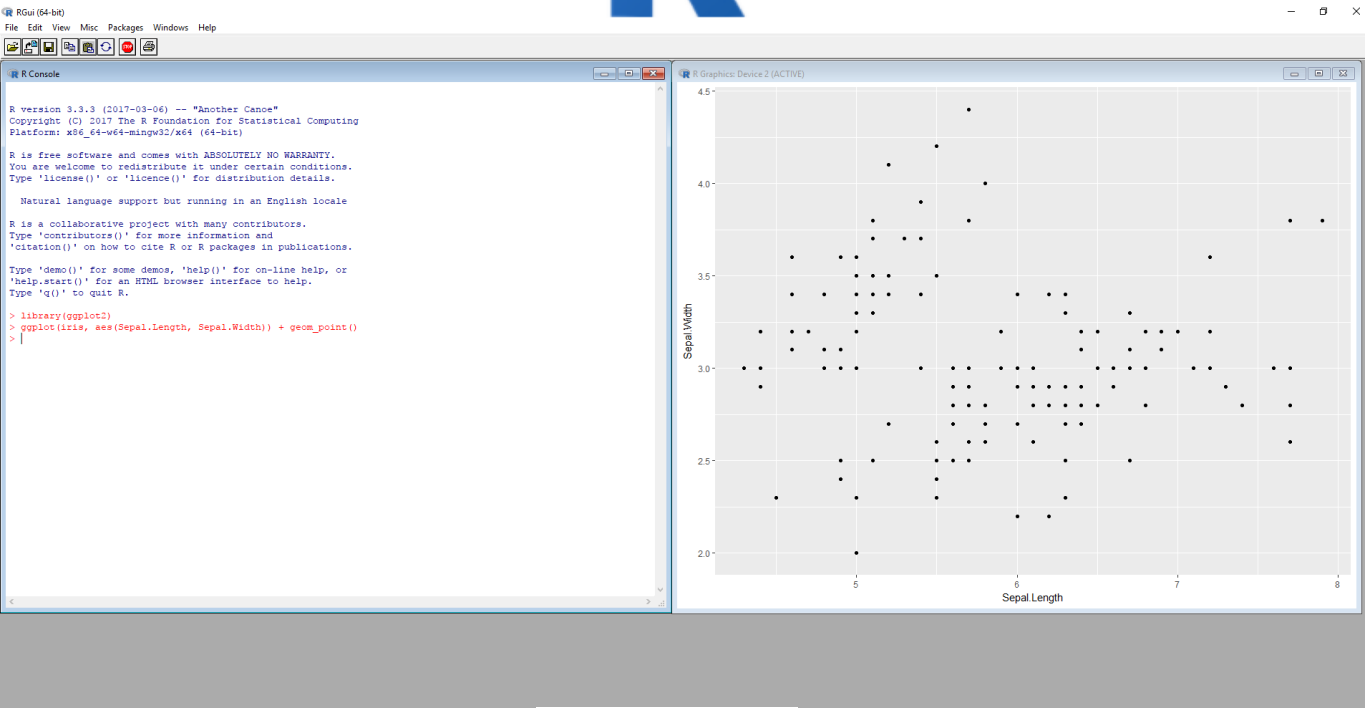
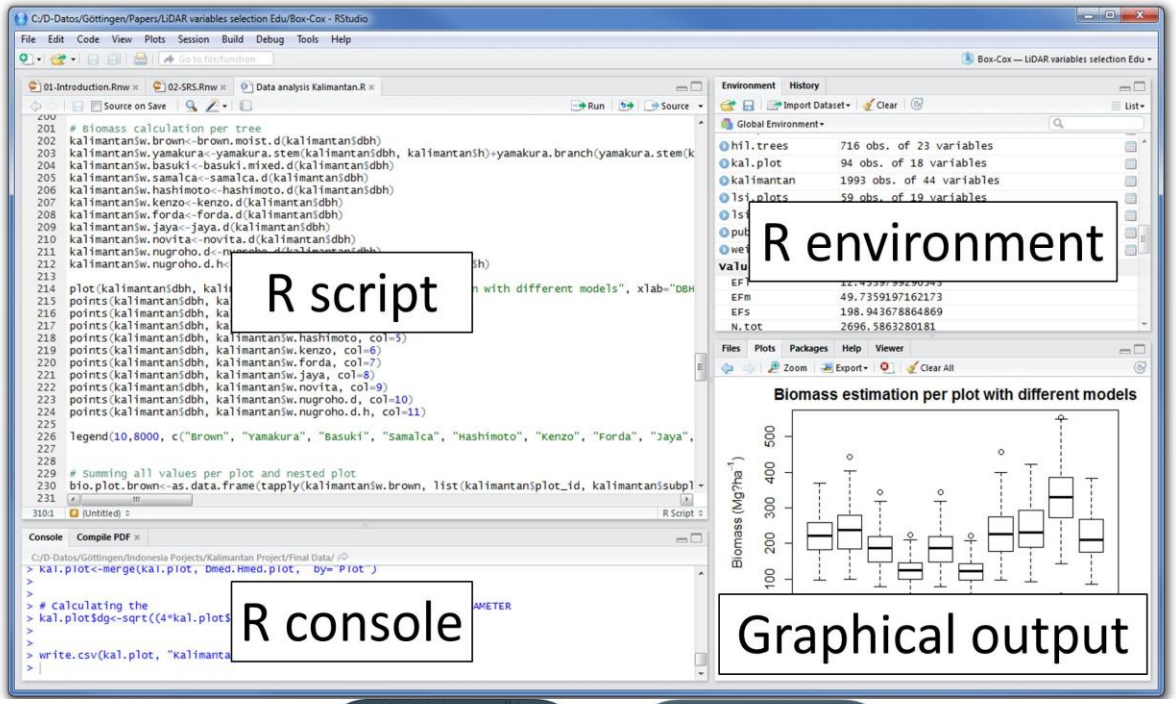


A Beginner's Guide to Learn R Programming

许忠平


RGui (64-bit)
File Edit View Misc Packages Windows Help
R Console
R version 3.3.3 (2017-03-06) -- "Another Canoe"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
Natural language support but running in an English locale
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
> library(ggplot2)
> qplot(iris, aes(Sepal.Length, Sepal.Width)) + geom_point()
R Graphics: Device 2 (ACTIVE)
Sepal.Width
Sepal.Length


R script
R environment
R console
Graphical output
Biomass estimation per plot with different models
Biomass (Mg/ha⁻¹)

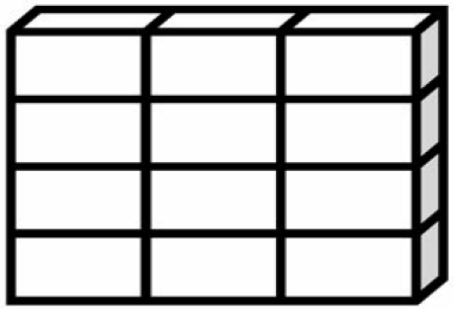

9:41
Tuesday, September 12

数据类型 (Data Types)

(a) 向量

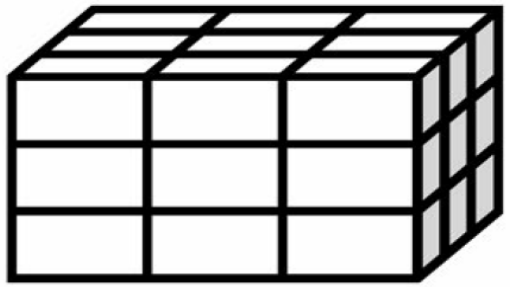
```
a <- c(1, 2, 5, 3, 6, -2, 4)
b <- c("one", "two", "three")
c <- c(TRUE, TRUE)
```

(b) 矩阵



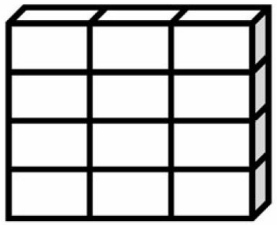
```
y <- matrix(1:12, nrow=3, ncol=4)
y
     [,1] [,2] [,3] [,4]
[1,] 1    4    7   10
[2,] 2    5    8   11
[3,] 3    6    9   12
```

(c) 数组



```
dim1 <- c("A1", "A2")
dim2 <- c("B1", "B2", "B3")
dim3 <- c("C1", "C2")
z <- array(1:12, c(2, 3, 2),
      dimnames=list(dim1, dim2, dim3))
z
, , C1
    B1  B2  B3
A1  1   3   5
A2  2   4   6
, , C2
    B1  B2  B3
A1  7   9   11
A2  8   10  12
```

(d) 数据框



各列的模式（modes）可以不同

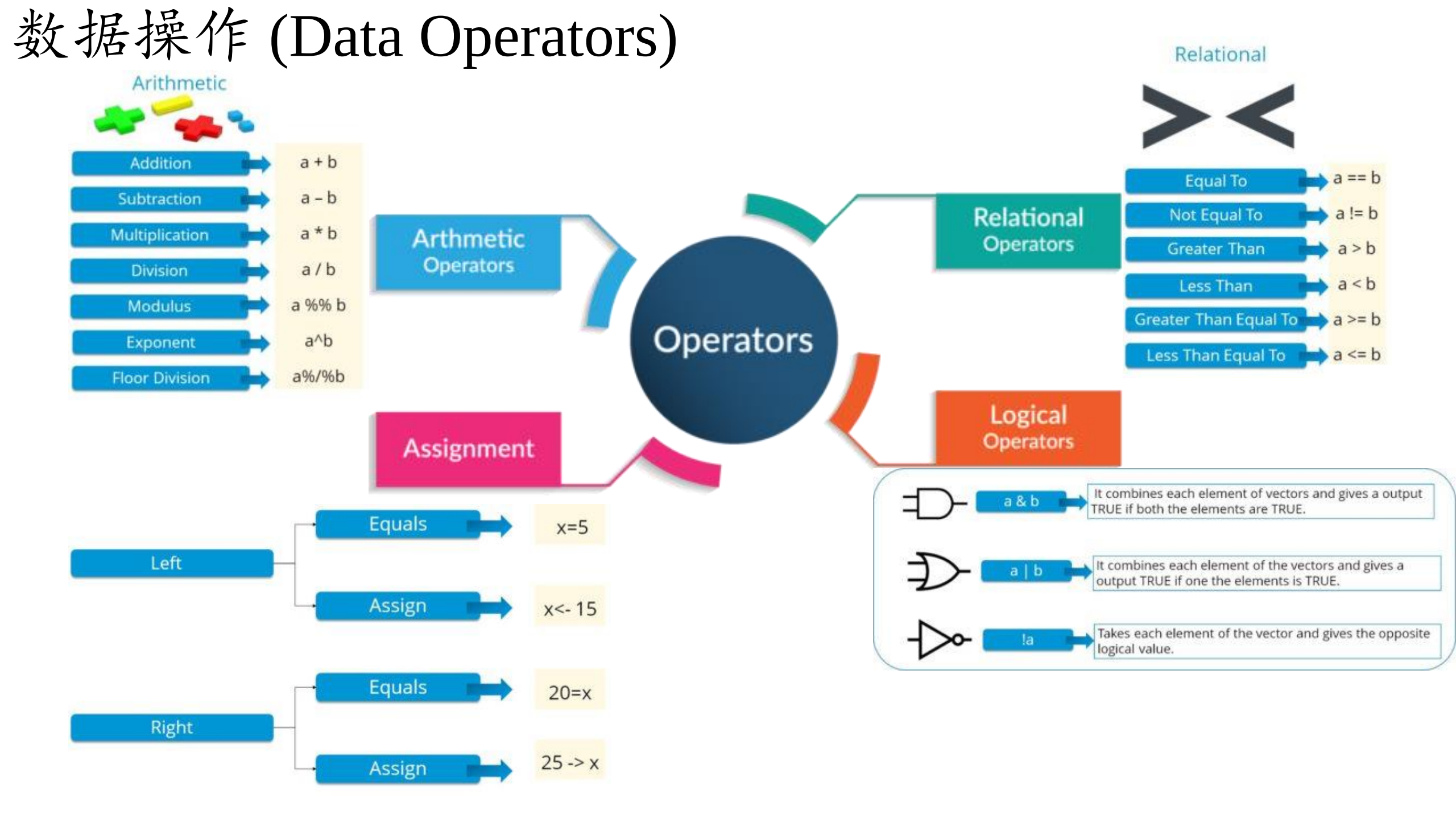
```
patientID <- c(1, 2, 3, 4)
age <- c(25, 34, 28, 52)
diabetes <- c("Type1", "Type2", "Type1", "Type1")
status <- c("Poor", "Improved", "Excellent", "Poor")
patientdata <- data.frame(patientID, age, diabetes, status)
patientdata
  patientID age diabetes status
1         1  25    Type1   Poor
2         2  34    Type2 Improved
3         3  28    Type1 Excellent
4         4  52    Type1   Poor
```

(e) 列表

- 向量
- 数组
- 数据框
- 列表

```
g <- “My First List”
h <- c(25, 26, 18, 39)
j <- matrix(1:4, nrow=2)
k <- c("one", "two", "three")
mylist <- list(title=g, ages=h, j, k)
mylist
$title
[1] "My First List"
$ages
[1] 25 26 18 39
[[3]]
     [,1] [,2]
[1,] 1    3
[2,] 2    4
[[4]]
[1] "one" "two" "three"
```

数据操作 (Data Operators)



函数(Functions)

求平均：mean(x)

求　和：sum(x)

开平方：sqrt(x)

……

function_name <–function(arg_1, arg_2, …) {

//Function body

}

```
> example("mean")

mean> x <- c(0:10, 50)

mean> xm <- mean(x)

mean> c(xm, mean(x, trim = 0.10))
[1] 8.75 5.50

```

```
sum_of_square <- function(x,y) {
x^2 + y^2
}
sum_of_sqares(3,4)

```

R 包 (R packages)

R packages are collections of functions and data sets developed by the community.

CRAN

install.packages(“XXX”)

Bioconductor

source("https://bioconductor.org/biocLite.R")
biocLite(“XXX")

Github

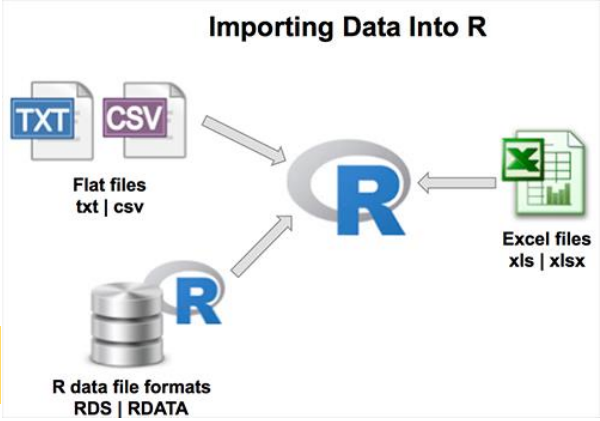
install.packages("devtools")
devtools::install_github("hadley/babynames")

library(XXX)

help(XXX)

search()

读取数据 (Import from files)



R base functions:

```
# Command to copy&paste tables from Excel or other programs into R.
read.delim("clipboard", header=T)
-----------------------------------------------------------------------------------------------
# Reads in table and assigns it to data frame, with info on column headers and field separators.
read.table(file="file", header=TRUE, sep="\t")
-----------------------------------------------------------------------------------------------
# Reads a file in table format and creates a data frame from
read.csv("file",header = T)

```

.txt

.csv

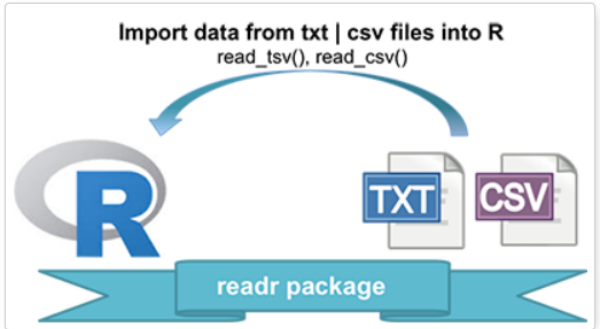
- file参数：这是必须的，可以是相对路径或者绝对路径（注意：Windows下路径要用斜杠'/'或者双反斜杠'\\'）。
- 如果数据集中含有中文，直接导入很有可能不识别中文，这时加上参fileEncoding='utf-8'

R package (readr):

```
install.packages("readr")
# Loading
library("readr")
read_csv(file, col_names = TRUE)
read_tsv(file, col_names = TRUE)

```

Compared to R base functions, readr functions are:

- much faster (X10),
- have a helpful progress bar if loading is going to take a while
- all functions work exactly the same way.



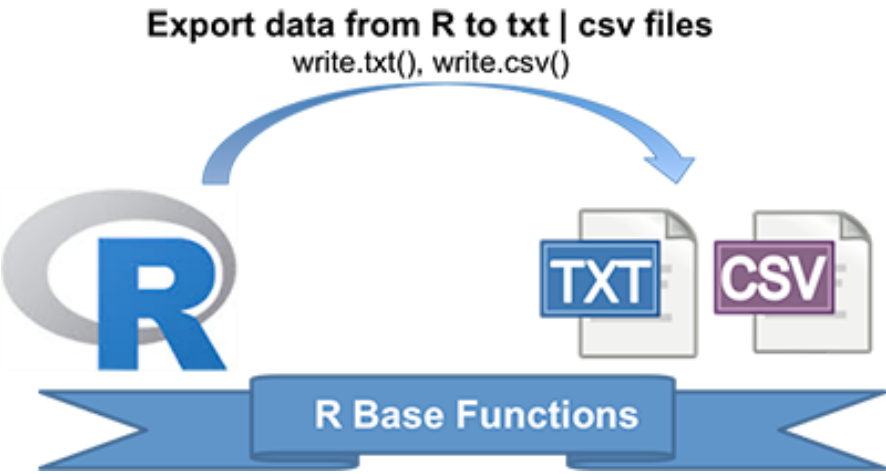
导出数据 (Export data from R)

R base functions:

```
data("mtcars")
write.table(mtcars, file = "mtcars.txt", sep = "\t",
            row.names = TRUE, col.names = NA)
-------------------------------------------------------
write.csv(mtcars, file = "mtcars.csv")
```

R package (readr):

```
write_csv(challenge, "challenge.csv")
```



数据转换 (Data transformation)

清洗和整理

```r
library(nycflights13)
library(tidyverse)
# 1.1 筛选: filter()
(jan1 <- filter(flights, month == 1, day == 1))
# 1.2 排列: arrange()
arrange(flights, year, month, day)
arrange(flights, desc(arr_delay))
# 1.3 选择: select()
select(flights, year, month, day)
# 1.4 变形: mutate()
flights_sml <- select(flights,
  year:day,
  ends_with("delay"),
  distance,
  air_time )
##### 新添加的列可以用于后续计算
mutate(flights_sml,
  gain = arr_delay - dep_delay,
  hours = air_time / 60,
  gain_per_hour = gain / hours )
##### 只保留变形后的列
transmute(flights,
  gain = arr_delay - dep_delay,
  hours = air_time / 60,
  gain_per_hour = gain / hours )
# 1.5 汇总: summarise()
summarise(flights, delay = mean(dep_delay, na.rm = TRUE))
# 1.6 分组: group_by()
by_day <- group_by(flights, year, month, day)
summarise(by_day, delay = mean(dep_delay, na.rm = TRUE))

```

筛选 按给定的逻辑判断筛选出符合要求的子数据集

排列 按给定的列名依次对行进行排序

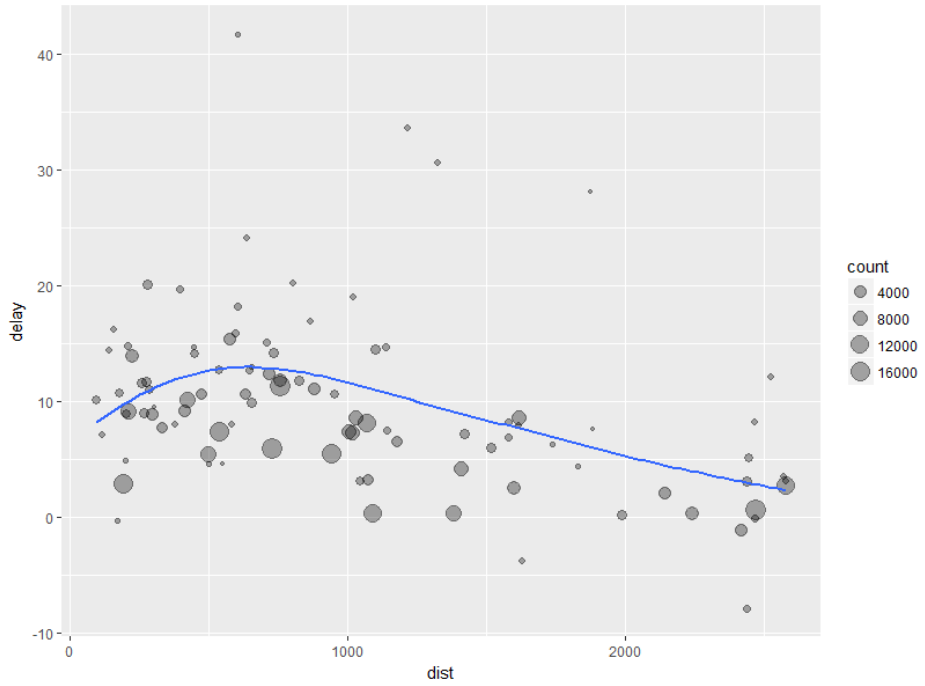
选择 用列名作参数来选择子数据集

变形 对已有列进行数据运算并添加为新列

汇总 对数据框调用其它函数进行汇总操作，返回一维的结果

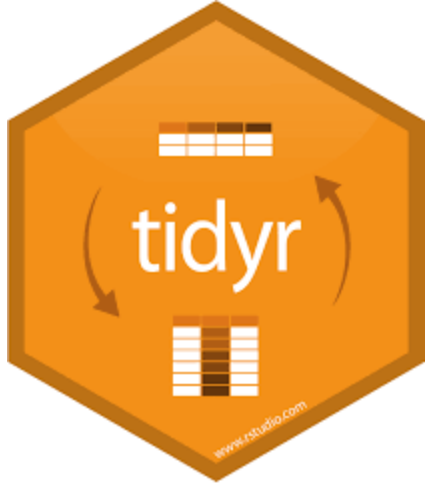
管道函数(%>%) 和 绘图

```
1  delays <- flights %>%
2    group_by(dest) %>%
3    summarise(
4      count = n(),
5      dist = mean(distance, na.rm = TRUE),
6      delay = mean(arr_delay, na.rm = TRUE)
7    ) %>%
8    filter(count > 20, dest != "HNL")
9  ggplot(data = delay, mapping = aes(x = dist, y = delay)) +
10   geom_point(aes(size = count), alpha = 1/3) +
11   geom_smooth(se = FALSE)
12
```



数据整形 (Reshaping Data)

```r
install.packages("tibble")
library("tibble")
# Create a new tibble use data_frame()
friends_data <- data_frame(
  name = c("Nicolas", "Thierry", "Bernard", "Jerome"),
  age = c(27, 25, 29, 26),
  height = c(180, 170, 185, 169),
  married = c(TRUE, FALSE, TRUE, TRUE)
)
friends_data
Source: local data frame [4 x 4]
     name   age height married

1 Nicolas    27    180    TRUE
2 Thierry    25    170   FALSE
3 Bernard    29    185    TRUE
4  Jerome    26    169    TRUE
# Convert your data as a tibble
data("iris")
class(iris)
[1] "data.frame"
my_data <- as_data_frame(iris)
class(my_data)
[1] "tbl_df"     "tbl"        "data.frame"
# Turn a tibble back to a data frame
my_data2 <- as.data.frame(my_data)

```



Tibble Data Format in R: Best and Modern Way to Work with Data

```
> flights
Source: local data frame [336,776 x 16]

    year month   day dep_time dep_delay arr_time arr_delay carrier tailnum
   <int> <int> <int>    <int>     <dbl>    <int>     <dbl>   <chr>   <chr>
1   2013     1     1      517         2      830        11      UA  N14228
2   2013     1     1      533         4      850        20      UA  N24211
3   2013     1     1      542         2      923        33      AA  N619AA
4   2013     1     1      54          4                -18      B6  N804JB
5   2013     1     1      55          2                -25      DL  N668DN
6   2013     1     1      55          0                 12      UA  N39463
7   2013     1     1      55          3                 19      B6  N516JB
8   2013     1     1      55          9                -14      EV  N829AS
9   2013     1     1      55          8                 -8      B6  N593JB
10  2013     1     1      558        -2      753         8      AA  N3ALAA
..   ...   ...   ...      ...       ...      ...       ...     ...     ...

Variables not shown: flight <int>, origin <chr>, dest <chr>, air_time
  <dbl>, distance <dbl>, hour <dbl>, minute <dbl>.
```

Column types

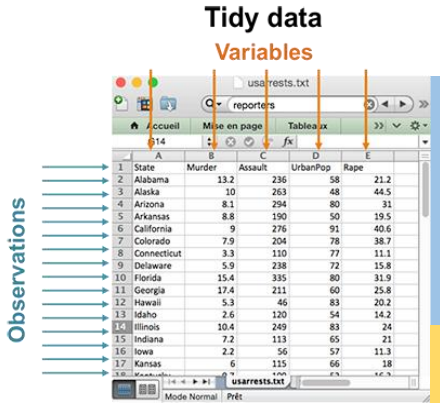
First 10 rows printed by default (Useful with large data sets)

Columns that don't fit to the screen are not shown

- **data_frame()**: create a new tibble data frame (tbl_df)
- **as_data_frame()**: convert your data as tbl_df

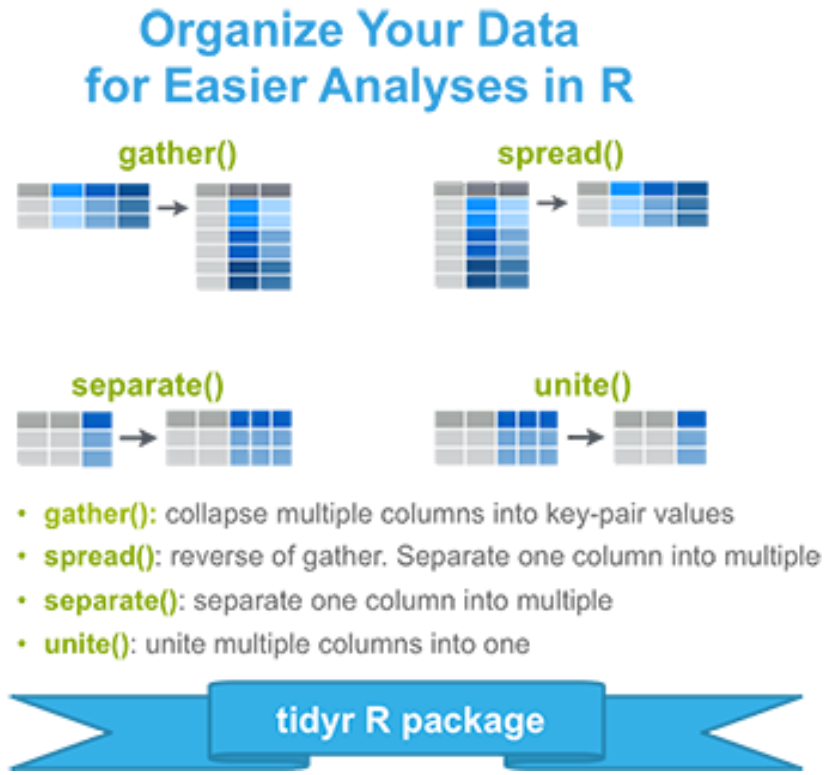
tibble R package

```
> library("tidyr")
> my_data <- USArrests[c(1, 10, 20, 30), ]
> my_data
           Murder Assault UrbanPop Rape
Alabama      13.2     236       58 21.2
Georgia      17.4     211       60 25.8
Maryland     11.3     300       67 27.8
New Jersey    7.4     159       89 18.8
> my_data <- cbind(state = rownames(my_data), my_data)
> my_data
                 state Murder Assault UrbanPop Rape
Alabama        Alabama   13.2     236       58 21.2
Georgia        Georgia   17.4     211       60 25.8
Maryland      Maryland   11.3     300       67 27.8
New Jersey New Jersey    7.4     159       89 18.8
```



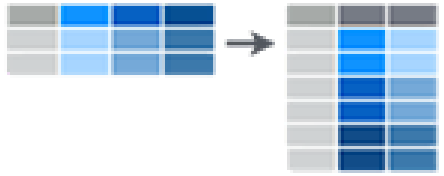
tidy data set:

- ➢ each column represents a variable
- ➢ each row represents an observation
- ❖ The opposite of tidy is messy data



- ✓ gather和spread函数将数据在长格式和宽格式之间相互转化，应用在比如稀疏矩阵和稠密矩阵之间的转化;
- ✓ separate和union方法提供了数据分组拆分、合并的功能，应用在nominal数据的转化上;

gather() 宽数据转为长数据



gather(data, key, value, ...)

- **data:** A data frame
- **key, value:** Names of key and value columns to create in output
- …: Specification of columns to gather. Allowed values are:
  - variable names
  - if you want to select all variables between a and e, use a:e
  - if you want to exclude a column name y use -y
  - for more options, see: dplyr::select()

```
my_data2 <- gather(my_data,
                   key = "arrest_attribute",
                   value = "arrest_estimate",
                   -state)
# Only Murder and Assault columns,
# remaining columns have been duplicated.
my_data2 <- gather(my_data,
                   key = "arrest_attribute",
                   value = "arrest_estimate",
                   Murder, Assault)

```

```
> my_data
                state Murder Assault UrbanPop Rape
Alabama       Alabama   13.2     236       58 21.2
Georgia       Georgia   17.4     211       60 25.8
Maryland     Maryland   11.3     300       67 27.8
New Jersey New Jersey    7.4     159       89 18.8
-------------------------------------------------
> my_data2
         state arrest_attribute arrest_estimate
1      Alabama           Murder            13.2
2      Georgia           Murder            17.4
3     Maryland           Murder            11.3
4   New Jersey           Murder             7.4
5      Alabama          Assault           236.0
6      Georgia          Assault           211.0
7     Maryland          Assault           300.0
8   New Jersey          Assault           159.0
9      Alabama         UrbanPop            58.0
10     Georgia         UrbanPop            60.0
11    Maryland         UrbanPop            67.0
12 New Jersey          UrbanPop            89.0
13     Alabama             Rape            21.2
14     Georgia             Rape            25.8
15    Maryland             Rape            27.8
16 New Jersey              Rape            18.8

```

spread() 长数据转为宽数据

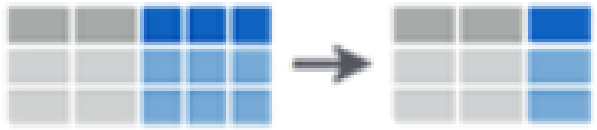
spread(data, key, value)

- **data:** A data frame
- **key:** The (unquoted) name of the column whose values will be used as column headings.
- **value:**The (unquoted) names of the column whose values will populate the cells.

```
my_data3 <- spread(my_data2,
                   key = "arrest_attribute",
                   value = "arrest_estimate"
                   )

```

```
> my_data
               state Murder Assault UrbanPop Rape
Alabama      Alabama   13.2     236       58 21.2
Georgia      Georgia   17.4     211       60 25.8
Maryland    Maryland   11.3     300       67 27.8
New Jersey New Jersey    7.4     159       89 18.8
-------------------------------------------------
> my_data2
         state arrest_attribute arrest_estimate
1      Alabama           Murder            13.2
2      Georgia           Murder            17.4
3     Maryland           Murder            11.3
4   New Jersey           Murder             7.4
5      Alabama          Assault           236.0
6      Georgia          Assault           211.0
7     Maryland          Assault           300.0
8   New Jersey          Assault           159.0
9      Alabama         UrbanPop            58.0
10     Georgia         UrbanPop            60.0
11    Maryland         UrbanPop            67.0
12  New Jersey         UrbanPop            89.0
13     Alabama             Rape            21.2
14     Georgia             Rape            25.8
15    Maryland             Rape            27.8
16  New Jersey             Rape            18.8
-------------------------------------------------
> my_data3
        state Assault Murder Rape UrbanPop
1     Alabama     236   13.2 21.2       58
2     Georgia     211   17.4 25.8       60
3    Maryland     300   11.3 27.8       67
4  New Jersey     159    7.4 18.8       89

```

unite() 多列合并为一列



unite(data, col, ..., sep = "_")

- ✓ **data**: A data frame
- ✓ **col**: The new (unquoted) name of column to add.
- ✓ **sep**: Separator to use between values

```
> my_data
            Murder Assault UrbanPop Rape
Alabama       13.2     236       58 21.2
Georgia       17.4     211       60 25.8
Maryland      11.3     300       67 27.8
New Jersey     7.4     159       89 18.8
> my_data4
            Murder_Assault UrbanPop Rape
Alabama           13.2_236       58 21.2
Georgia           17.4_211       60 25.8
Maryland          11.3_300       67 27.8
New Jersey         7.4_159       89 18.8

```

```
my_data4 <- unite(my_data,
                  col = "Murder_Assault",
                  Murder, Assault,
                  sep = "_")

```

separate() 将一列分离为多列

separate(data, col, into, sep = "[^[:alnum:]]+")

- ✓ **data:** A data frame
- ✓ **col:** Unquoted column names
- ✓ **into:** Character vector specifying the names of new variables to be created.
- ✓ **sep:** Separator between columns:
  - ✓ If character, is interpreted as a regular expression.
  - ✓ If numeric, interpreted as positions to split at. Positive values start at 1 at the far-left of the string; negative value start at -1 at the far-right of the string.

```
my_data5 <- separate(my_data4,
                     col = "Murder_Assault",
                     into = c("Murder", "Assault"),
                     sep = "_")

```

```
> my_data
           Murder Assault UrbanPop Rape
Alabama      13.2     236       58 21.2
Georgia      17.4     211       60 25.8
Maryland     11.3     300       67 27.8
New Jersey    7.4     159       89 18.8
> my_data4
           Murder_Assault UrbanPop Rape
Alabama           13.2_236       58 21.2
Georgia           17.4_211       60 25.8
Maryland          11.3_300       67 27.8
New Jersey         7.4_159       89 18.8
> my_data5
           Murder Assault UrbanPop Rape
Alabama      13.2     236       58 21.2
Georgia      17.4     211       60 25.8
Maryland     11.3     300       67 27.8
New Jersey    7.4     159       89 18.8

```

管道函数(%>%)

gather() + unite()

```
my_data6 <- my_data %>% gather(key = "arrest_attribute",
                    value = "arrest_estimate",
                    Murder:UrbanPop) %>%
  unite(col = "attribute_estimate",
        arrest_attribute, arrest_estimate)
------------------------------------------------------------
> my_data
           Murder Assault UrbanPop Rape
Alabama      13.2     236       58 21.2
Georgia      17.4     211       60 25.8
Maryland     11.3     300       67 27.8
New Jersey    7.4     159       89 18.8
> my_data6
   Rape attribute_estimate
1  21.2        Murder_13.2
2  25.8        Murder_17.4
3  27.8        Murder_11.3
4  18.8         Murder_7.4
5  21.2        Assault_236
6  25.8        Assault_211
7  27.8        Assault_300
8  18.8        Assault_159
9  21.2        UrbanPop_58
10 25.8        UrbanPop_60
11 27.8        UrbanPop_67
12 18.8        UrbanPop_89
```

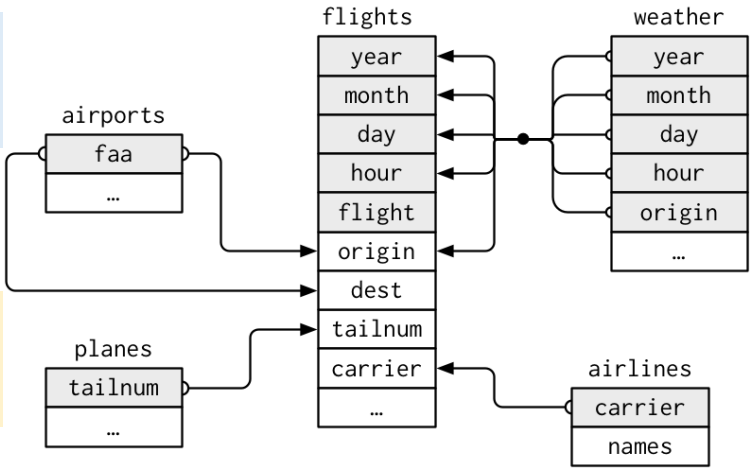
关系型数据 (Relational data)

```
library(nycflights13)
> class(flights)
[1] "tbl_df"     "tbl"        "data.frame"
> flights
# A tibble: 336,776 x 19
    year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time arr_delay carrier flight tailnum origin  dest
   <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>     <dbl>   <chr>  <int>   <chr>  <chr> <chr>
 1  2013     1     1      517            515         2      830            819        11      UA   1545  N14228    EWR   IAH
 2  2013     1     1      533            529         4      850            830        20      UA   1714  N24211    LGA   IAH
 3  2013     1     1      542            540         2      923            850        33      AA   1141  N619AA    JFK   MIA
 4  2013     1     1      544            545        -1     1004           1022       -18      B6    725  N804JB    JFK   BQN
 5  2013     1     1      554            600        -6      812            837       -25      DL    461  N668DN    LGA   ATL
 6  2013     1     1      554            558        -4      740            728        12      UA   1696  N39463    EWR   ORD
 7  2013     1     1      555            600        -5      913            854        19      B6    507  N516JB    EWR   FLL
 8  2013     1     1      557            600        -3      709            723       -14      EV   5708  N829AS    LGA   IAD
 9  2013     1     1      557            600        -3      838            846        -8      B6     79  N593JB    JFK   MCO
10  2013     1     1      558            600        -2      753            745         8      AA    301  N3ALAA    LGA   ORD
# ... with 336,766 more rows, and 5 more variables: air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>, time_hour <dttm>
planes %>% count(tailnum) %>% filter(n > 1) # verify the key

```

key

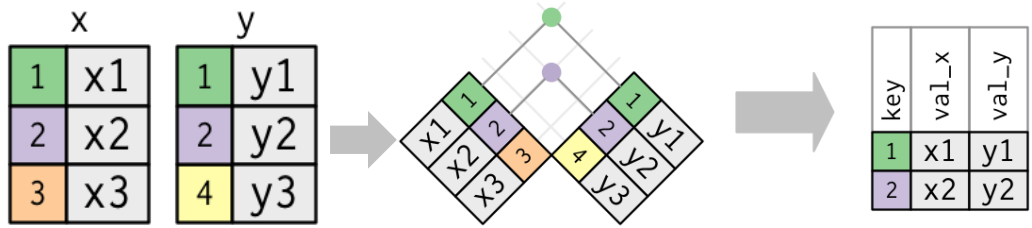
primary key: uniquely identifies an observation in its own table.
planes$tailnum

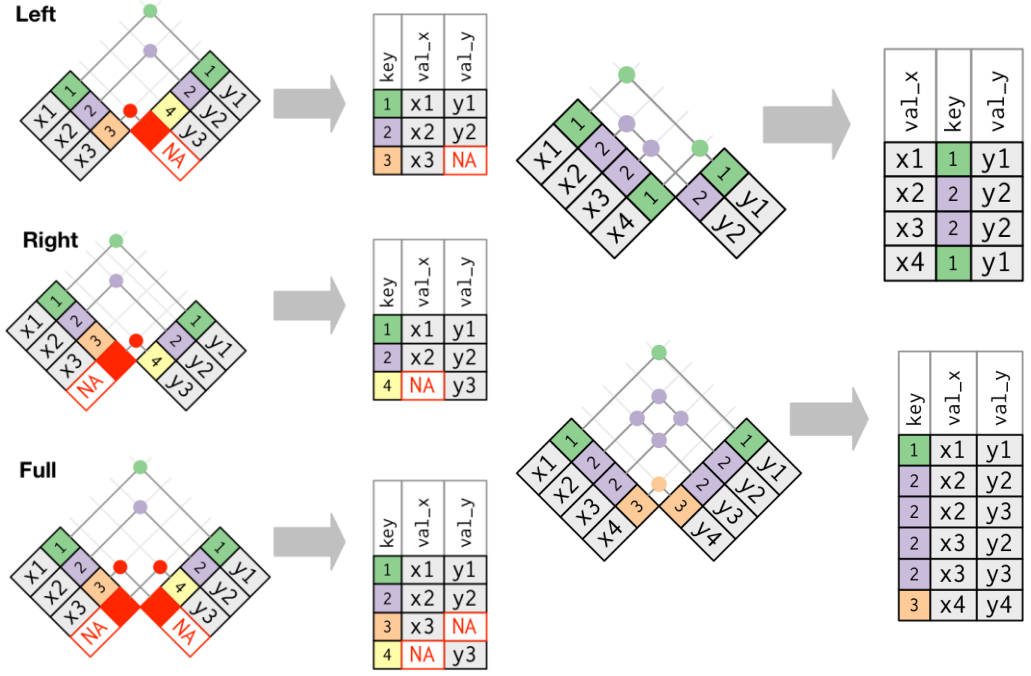
foreign key: uniquely identifies an observation in another table.
flights$tailnum

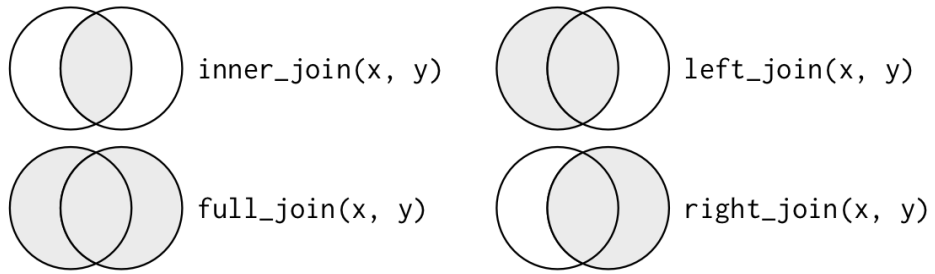


Mutating joins

```
> flights2 <- flights %>% select(year:day, hour, origin, dest, tailnum, carrier)
> flights2
# A tibble: 336,776 x 8
    year month   day  hour origin  dest tailnum carrier
   <int> <int> <int> <dbl>  <chr> <chr>   <chr>   <chr>
 1  2013     1     1     5    EWR   IAH  N14228      UA
 2  2013     1     1     5    LGA   IAH  N24211      UA
 3  2013     1     1     5    JFK   MIA  N619AA      AA
 4  2013     1     1     5    JFK   BQN  N804JB      B6
 5  2013     1     1     6    LGA   ATL  N668DN      DL
 6  2013     1     1     5    EWR   ORD  N39463      UA
 7  2013     1     1     6    EWR   FLL  N516JB      B6
 8  2013     1     1     6    LGA   IAD  N829AS      EV
 9  2013     1     1     6    JFK   MCO  N593JB      B6
10  2013     1     1     6    LGA   ORD  N3ALAA      AA
# ... with 336,766 more rows
> flights2 %>% select(-origin, -dest) %>% left_join(airlines, by = "carrier")
# A tibble: 336,776 x 7
    year month   day  hour tailnum carrier                    name
   <int> <int> <int> <dbl>   <chr>   <chr>                   <chr>
 1  2013     1     1     5  N14228      UA    United Air Lines Inc.
 2  2013     1     1     5  N24211      UA    United Air Lines Inc.
 3  2013     1     1     5  N619AA      AA   American Airlines Inc.
 4  2013     1     1     5  N804JB      B6          JetBlue Airways
 5  2013     1     1     6  N668DN      DL     Delta Air Lines Inc.
 6  2013     1     1     5  N39463      UA    United Air Lines Inc.
 7  2013     1     1     6  N516JB      B6          JetBlue Airways
 8  2013     1     1     6  N829AS      EV ExpressJet Airlines Inc.
 9  2013     1     1     6  N593JB      B6          JetBlue Airways
10  2013     1     1     6  N3ALAA      AA   American Airlines Inc.
# ... with 336,766 more rows
```



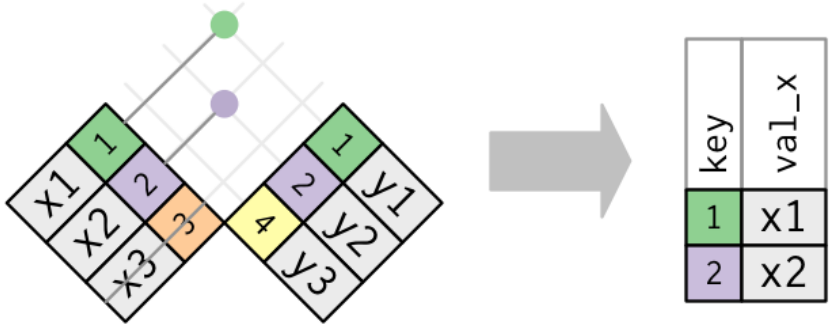




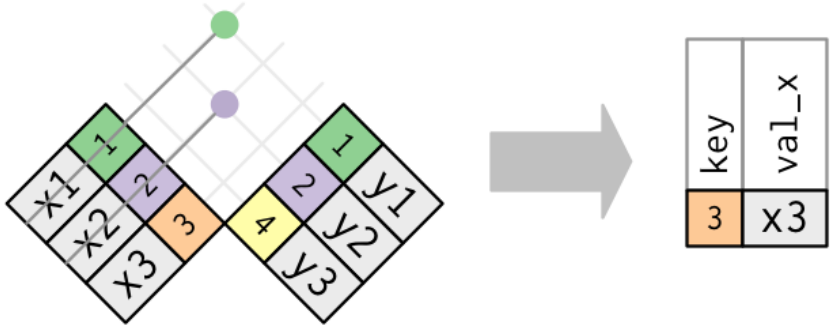
Filtering joins

```
> top_dest <- flights %>% count(dest, sort = TRUE) %>% head(10)
> flights %>% semi_join(top_dest)
> dim(flights)
[1] 336776     19
> dim(flights %>% semi_join(top_dest))
[1] 80262    19
> flights %>% anti_join(planes, by = "tailnum") %>% count(tailnum, sort = TRUE)
# A tibble: 722 x 2
   tailnum     n
     <chr> <int>
 1    <NA>  2512
 2  N725MQ   575
 3  N722MQ   513
 4  N723MQ   507
 5  N713MQ   483
 6  N735MQ   396
# ... with 712 more rows
```

semi_join(x, y): keeps all observations in x that have a match in y



anti_join(x, y): drops all observations in x that have a match in y



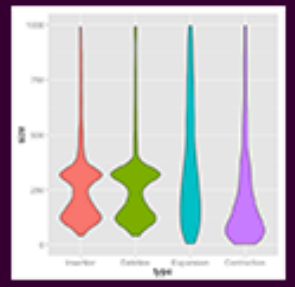
Set operations

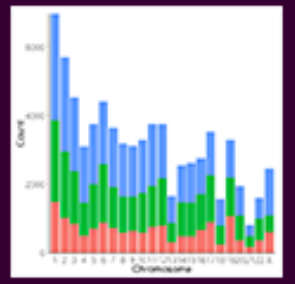
```
df1 <- tribble(
  ~x, ~y,
  1,  1,
  2,  1
)
df2 <- tribble(
  ~x, ~y,
  1,  1,
  1,  2
)
> df1
# A tibble: 2 x 2
      x     y
  <dbl> <dbl>
1     1     1
2     2     1
> df2
# A tibble: 2 x 2
      x     y
  <dbl> <dbl>
1     1     1
2     1     2

```

- ✓ **intersect(x, y):** return only observations in both x and y.
- ✓ **union(x, y):** return unique observations in x and y.
- ✓ **setdiff(x, y):** return observations in x, but not in y.

```
> intersect(df1, df2)
# A tibble: 1 x 2
      x     y
  <dbl> <dbl>
1     1     1
----------------------
> union(df1, df2)
# A tibble: 3 x 2
      x     y
  <dbl> <dbl>
1     1     2
2     2     1
3     1     1
----------------------
> setdiff(df1, df2)
# A tibble: 1 x 2
      x     y
  <dbl> <dbl>
1     2     1

```

Plotting in R for Biologists



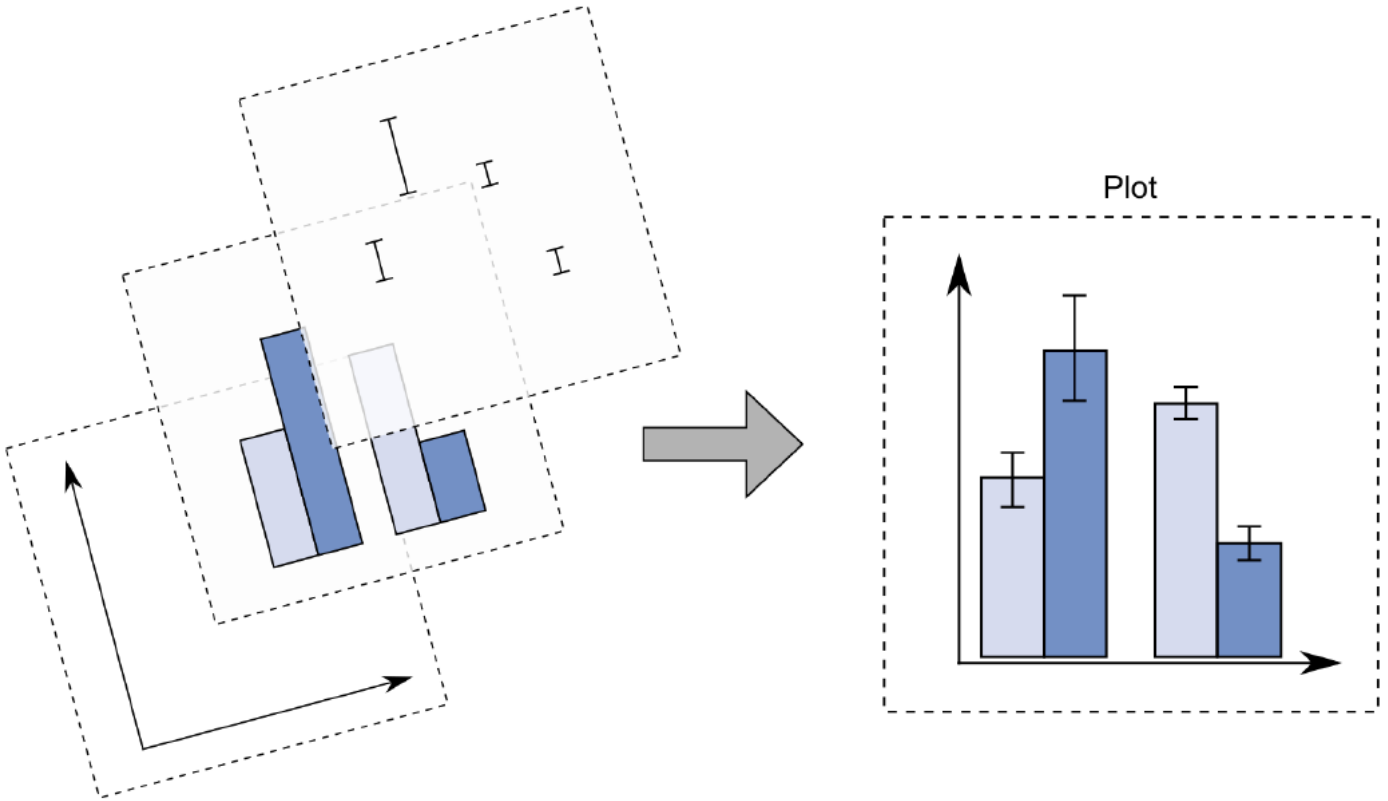


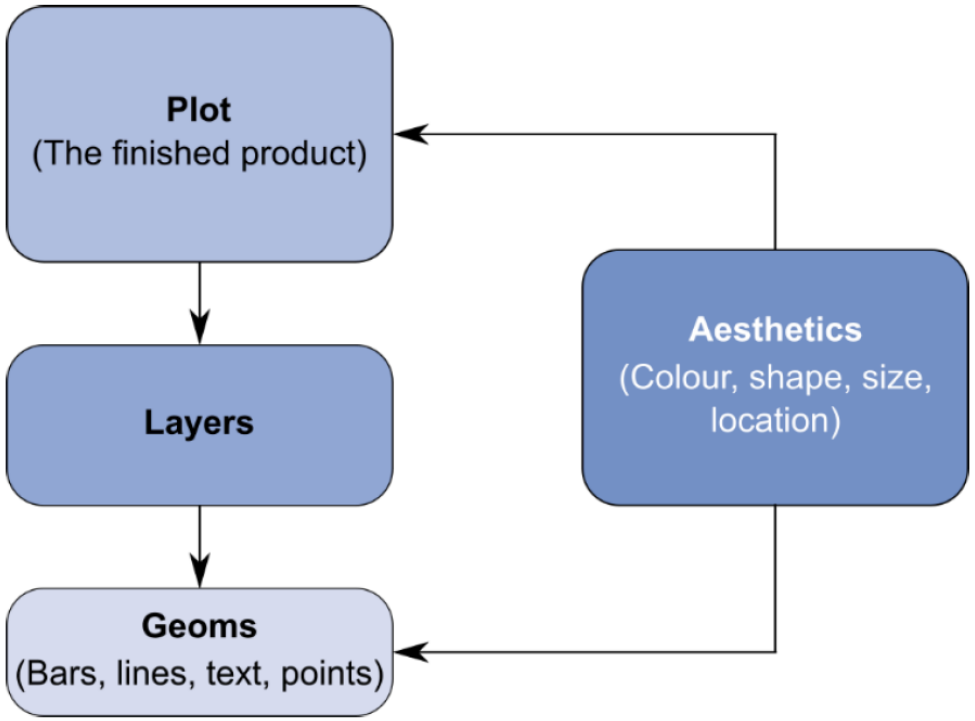
ggplot2

http://hadley.nz/

http://ggplot2.org/

图层 (layers)


Plot


Plot
(The finished product)
Layers
Geoms
(Bars, lines, text, points)
Aesthetics
(Colour, shape, size, location)

ggplot2的基本概念

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

Data

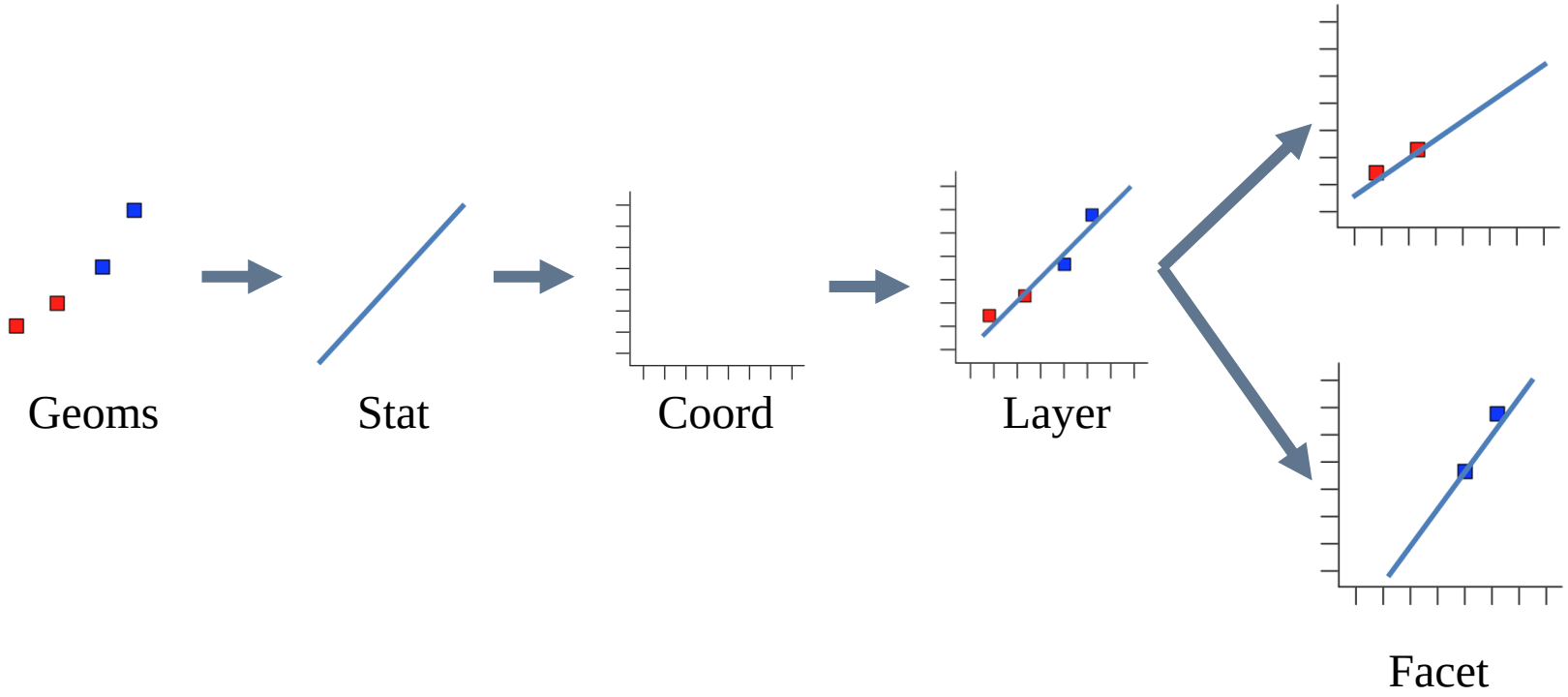
length	width	depth	trt
2	3	4	a
1	2	1	a
4	5	15	b
9	10	80	b

Mapping

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

Scale

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



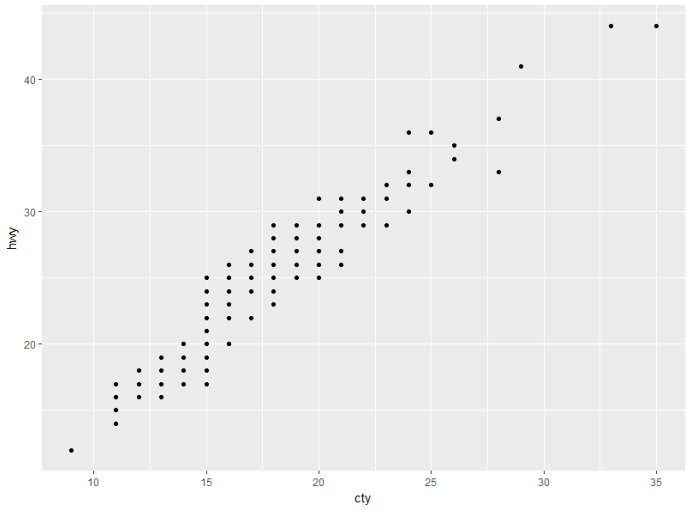
实例数据

```
> library(ggplot2)
> str(mpg)
Classes 'tbl_df', 'tbl' and 'data.frame':    234 obs. of  11 variables:
 $ manufacturer: chr  "audi" "audi" "audi" "audi" ...
 $ model       : chr  "a4" "a4" "a4" "a4" ...
 $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
 $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
 $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
 $ trans       : chr  "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
 $ drv         : chr  "f" "f" "f" "f" ...
 $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
 $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
 $ fl          : chr  "p" "p" "p" "p" ...
 $ class       : chr  "compact" "compact" "compact" "compact" ...

> head(mpg)
# A tibble: 6 x 11
  manufacturer model displ  year   cyl      trans   drv   cty   hwy    fl   class
         <chr> <chr> <dbl> <int> <int>      <chr> <chr> <int> <int> <chr>   <chr>
1         audi    a4   1.8  1999     4   auto(l5)     f    18    29     p compact
2         audi    a4   1.8  1999     4 manual(m5)     f    21    29     p compact
3         audi    a4   2.0  2008     4 manual(m6)     f    20    31     p compact
4         audi    a4   2.0  2008     4   auto(av)     f    21    30     p compact
5         audi    a4   2.8  1999     6   auto(l5)     f    16    26     p compact
6         audi    a4   2.8  1999     6 manual(m5)     f    18    26     p compact
```

散点图

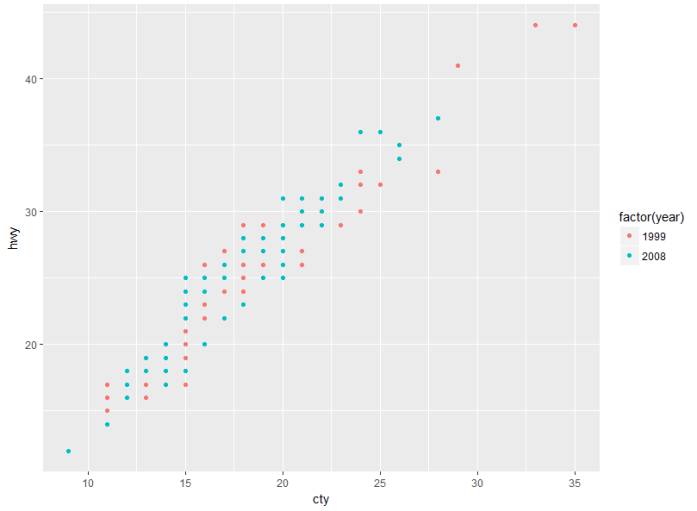
aesthetics

```
p <- ggplot(data=mpg, mapping=aes(x=cty, y=hwy))
p + geom_point()
```



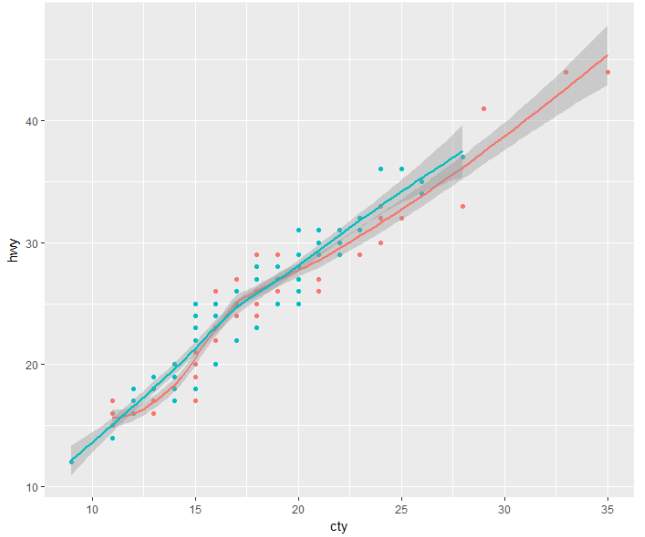
将年份映射到颜色属性 (Mapping):

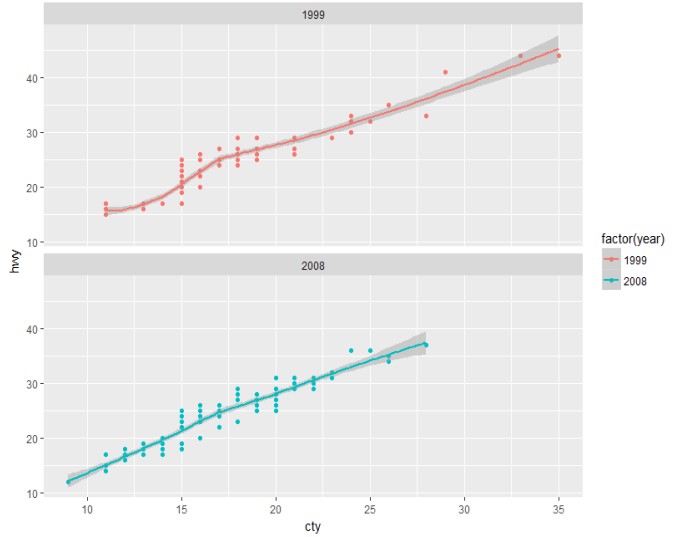
```
p <- ggplot(mpg,aes(x=cty, y=hwy, colour=factor(year)))
p + geom_point()
```



增加平滑曲线 (Statistics):

```
p + geom_point() + stat_smooth()
> summary(p + geom_point() + stat_smooth())
geom_smooth: na.rm = FALSE
stat_smooth: method = auto, formula = y ~ x,
             se = TRUE, n = 80, fullrange = FALSE,
             level = 0.95, na.rm = FALSE,
             method.args = list(), span = 0.75
position_identity
```



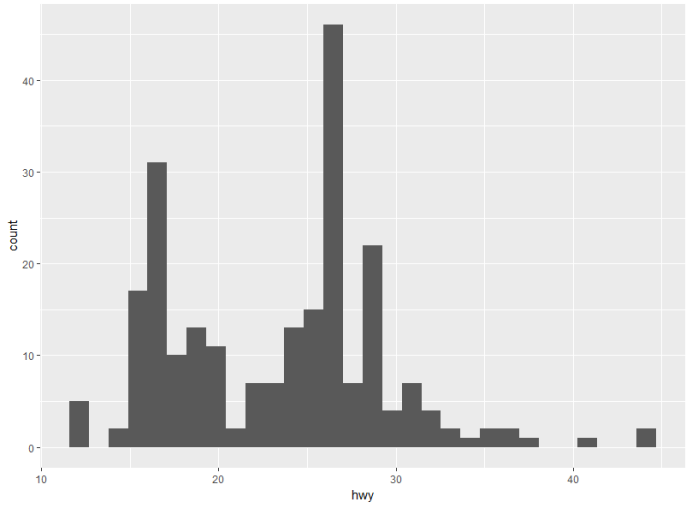


分面 (Facet):

```
p + geom_point() + stat_smooth()+facet_wrap(~ year, ncol=1)
```

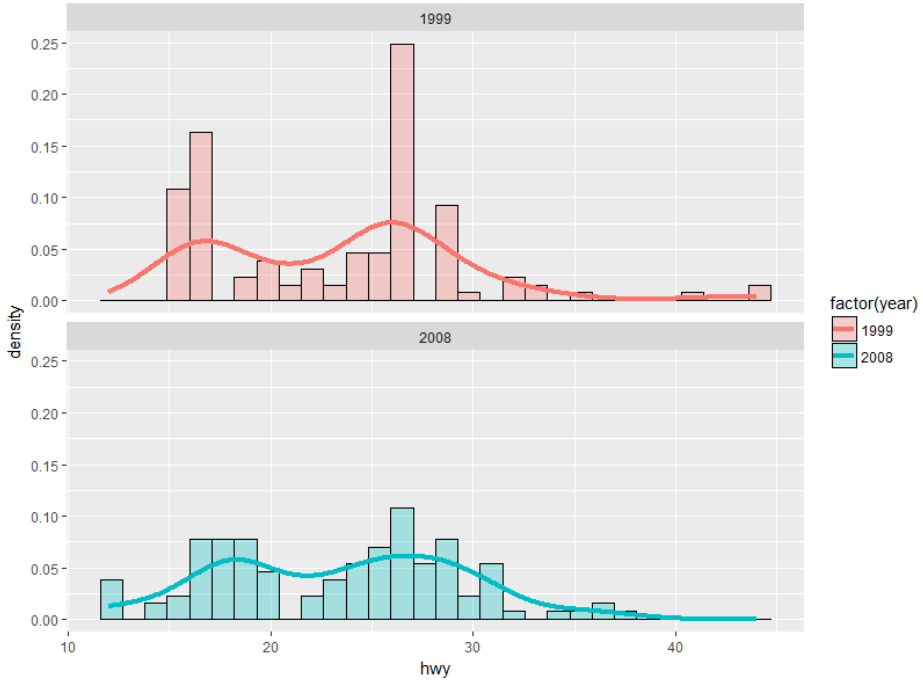
直方图

```
1   p <- ggplot(mpg,aes(x=hwy))
2   p + geom_histogram()
3
```



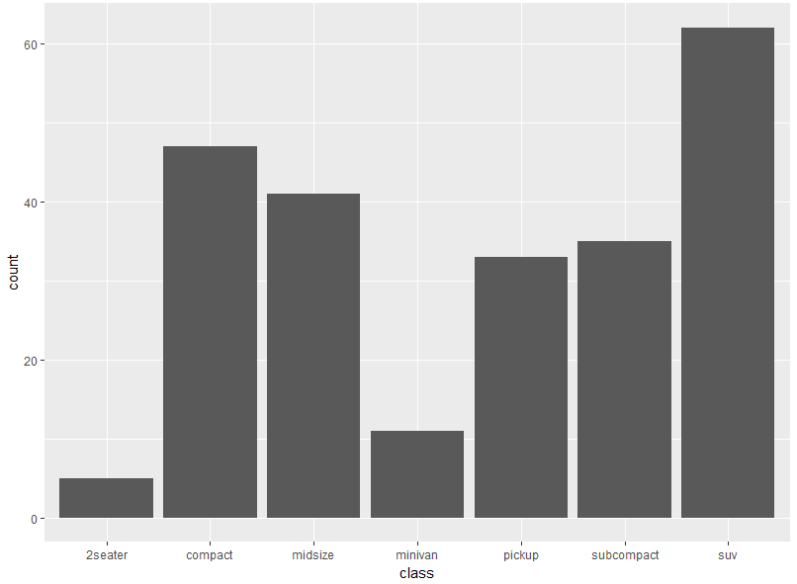
统计变换+分面：

```
1   # Histogram with density plot
2   p + geom_histogram(aes(fill=factor(year),y=..density..), alpha=0.3,colour='black') +
3       stat_density(geom='line',position='identity',size=1.5, aes(colour=factor(year))) +
4       facet_wrap(~year,ncol=1)
5
```



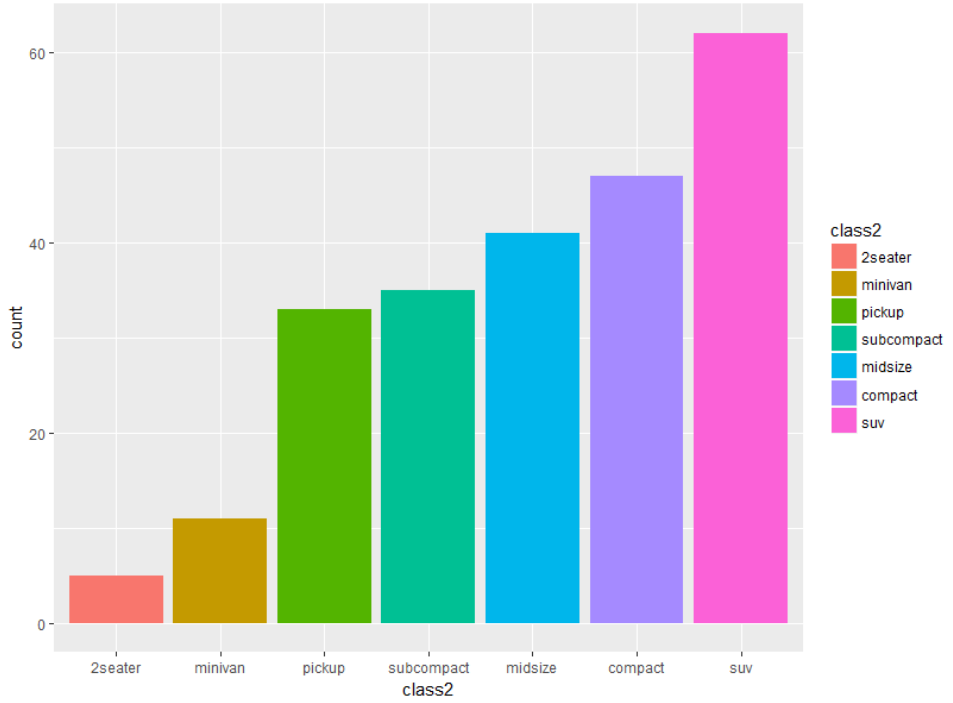
条形图

```
1   p <- ggplot(mpg, aes(x=class))
2   p + geom_bar()
3
```



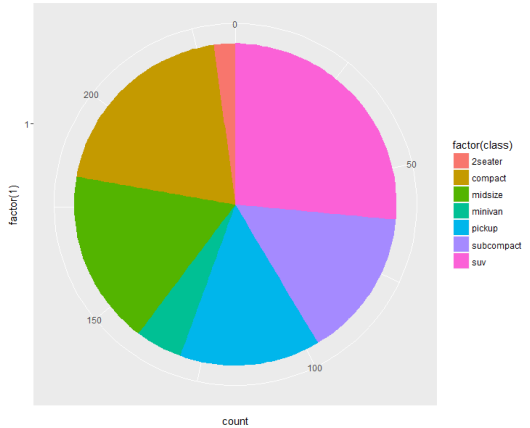
根据计数排序后绘制的条形图：

```
1    class2 <- mpg$class
2    class2 <- reorder(class2,class2,length)
3    mpg$class2 <- class2
4    p <- ggplot(mpg, aes(x=class2))
5    p + geom_bar(aes(fill=class2))
6    ----------------------------------------
7    help(reorder)
8
9    Reorder Levels of a Factor
10   Usage
11   reorder(x, X, FUN = mean, ...,
12           order = is.ordered(x))
13   ----------------------------------------
14   class(class2)
15   [1] "factor"
16
```



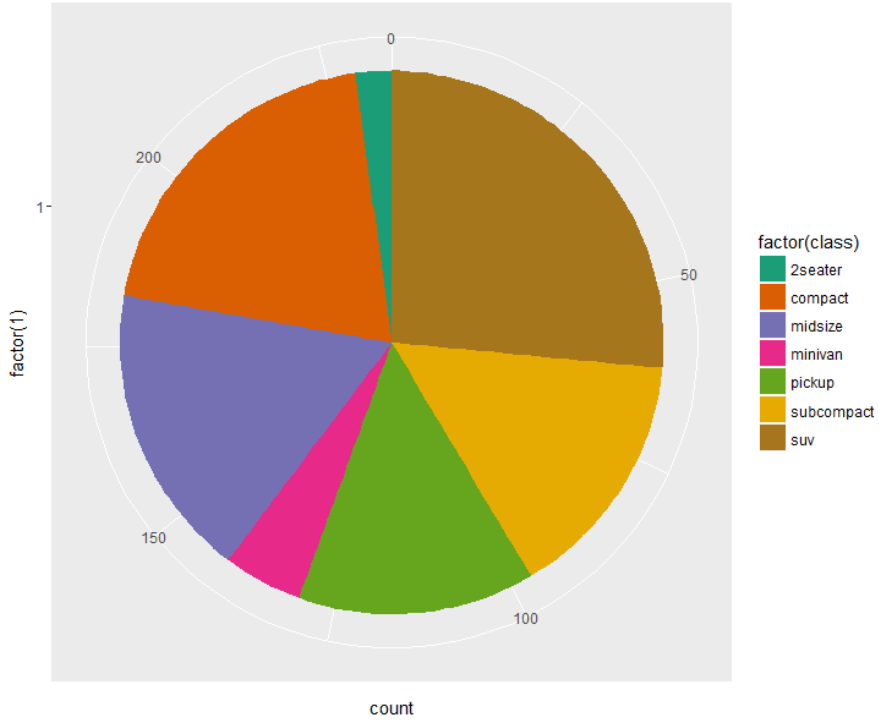
饼图

```
p <- ggplot(mpg, aes(x = factor(1), fill = factor(class))) +
     geom_bar(width = 1)
p


p + coord_polar(theta = "y")

```



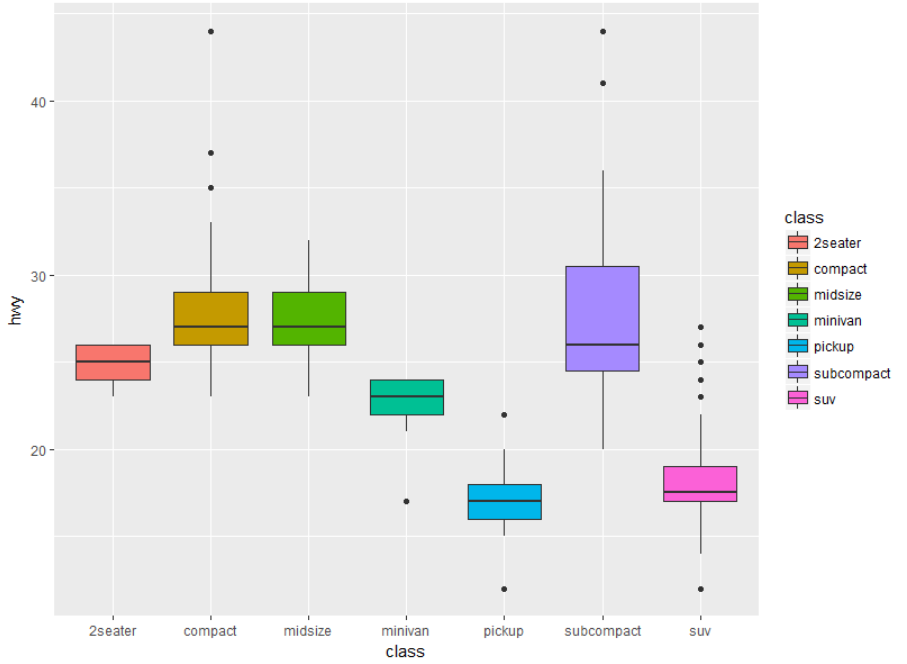
改变填充颜色:

```
p + coord_polar(theta = "y") +
     scale_fill_brewer(palette="Dark2")

```



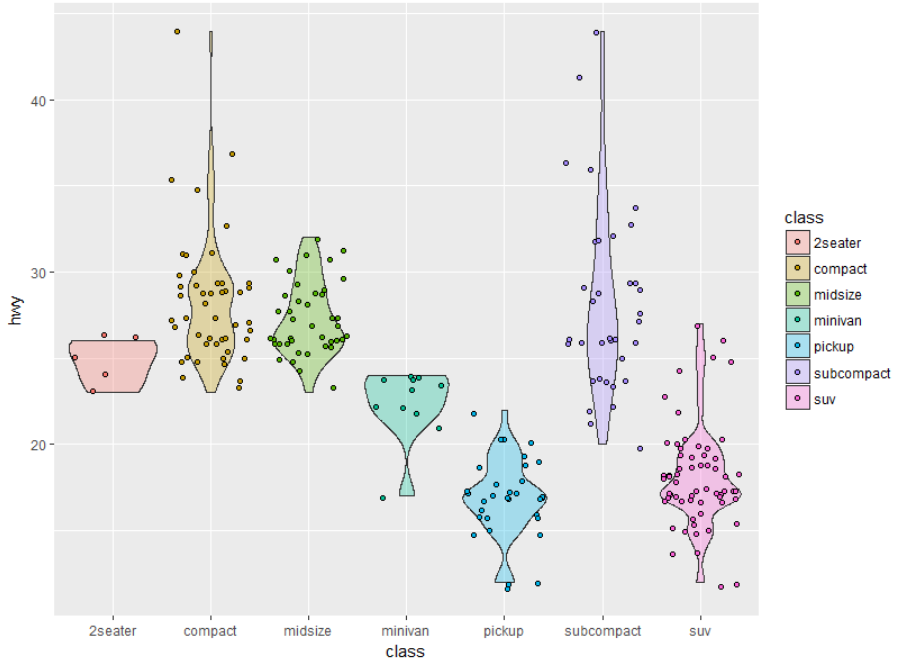
箱线图

```
1  p <- ggplot(mpg, aes(class,hwy,fill=class))
2  p +  geom_boxplot()
3
```



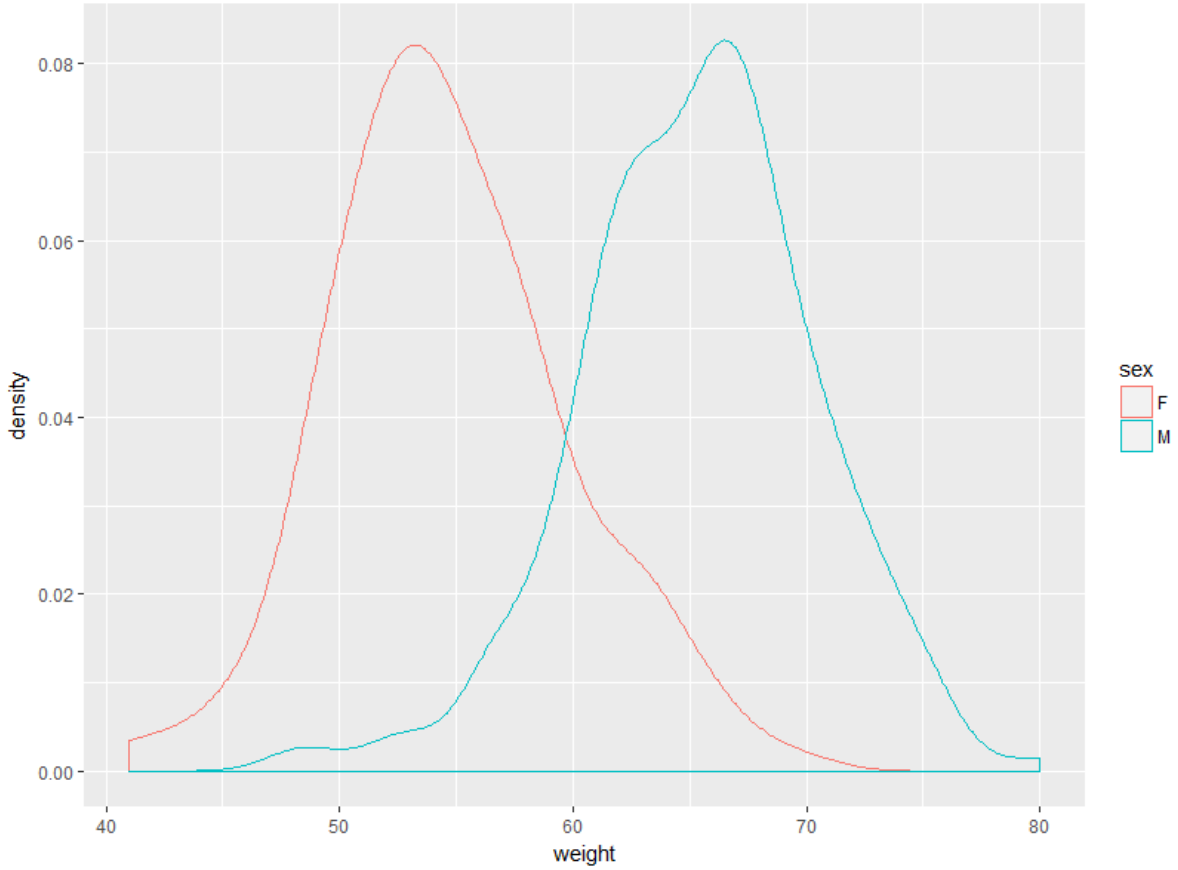
小提琴图

```
1  p + geom_violin(alpha=0.3,width=0.9)+
2      geom_jitter(shape=21)
3
```



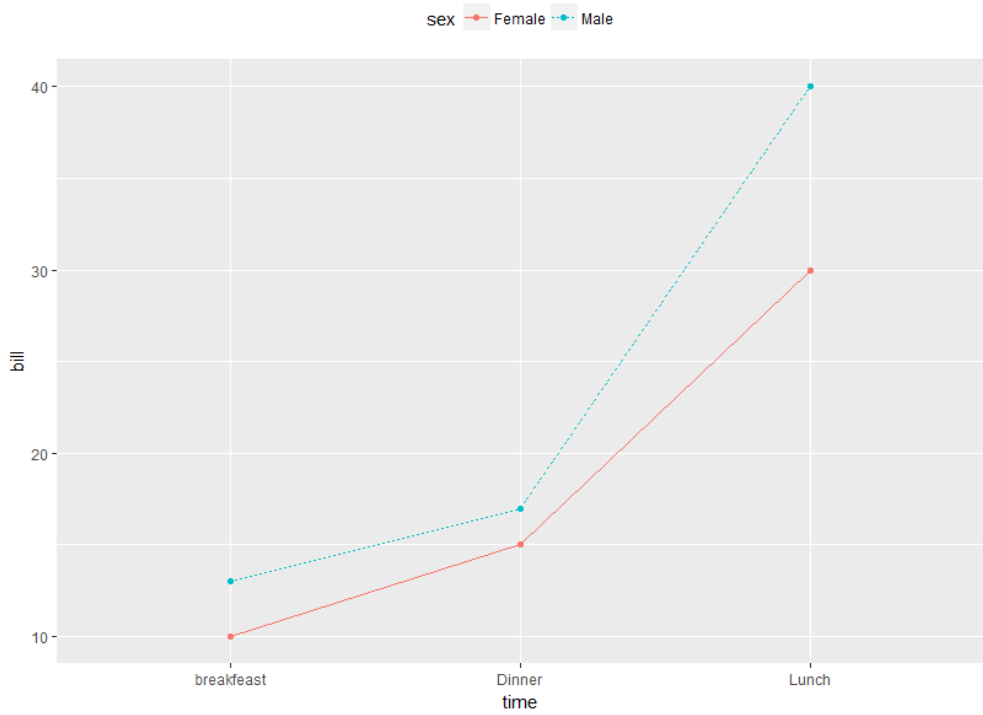
密度图

```
1   set.seed(1234)
2   df <- data.frame(
3     sex=factor(rep(c("F", "M"), each=200)),
4     weight=round(c(rnorm(200, mean=55, sd=5),
5                    rnorm(200, mean=65, sd=5)))
6   )
7   head(df)
8     sex weight
9   1   F     49
10  2   F     56
11  3   F     60
12  4   F     43
13  5   F     57
14  6   F     58
15  p <- ggplot(df, aes(x=weight, color=sex)) +
16    geom_density()
17  p
18
```



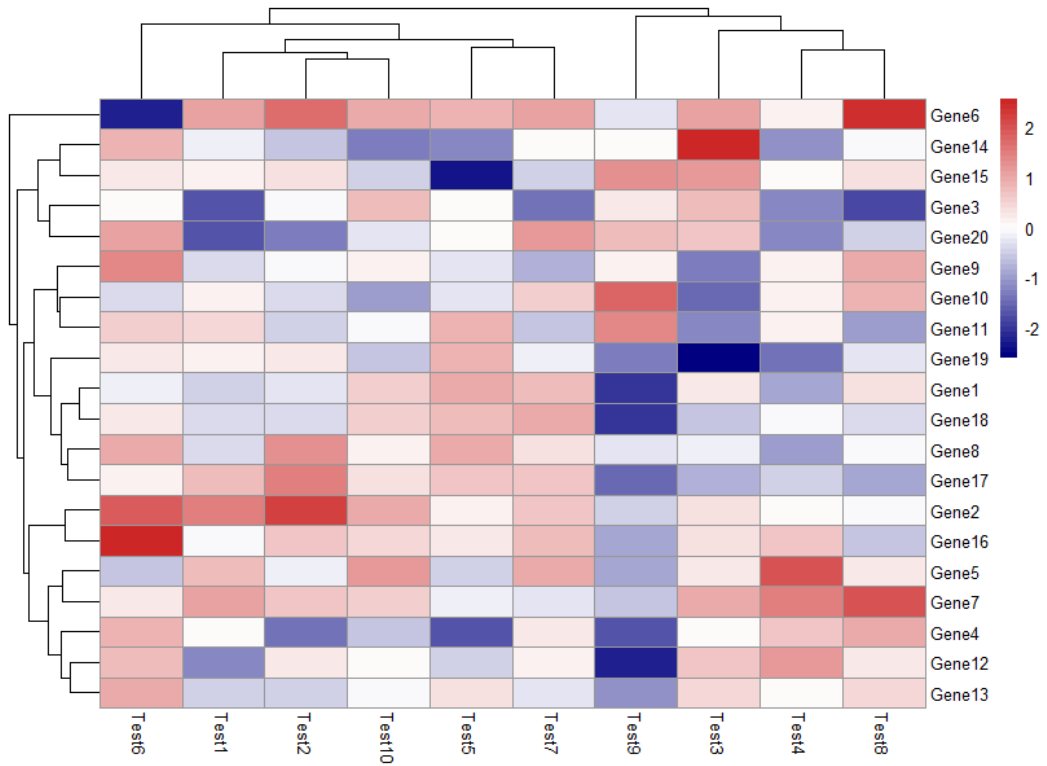
线图

```
df2 <- data.frame(sex = rep(c("Female", "Male"), each=3),
                  time=c("breakfeast", "Lunch", "Dinner"),
                  bill=c(10, 30, 15, 13, 40, 17) )
head(df2)
     sex       time bill
1 Female breakfeast   10
2 Female      Lunch   30
3 Female     Dinner   15
4   Male breakfeast   13
5   Male      Lunch   40
6   Male     Dinner   17
P <- ggplot(df2, aes(x=time, y=bill, group=sex)) +
  geom_line(aes(linetype=sex, color=sex))+
  geom_point(aes(color=sex))+
  theme(legend.position="top")
P

```



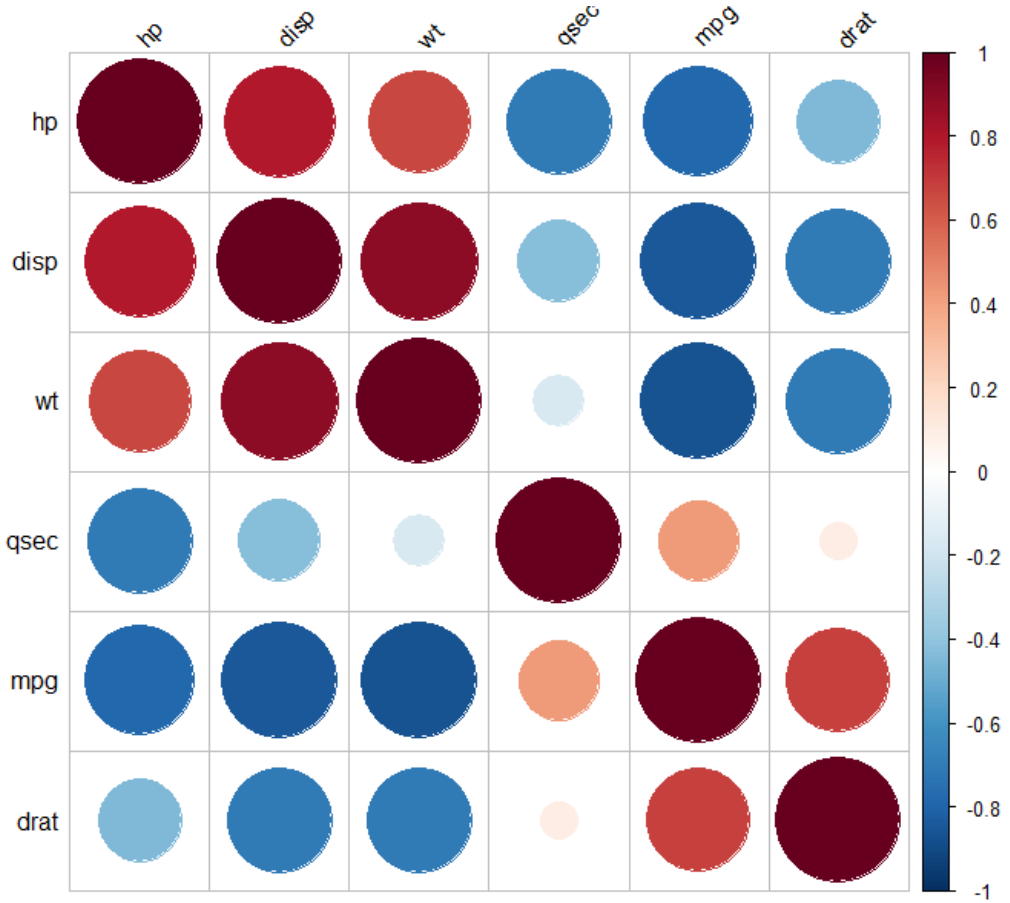
热图

```
1    library(pheatmap)
2    test = matrix(rnorm(200), 20, 10)
3    help(rnorm)
4    ------------------------------------------------------------------------
5    random generation for the normal distribution with
6    mean equal to mean and standard deviation equal to sd.
7    rnorm(n, mean = 0, sd = 1)
8    ------------------------------------------------------------------------
9    colnames(test) = paste("Test", 1:10, sep = "")
10   rownames(test) = paste("Gene", 1:20, sep = "")
11   pheatmap(test, color = colorRampPalette(c("navy", "white", "firebrick3"))(50))
12
```



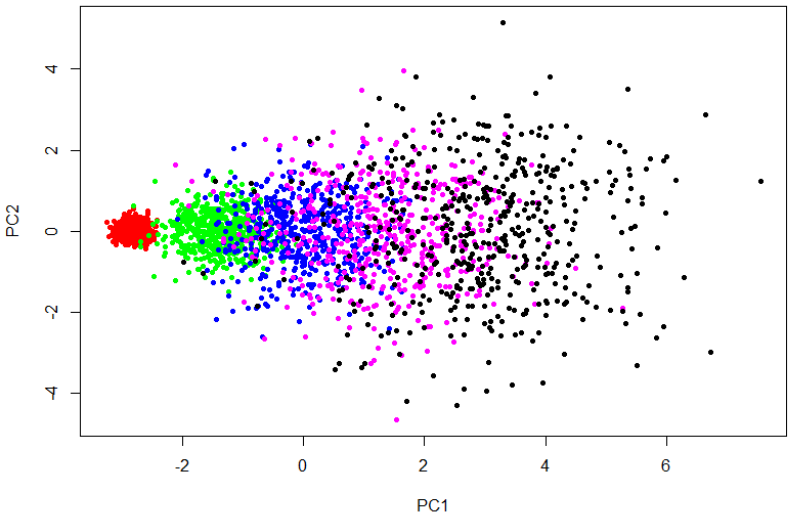
相关性图

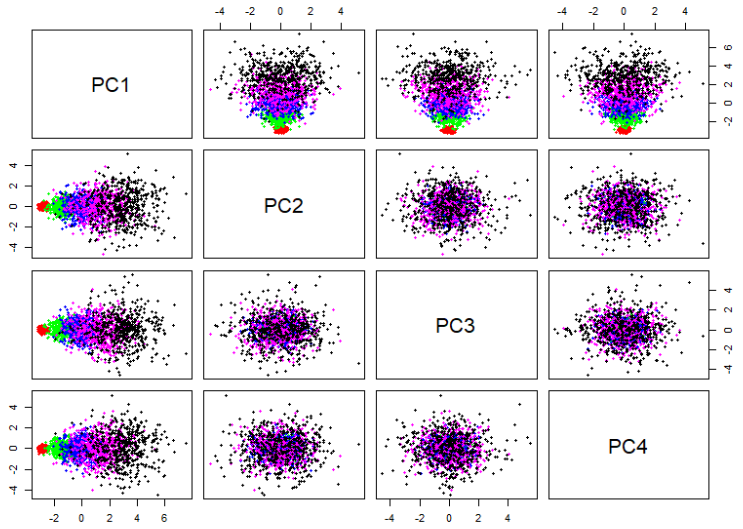
```
1    install.packages("corrplot")
2    library(corrplot)
3    mydata <- select(mtcars,hp,disp,wt,qsec,mpg,drat)
4    source("http://www.sthda.com/upload/rquery_cormat.r")
5  ⊞ rquery.cormat<-function(x, type=c('lower', 'upper', 'full', 'flatten'),...
8  ⊞ {...
91   }
92   rquery.cormat(mydata, type="full")
93
```

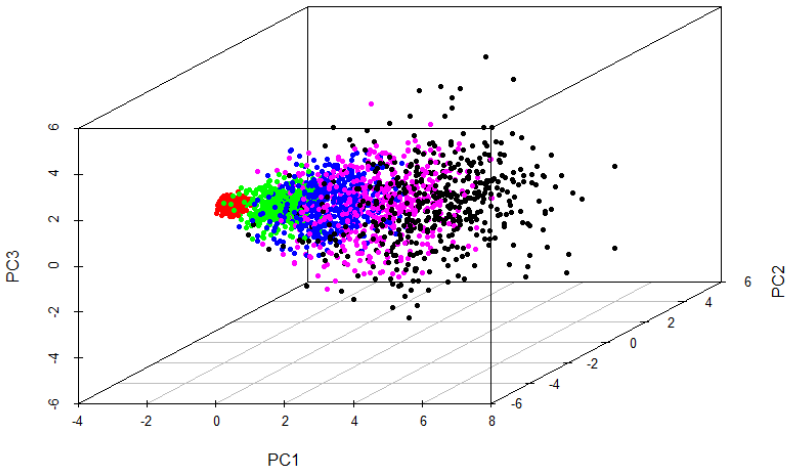


主成份分析 (PCA)

```
z1 <- rnorm(10000, mean=1, sd=1)
z2 <- rnorm(10000, mean=3, sd=3)
z3 <- rnorm(10000, mean=5, sd=5)
z4 <- rnorm(10000, mean=7, sd=7)
z5 <- rnorm(10000, mean=9, sd=9)
mydata <- matrix(c(z1, z2, z3, z4, z5), 2500, 20, byrow=T, dimnames=list(paste("R", 1:2500, sep=""), paste("C", 1:20, sep="")))
pca <- prcomp(mydata, scale=T)
summary(pca)$importance[, 1:6]
mycolors <- c("red", "green", "blue", "magenta", "black")
-------------------------------------------------------------------------------------------------------------------------
plot(pca$x[,1:2], pch=20, col=mycolors[sort(rep(1:5, 500))])
-------------------------------------------------------------------------------------------------------------------------
pairs(pca$x[,1:4], pch=20, col=mycolors[sort(rep(1:5, 500))])
-------------------------------------------------------------------------------------------------------------------------
library(scatterplot3d)
scatterplot3d(pca$x[,1:3], pch=20, color=mycolors[sort(rep(1:5, 500))])
```





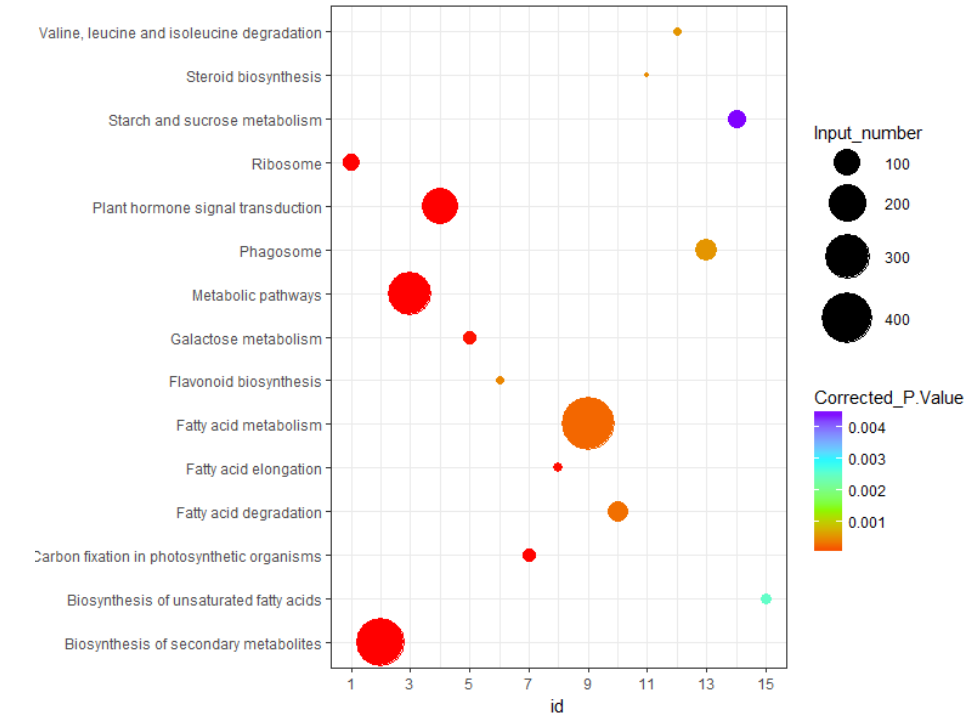


气泡图 (Bubbles)

```
require(ggplot2)
df<- read.csv("Bubbles.csv")
```

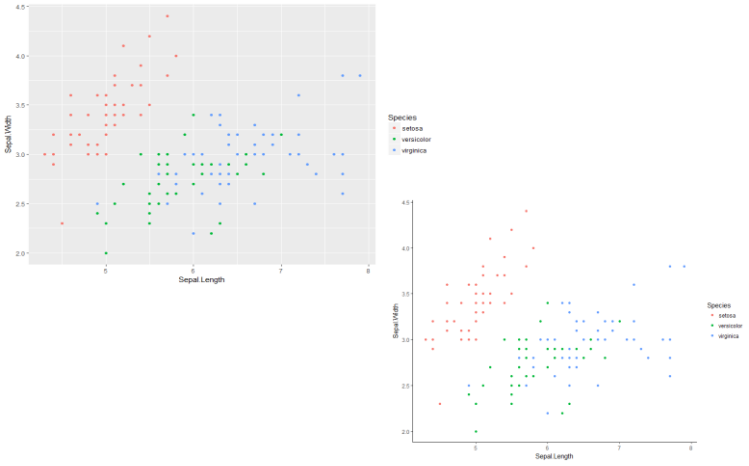
```
> df
                                      Term Input_number                P.Value id
1                                 Ribosome           45 0.0000000000000797516  1
2       Biosynthesis of secondary metabolites          354 0.0000000000000149654  2
3                           Metabolic pathways          282 0.0000000000394082733  3
4           Plant hormone signal transduction          193 0.0000001163614774940  4
5                         Galactose metabolism           29 0.0000148971286000000  5
6                       Flavonoid biosynthesis           18 0.0004835170237970000  6
7  Carbon fixation in photosynthetic organisms           30 0.0000019771230000000  7
8                         Fatty acid elongation           20 0.0000085289000000000  8
9                         Fatty acid metabolism          424 0.0002962069999999999  9
10                      Fatty acid degradation           58 0.0003399950000000000 10
11                        Steroid biosynthesis           16 0.0005169999999999999 11
12  Valine, leucine and isoleucine degradation           19 0.0005875519999999997 12
13                                  Phagosome           67 0.0005879299999999996 13
14              Starch and sucrose metabolism           48 0.0044200000000000032 14
15    Biosynthesis of unsaturated fatty acids           23 0.0025335000000000016 15
```

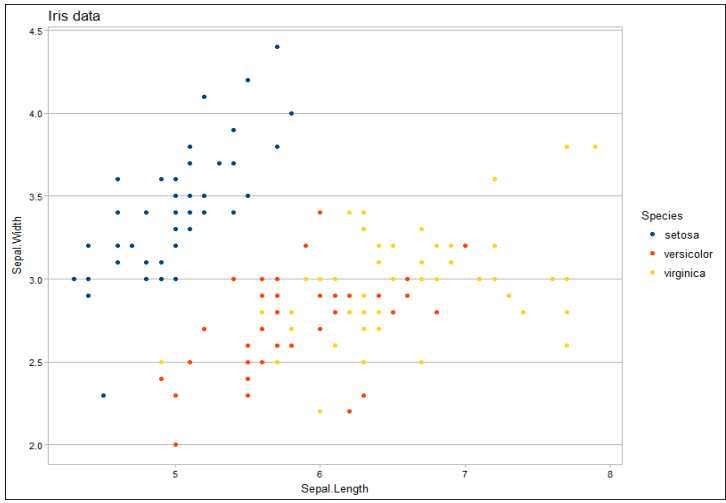
```
ggplot(df, aes(x = id,y=Term,label = Term)) +
  geom_point(aes(size = Input_number, colour = P.Value)) +
  #geom_text(hjust = 1, size = 2) +
  scale_size(range = c(1,15)) +
  scale_x_continuous(breaks = seq(1, 15, 2)) +
  scale_colour_gradientn(colours=rainbow(4)) +
  theme_bw()
```

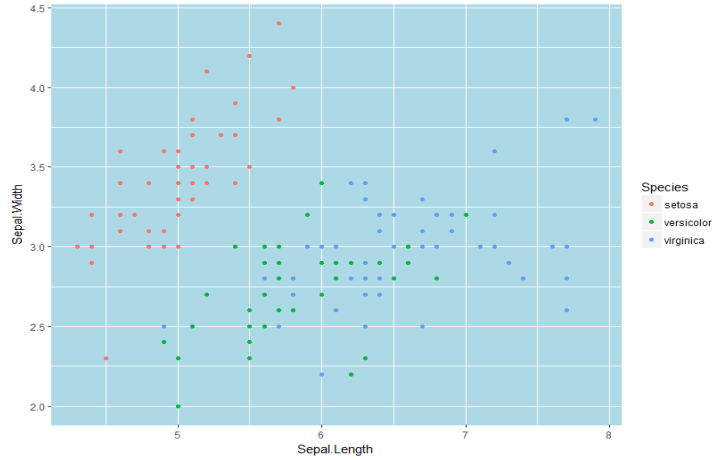


美化 (themes and background)

```r
# ggplot2自带主题
p <- ggplot(iris, aes(Sepal.Length, Sepal.Width, colour = Species))+
  geom_point()
p
p + theme_classic()
----------------------------------------------------------------------
# 主题包
install.packages("ggthemes") # Install
library(ggthemes) # Load
p + theme_calc()+ scale_colour_calc()+
  ggtitle("Iris data")
----------------------------------------------------------------------
# 定制主题
p + theme(
  panel.background = element_rect(fill = "lightblue",
                                  colour = "lightblue",
                                  size = 0.5, linetype = "solid"),
  panel.grid.major = element_line(size = 0.5, linetype = 'solid',
                                  colour = "white"),
  panel.grid.minor = element_line(size = 0.25, linetype = 'solid',
                                  colour = "white")
)

```



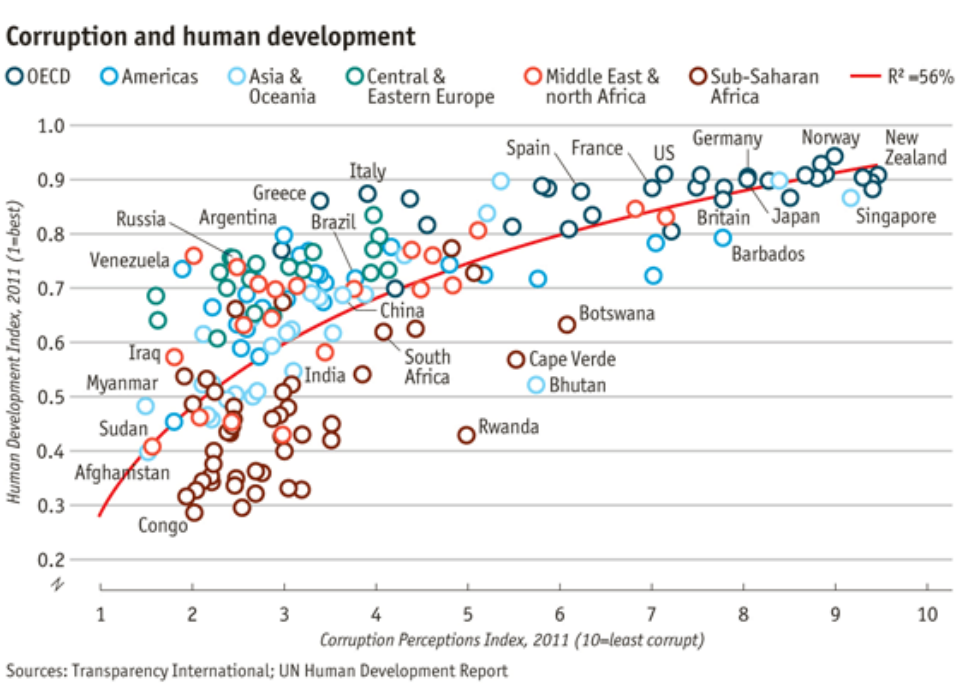




RNA-Seq (DESeq2)

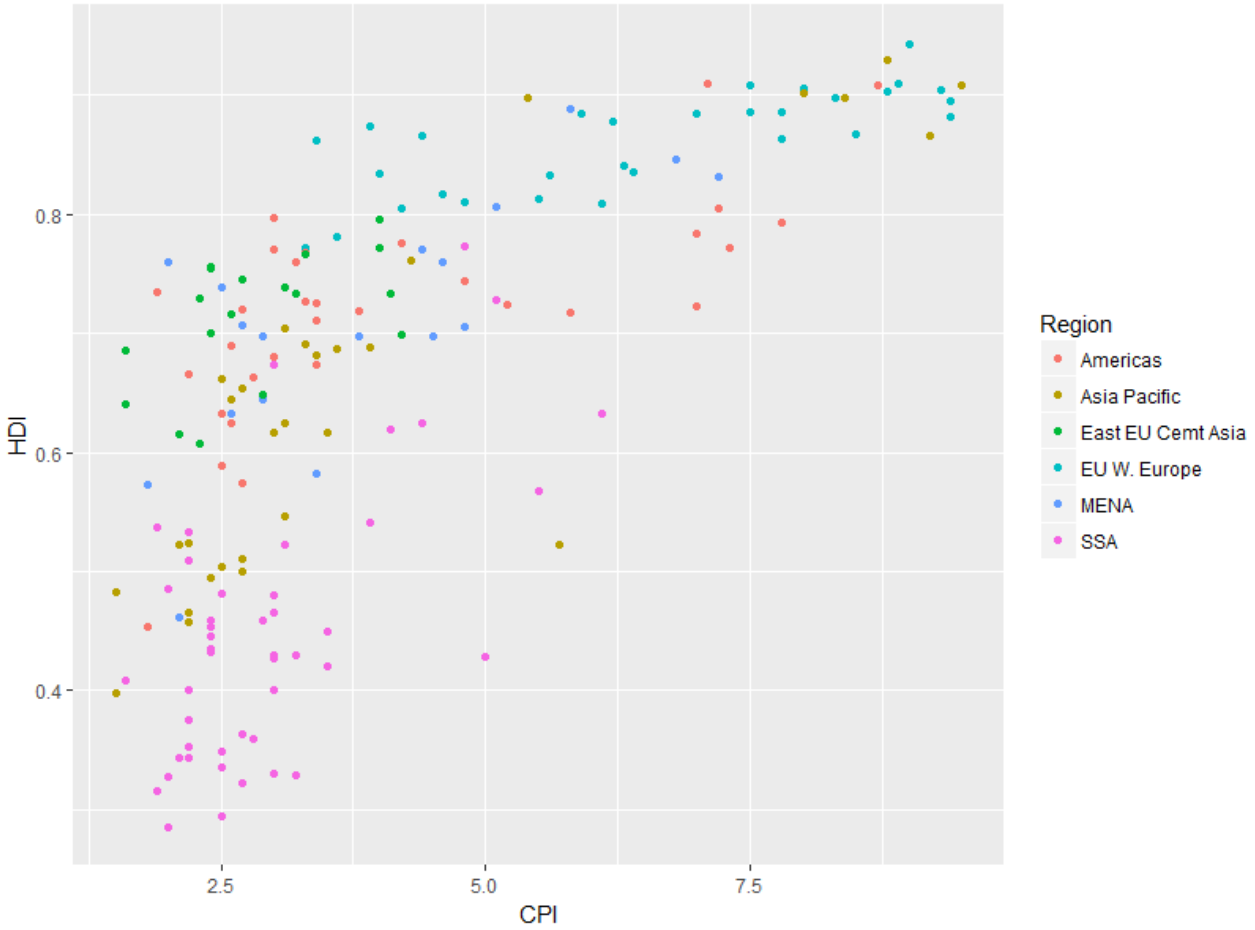
```
library(DESeq2);library(limma);library(pasilla);data(pasillaGenes);exprSet=counts(pasillaGenes)
# 表达距阵
head(exprSet)
            treated1fb treated2fb treated3fb untreated1fb untreated2fb untreated3fb untreated4fb
FBgn0000003          0          0          1            0            0            0            0
FBgn0000008         78         46         43           47           89           53           27
FBgn0000014          2          0          0            0            0            1            0
FBgn0000015          1          0          1            0            1            1            2
FBgn0000017       3187       1672       1859         2445         4615         2063         1711
FBgn0000018        369        150        176          288          383          135          174
# 构建dds对象
colData <- data.frame(row.names=colnames(exprSet), group_list=group_list)
> colData
            group_list
treated1fb      treated
treated2fb      treated
treated3fb      treated
untreated1fb  untreated
untreated2fb  untreated
untreated3fb  untreated
untreated4fb  untreated
dds <- DESeqDataSetFromMatrix(countData = exprSet,colData = colData,design = ~ group_list)
# normalization
dds2 <- DESeq(dds)
# 提取差异分析结果
resultsNames(dds2)
res <-  results(dds2, contrast=c("group_list","treated","untreated"))
resOrdered <- res[order(res$padj),]
resOrdered=as.data.frame(resOrdered)
head(resOrdered)
             baseMean log2FoldChange     lfcSE      stat        pvalue          padj
FBgn0039155  453.2753      -3.714214 0.1600580 -23.20543 4.013291e-119 3.089431e-115
FBgn0029167 2165.0445      -2.082793 0.1035963 -20.10491  6.684454e-90  2.572846e-86
FBgn0035085  366.8279      -2.227243 0.1369744 -16.26028  1.888618e-59  4.846194e-56
FBgn0029896  257.9027      -2.206780 0.1586969 -13.90563  5.854593e-44  1.126716e-40
FBgn0034736  118.4074      -2.565002 0.1847628 -13.88268  8.067448e-44  1.242064e-40
FBgn0040091  610.6035      -1.430433 0.1201539 -11.90501  1.114552e-32  1.429970e-29

```

复杂图形修改


Corruption and human development
OECD
Americas
Asia & Oceania
Central & Eastern Europe
Middle East & north Africa
Sub-Saharan Africa
R² =56%
Human Development Index, 2011 (1=best)
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
Spain
France
US
Germany
Norway
New Zealand
Italy
Greece
Russia
Argentina
Brazil
Britain
Japan
Singapore
Venezuela
Barbados
China
Botswana
Iraq
South Africa
Cape Verde
India
Myanmar
Bhutan
Sudan
Rwanda
Afghanistan
Congo
1
2
3
4
5
6
7
8
9
10
Corruption Perceptions Index, 2011 (10=least corrupt)
Sources: Transparency International; UN Human Development Report

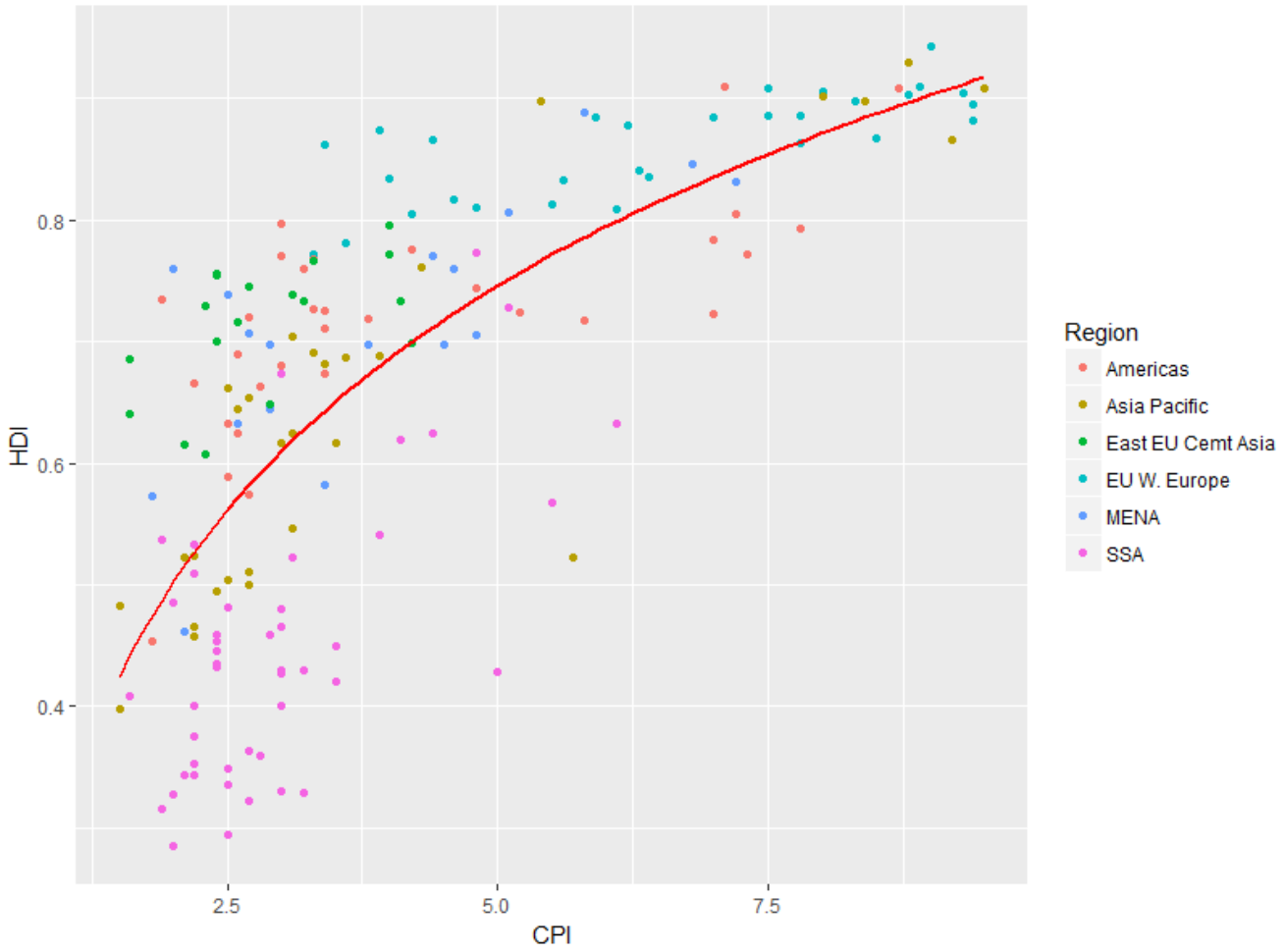
1. 基本图形

```
1  library(ggplot2)
2  dat <- read.csv("EconomistData.csv")
3  # Basic plot
4  pc1 <- ggplot(dat,aes(x = CPI, y = HDI, color = Region))+
5    geom_point()
6  pc1
7
```



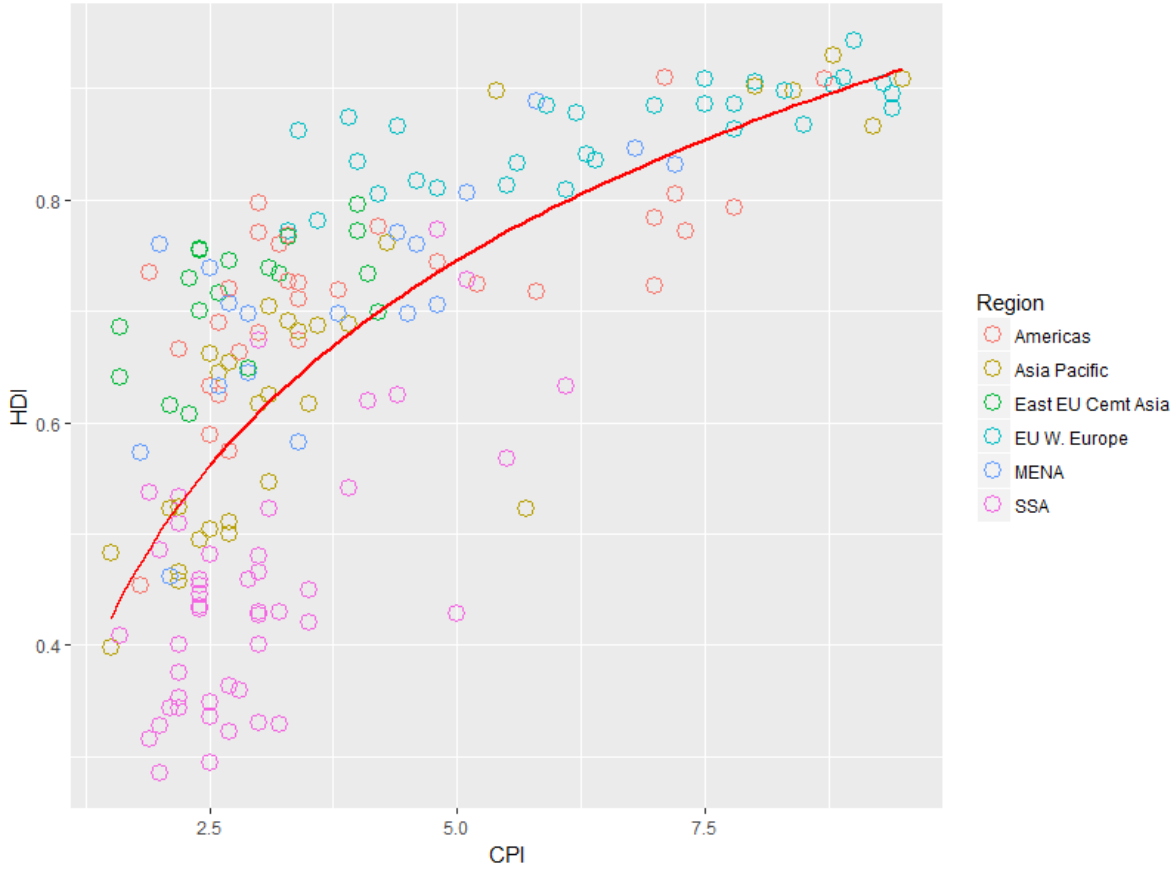
2. 添加趋势线

```
1  # Trend line
2  pc2 <- pc1 +
3    geom_smooth(aes(group = 1),
4                method = "lm",
5                formula = y ~ log(x),
6                se = FALSE,
7                color = "red")
8  pc2
9
```



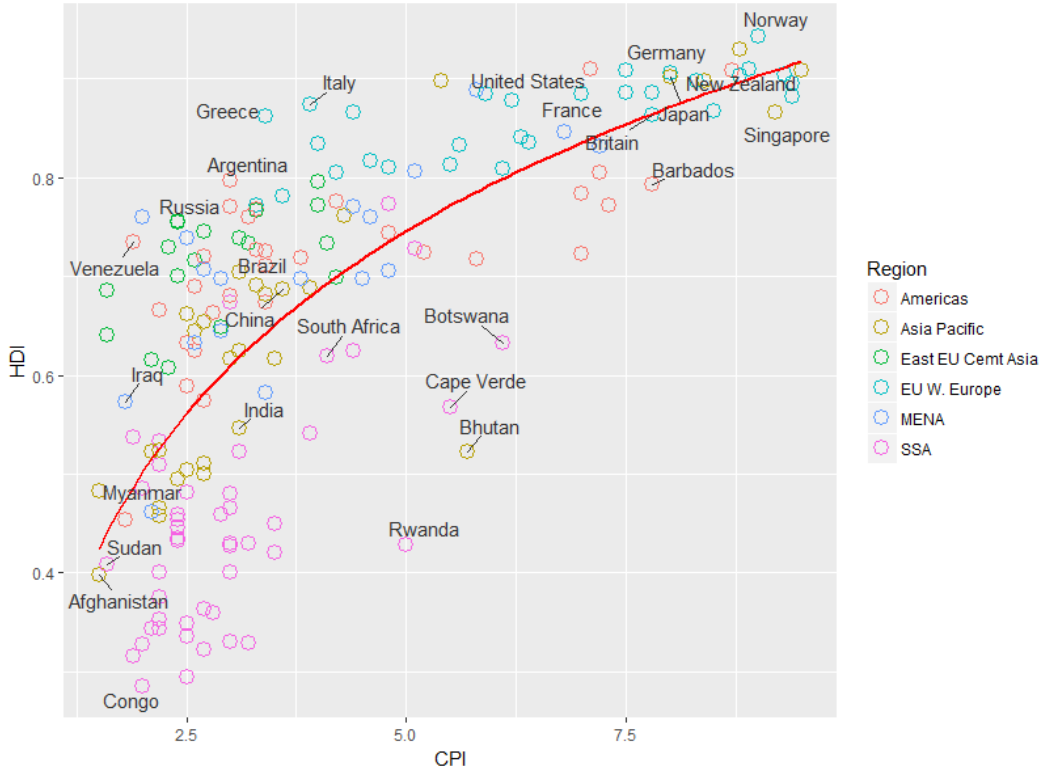
3. 圆圈

```
1
2
3   # Open points
4   pc3 <- ggplot(dat,aes(x = CPI, y = HDI, color = Region))+
5     geom_point(shape = 1, size = 4) +
6     geom_smooth(aes(group = 1),
7                 method = "lm",
8                 formula = y ~ log(x),
9                 se = FALSE,
10                color = "red")
11  pc3
12
```



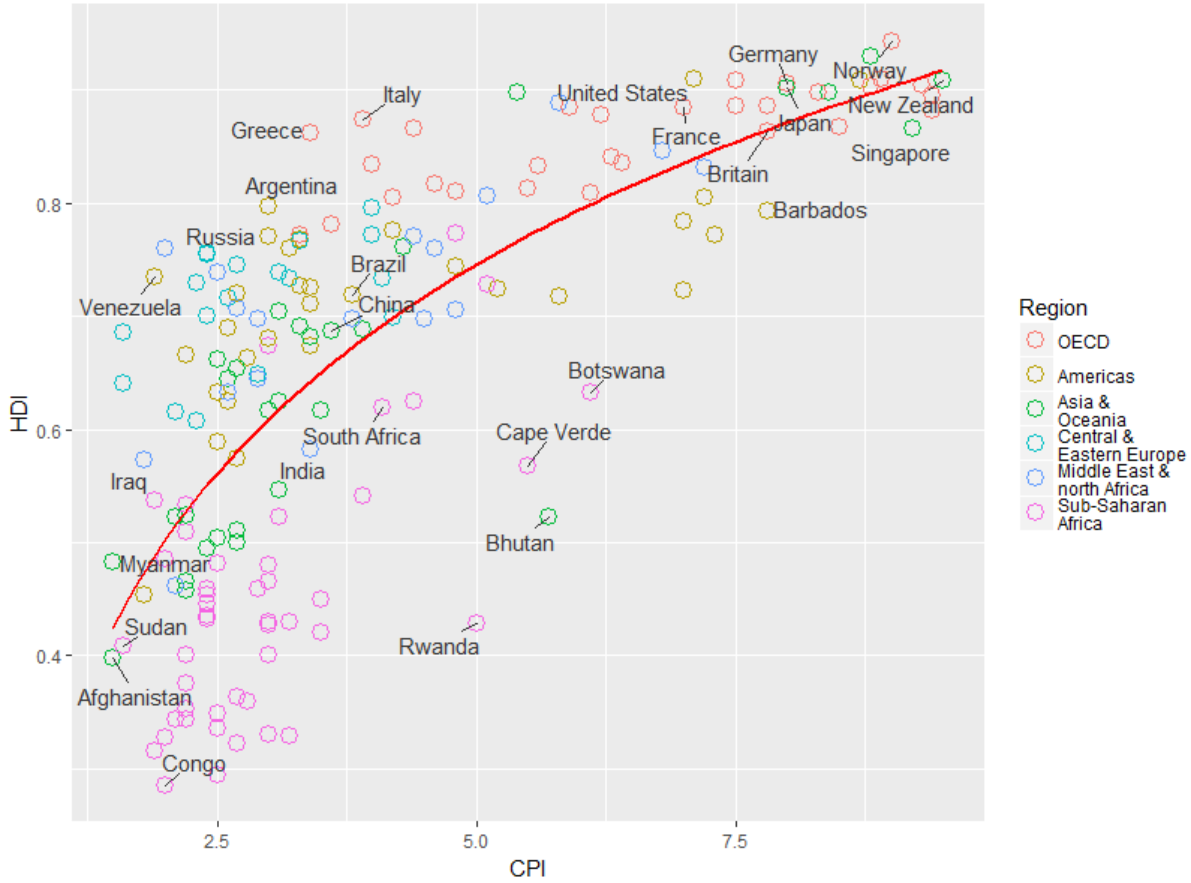
4. 标注想要的点

```
pointsToLabel <- c("Russia", "Venezuela", "Iraq", "Myanmar", "Sudan",
                   "Afghanistan", "Congo", "Greece", "Argentina", "Brazil",
                   "India", "Italy", "China", "South Africa", "Spane",
                   "Botswana", "Cape Verde", "Bhutan", "Rwanda",
                   "France","United States", "Germany", "Britain", "Barbados",
                   "Norway", "Japan",
                   "New Zealand", "Singapore")
library("ggrepel")
pc4 <- pc3 +  geom_text_repel(aes(label = Country),
                              color = "gray20",
                              data = subset(dat, Country %in% pointsToLabel),
                              force = 10)
pc4

```



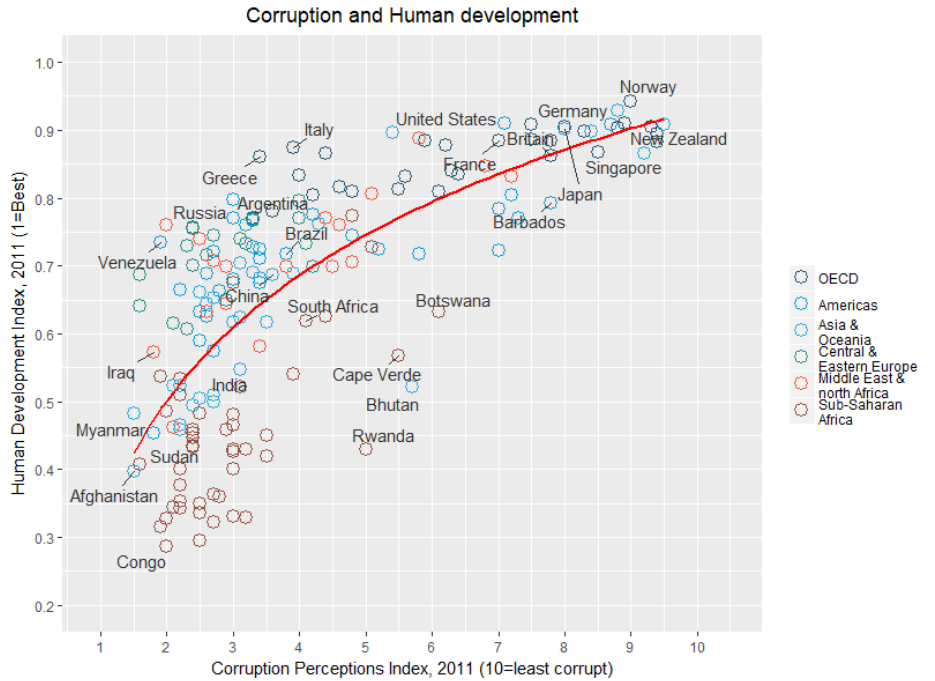
5. 修改图例值和顺序

```
dat$Region <- factor(dat$Region,
                     levels = c("EU W. Europe",
                                "Americas",
                                "Asia Pacific",
                                "East EU Cemt Asia",
                                "MENA",
                                "SSA"),
                     labels = c("OECD",
                                "Americas",
                                "Asia &\nOceania",
                                "Central &\nEastern Europe",
                                "Middle East &\nnorth Africa",
                                "Sub-Saharan\nAfrica"))
pc4$data <- dat
pc4

```



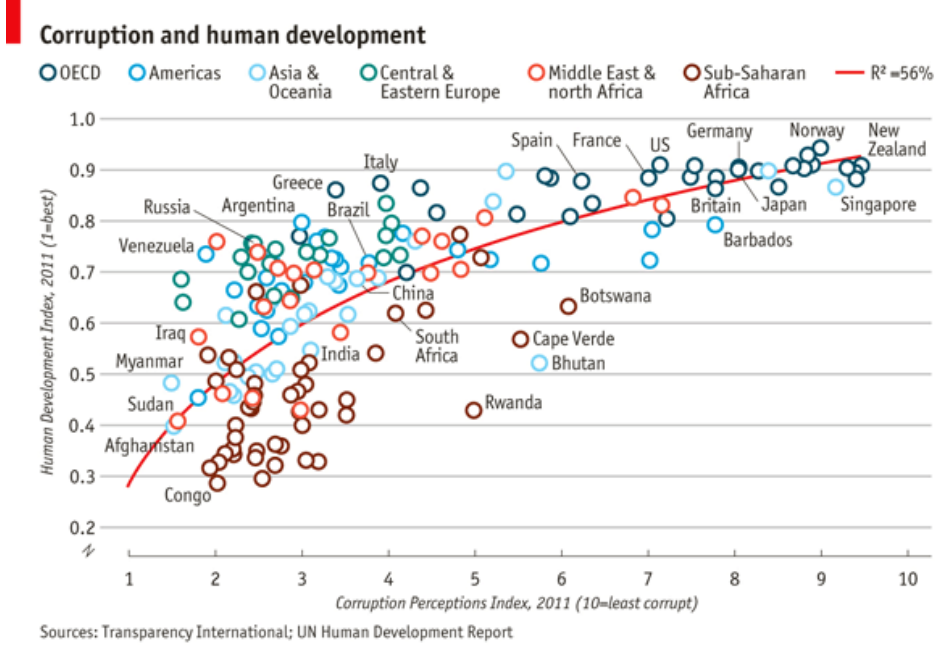
6. 利用scale来修改x，y轴，颜色和标出title

```
library(grid)
pc5 <- pc4 +
  scale_x_continuous(name = "Corruption Perceptions Index, 2011 (10=least corrupt)",
                     limits = c(.9, 10.5),
                     breaks = 1:10) +
  scale_y_continuous(name = "Human Development Index, 2011 (1=Best)",
                     limits = c(0.2, 1.0),
                     breaks = seq(0.2, 1.0, by = 0.1)) +
  scale_color_manual(name = "",
                     values = c("#24576D",
                                "#099DD7",
                                "#28AADC",
                                "#248E84",
                                "#F2583F",
                                "#96503F")) +
  ggtitle("Corruption and Human development") +
  theme(plot.title = element_text(hjust = 0.5))
pc5

```

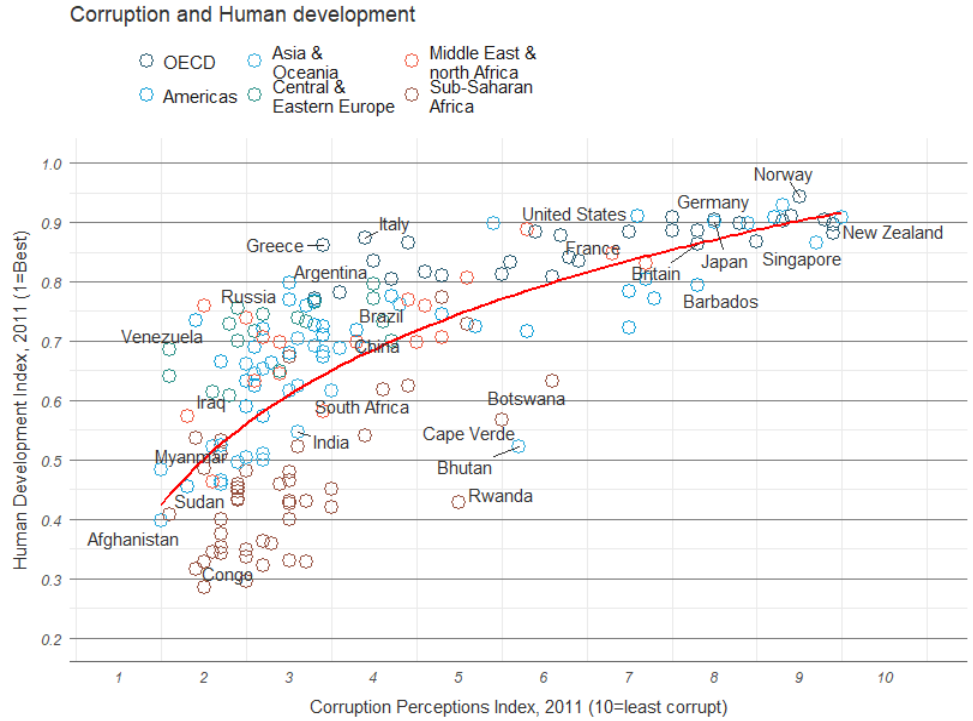


7. 微调主题

```
  library(grid) # for the 'unit' function
pc6 <- pc5 +
  theme_minimal() + # start with a minimal theme and add what we need
  theme(text = element_text(color = "gray20"),
        legend.position = c("top"), # position the legend in the upper left
        legend.direction = "horizontal",
        legend.justification = 0.1, # anchor point for legend.position.
        legend.text = element_text(size = 11, color = "gray10"),
        axis.text = element_text(face = "italic"),
        axis.title.x = element_text(vjust = -1), # move title away from axis
        axis.title.y = element_text(vjust = 2), # move away for axis
        axis.ticks.y = element_blank(), # element_blank() is how we remove elements
        axis.line = element_line(color = "gray40", size = 0.5),
        axis.line.y = element_blank(),
        panel.grid.major = element_line(color = "gray50", size = 0.5),
        panel.grid.major.x = element_blank()
  )
pc6

```



Sources: Transparency International; UN Human Development Report



THANKS FOR YOUR WATCHING

http://tiramisutes.github.io/

@Huazhong Agricultural University