

# Exploratory graphs

Roger D. Peng, Associate Professor of Biostatistics

May 18, 2016

# Why do we use graphs in data analysis?

- ▶ To understand data properties
- ▶ To find patterns in data
- ▶ To suggest modeling strategies
- ▶ To “debug” analyses
- ▶ To communicate results

# Exploratory graphs

- ▶ To understand data properties
- ▶ To find patterns in data
- ▶ To suggest modeling strategies
- ▶ To “debug” analyses
- ▶ To communicate results

## Characteristics of exploratory graphs

- ▶ They are made quickly
- ▶ A large number are made
- ▶ The goal is for personal understanding
- ▶ Axes/legends are generally not cleaned up
- ▶ Color/size are primarily used for information

# Housing data

The screenshot shows a web browser window with the URL [www.census.gov/acs/www/data\\_documentation/public\\_use\\_microdata\\_sample/](http://www.census.gov/acs/www/data_documentation/public_use_microdata_sample/). The page is titled "American Community Survey" and features a navigation bar with links for People, Business, Geography, Data, Research, and Newsroom. A search bar is also present. The main content area is titled "Public Use Microdata Sample (PUMS)" and includes sections for "Summary products", "Why Use PUMS?", "What's Available and How Can I Access PUMS?", and "Need Help with PUMS?". On the left, a sidebar lists categories such as Data Releases, Data Product Descriptions, Documentation, Geography, Downloadable data via FTP, Summary File, and Public Use Microdata Sample (PUMS), with the PUMS link currently selected.

## American Community Survey

Main | About the Survey | Guidance for Data Users | **Data & Documentation** | Methodology | Library

---

### Public Use Microdata Sample (PUMS)

Print | Share this page | Connect with us

The American Community Survey (ACS) Public Use Microdata Sample (PUMS) files are a set of untabulated records about individual people or housing units. The Census Bureau produces the PUMS files so that data users can create custom tables that are not available through pretabulated (or summary) ACS data products.

**Summary products**, such as the tables and profiles accessible via American FactFinder (AFF), show data that have already been tabulated for specific geographic areas.

**PUMS files**, in contrast, include population and housing unit records with individual response information such as relationship, sex, educational attainment, and employment status.

**Why Use PUMS?**  
PUMS files are perfect for people, such as students, who are looking for greater accessibility to inexpensive data for research projects. Social scientists often use the PUMS for regression analysis and modeling applications.

**What's Available and How Can I Access PUMS?**  
The Census Bureau produces 1-year, 3-year, and 5-year ACS PUMS files. The 3-year and 5-year PUMS files are multiyear combinations of the 1-year PUMS file with appropriate adjustments to the weights and inflation adjustment factors. The PUMS files are accessible via [American FactFinder](#), the Census Bureau's [FTP site](#), and [DataFerrett](#). Statistical software is needed to use the PUMS files from American FactFinder and the FTP site.

**Need Help with PUMS?**  
Learn more about PUMS in the Compass Products [What PUMS Data Users Need to Know](#) handbook and [Introduction to the PUMS training presentation](#).

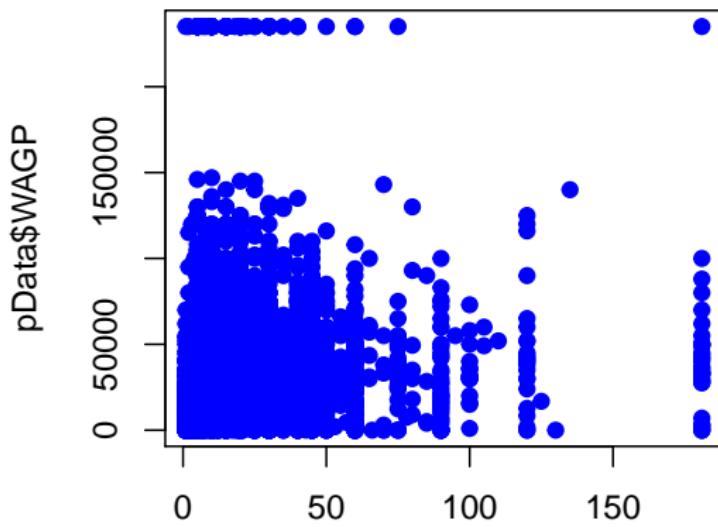
Geographic Areas Available

```
pData <- read.csv("./data/ss06pid.csv")
```

## Scatterplots

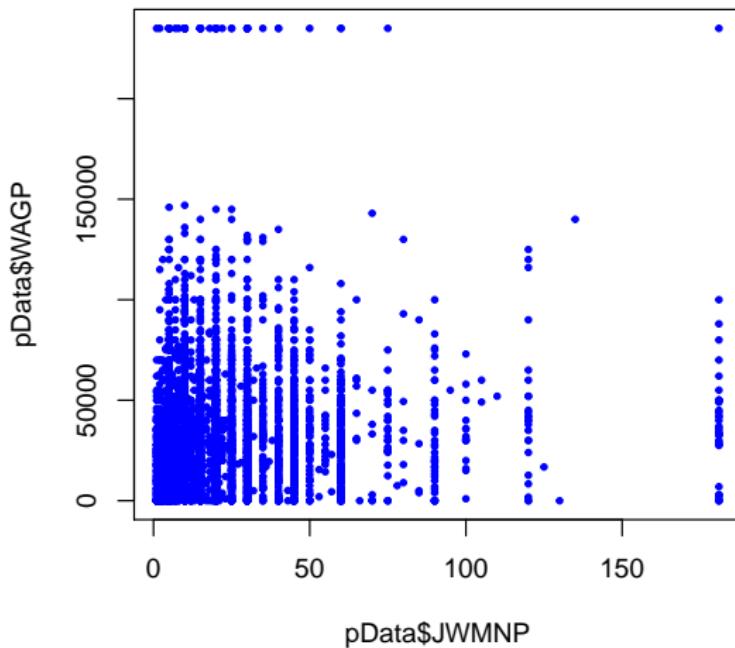
- ▶ Important parameters:  $x, y, type, xlab, ylab, xlim, ylim, cex, col, bg$
- ▶ See `?par` for more

```
plot(pData$JWMNP, pData$WAGP, pch=19, col="blue")
```



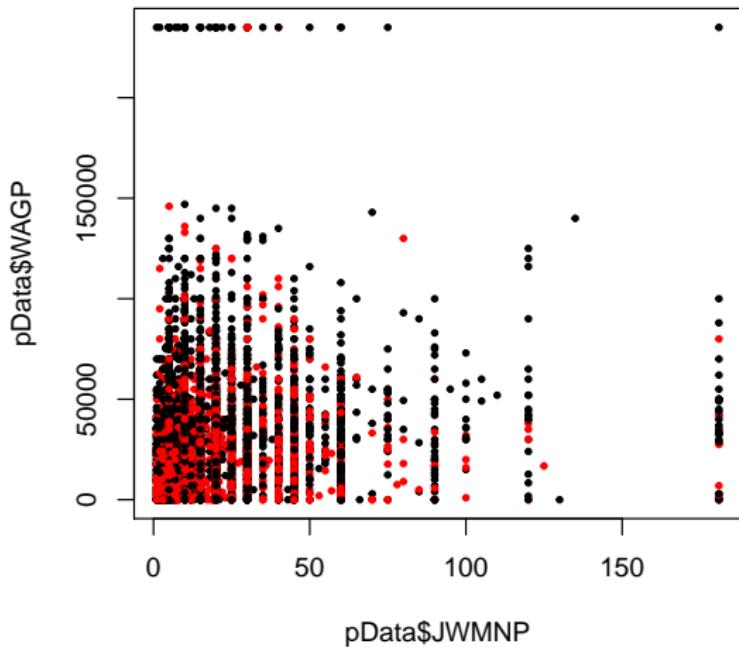
## Scatterplots - size matters

```
plot(pData$JWMNP,pData$WAGP,pch=19,col="blue",cex=0.5)
```



## Scatterplots - using color

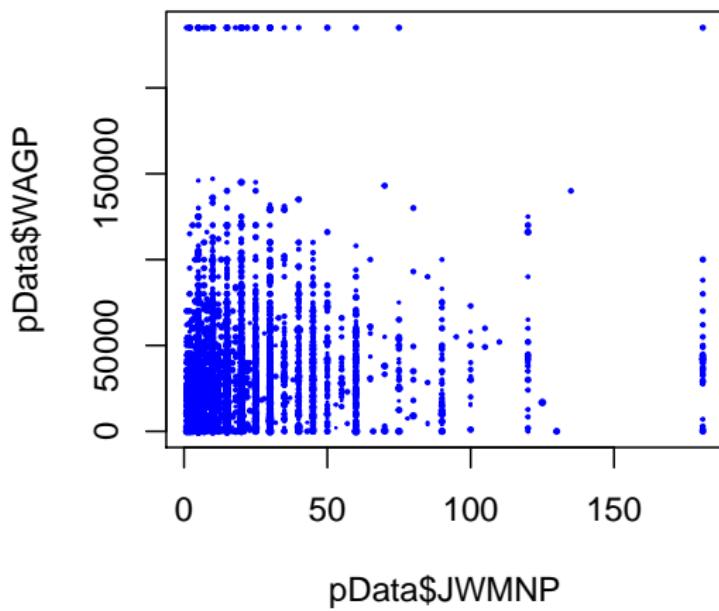
```
plot(pData$JWMNP,pData$WAGP,pch=19,col=pData$SEX,cex=0.5)
```



## Scatterplots - using size

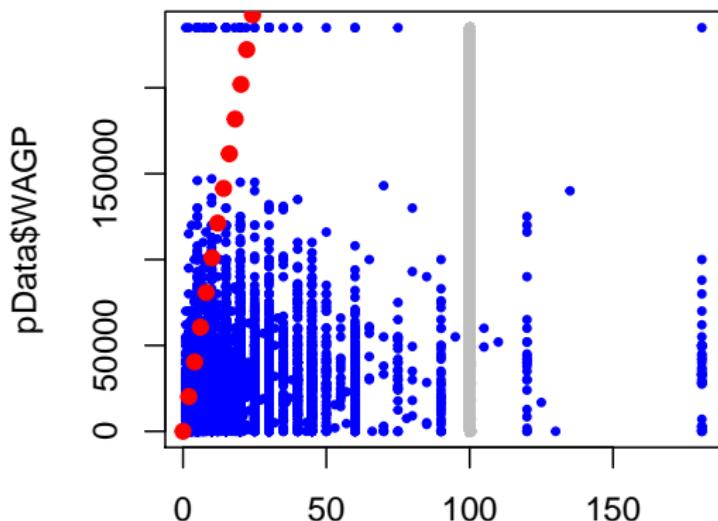
```
percentMaxAge <- pData$AGEP/max(pData$AGEP)
```

```
plot(pData$JWMNP,pData$WAGP,pch=19,col="blue",cex=percentMa
```



## Scatterplots - overlaying lines/points

```
plot(pData$JWMNP,pData$WAGP,pch=19,col="blue",cex=0.5)
lines(rep(100,dim(pData)[1]),pData$WAGP,col="grey",lwd=5)
points(seq(0,200,length=100),seq(0,20e5,length=100),col="red")
```



## Scatterplots - numeric variables as factors

```
library(Hmisc)

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

## Loading required package: ggplot2

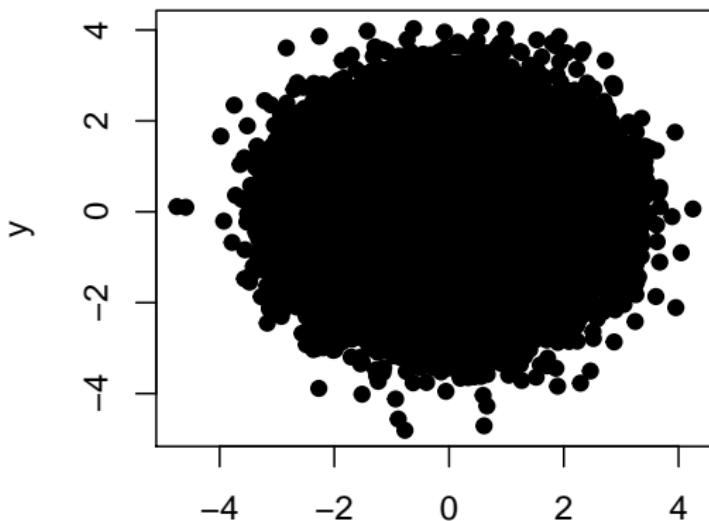
## 
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
## 
##     format.pval, round.POSIXt, trunc.POSIXt, units

ageGroups <- cut2(pData$AGEP,g=5)
plot(pData$JWMNP,pData$WAGP,pch=19,col=ageGroups,cex=0.5)
```

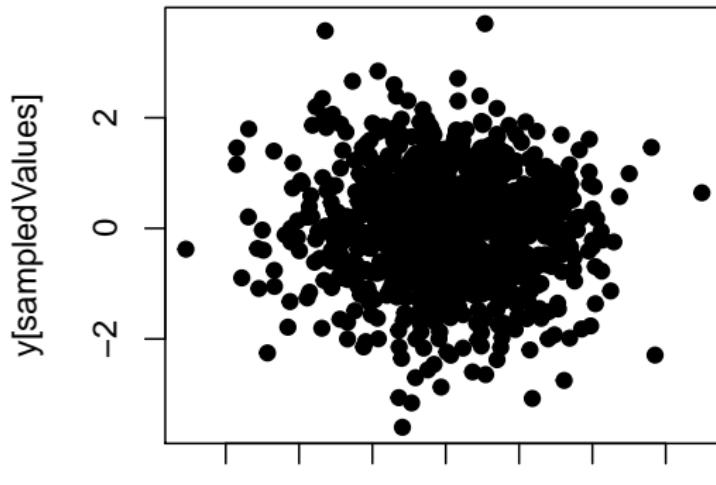
## If you have a lot of points

```
x <- rnorm(1e5)  
y <- rnorm(1e5)  
plot(x,y,pch=19)
```



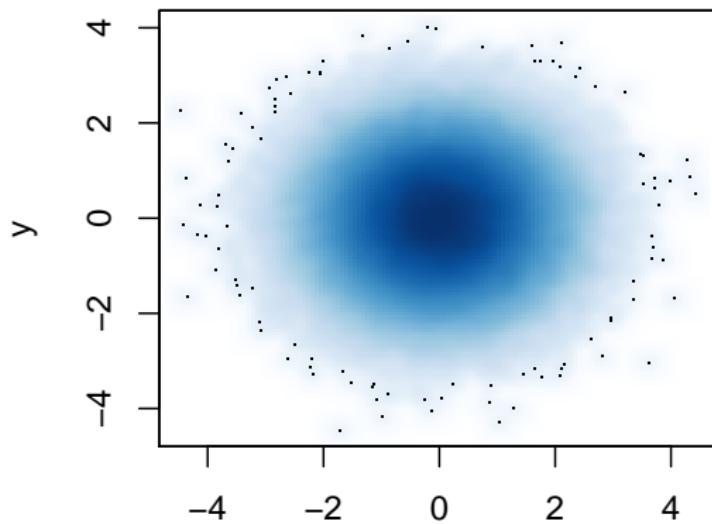
## If you have a lot of points - sampling

```
x <- rnorm(1e5)
y <- rnorm(1e5)
sampledValues <- sample(1:1e5, size=1000, replace=FALSE)
plot(x[sampledValues], y[sampledValues], pch=19)
```



## If you have a lot of points - smoothScatter

```
x <- rnorm(1e5)  
y <- rnorm(1e5)  
smoothScatter(x,y)
```

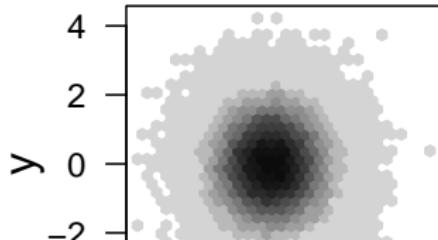


## If you have a lot of points - hexbin {hexbin}

```
library(hexbin)
```

```
## Loading required package: methods
```

```
x <- rnorm(1e5)
y <- rnorm(1e5)
hbo <- hexbin(x,y)
plot(hbo)
```



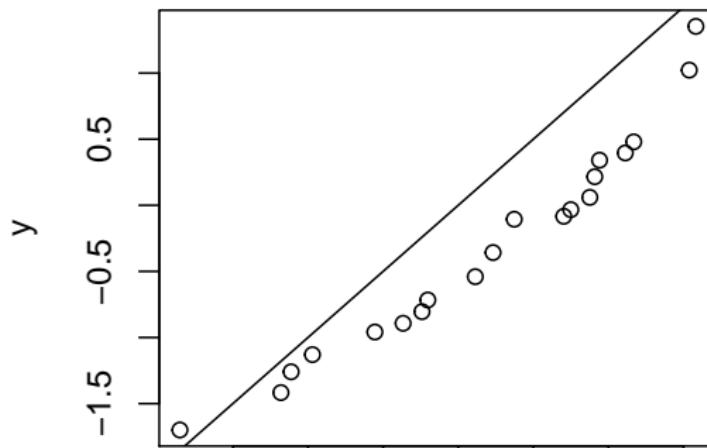
Counts

1196
1047
972
927
848
749
649
549
449
349
249
149
49
4

## QQ-plots

- Important parameters:  $x, y$

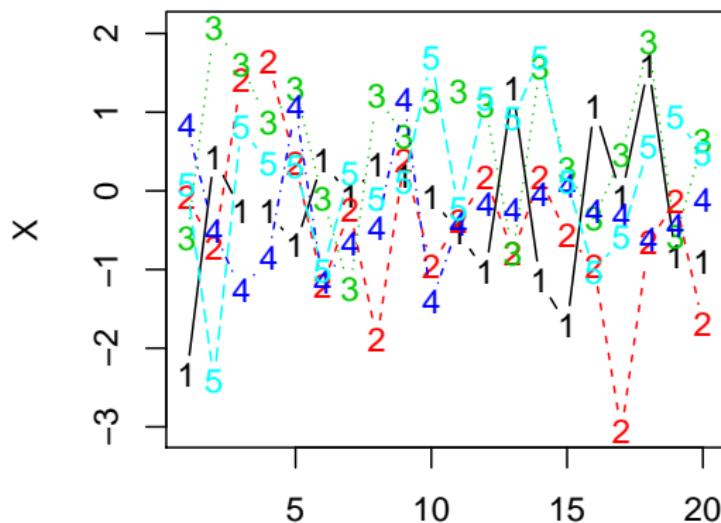
```
x <- rnorm(20); y <- rnorm(20)  
qqplot(x,y)  
abline(c(0,1))
```



# Matplot and spaghetti

- Important parameters:  $x, y, lty, lwd, pch, col$

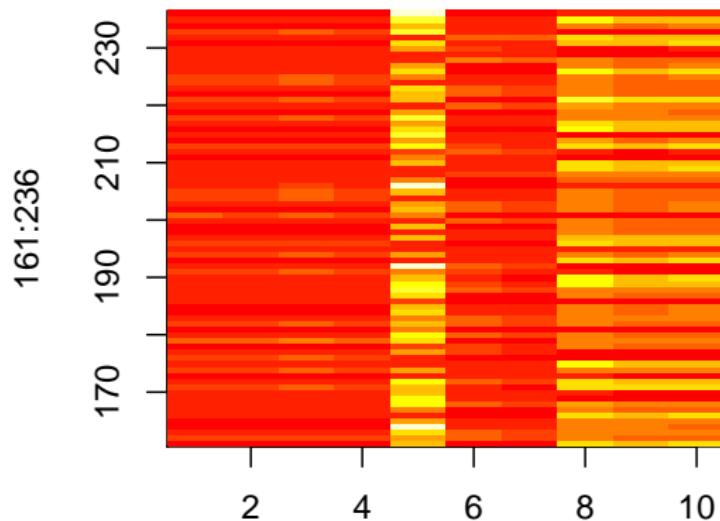
```
X <- matrix(rnorm(20*5), nrow=20)
matplot(X, type="b")
```



## Heatmaps

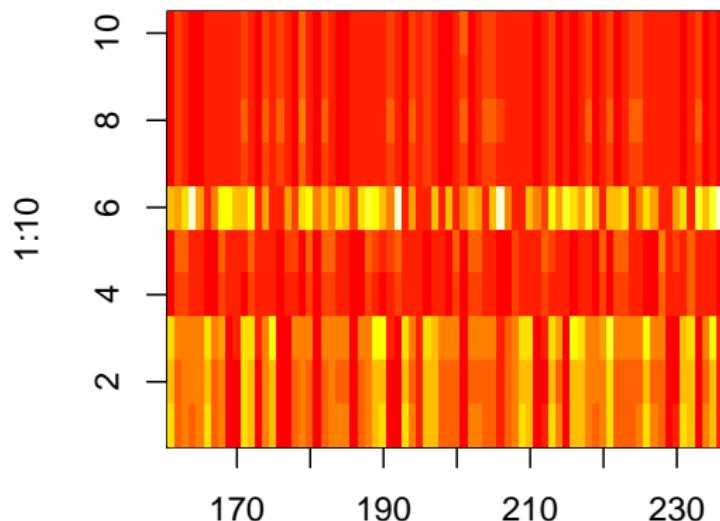
- ▶ Important parameters:  $x, y, z, col$

```
image(1:10, 161:236, as.matrix(pData[1:10, 161:236]))
```



## Heatmaps - matching intuition

```
newMatrix <- as.matrix(pData[1:10,161:236])
newMatrix <- t(newMatrix)[,nrow(newMatrix):1]
image(161:236, 1:10, newMatrix)
```



## Maps - very basics

You make need to run `install.packages("maps")` if you don't have the `maps` package installed.

```
library(maps)
```

```
##
```

```
## # maps v3.1: updated 'world': all lakes moved to separate
## # 'lakes' database. Type '?world' or 'news(package="maps")'
```

```
map("world")
```

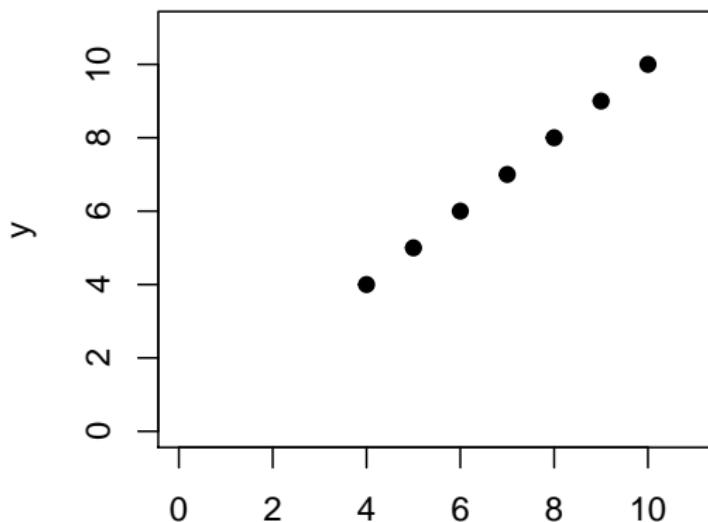
```
lat <- runif(40,-180,180); lon <- runif(40,-90,90)
```

```
points(lat,lon,col="blue",pch=19)
```



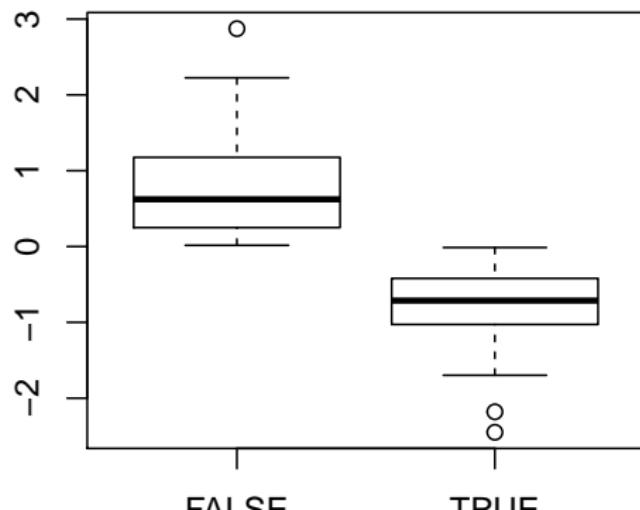
## Missing values and plots

```
x <- c(NA,NA,NA,4,5,6,7,8,9,10)
y <- 1:10
plot(x,y,pch=19,xlim=c(0,11),ylim=c(0,11))
```



## Missing values and plots

```
x <- rnorm(100)
y <- rnorm(100)
y[x < 0] <- NA
boxplot(x ~ is.na(y))
```



## Further resources

- ▶ R Graph Gallery
- ▶ ggplot2,ggplot2 basic introduction
- ▶ lattice package,lattice introduction
- ▶ R bloggers