Exploratory graphs

Part 2

Jeffrey Leek, Assistant Professor of Biostatistics
Johns Hopkins Bloomberg School of Public Health

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

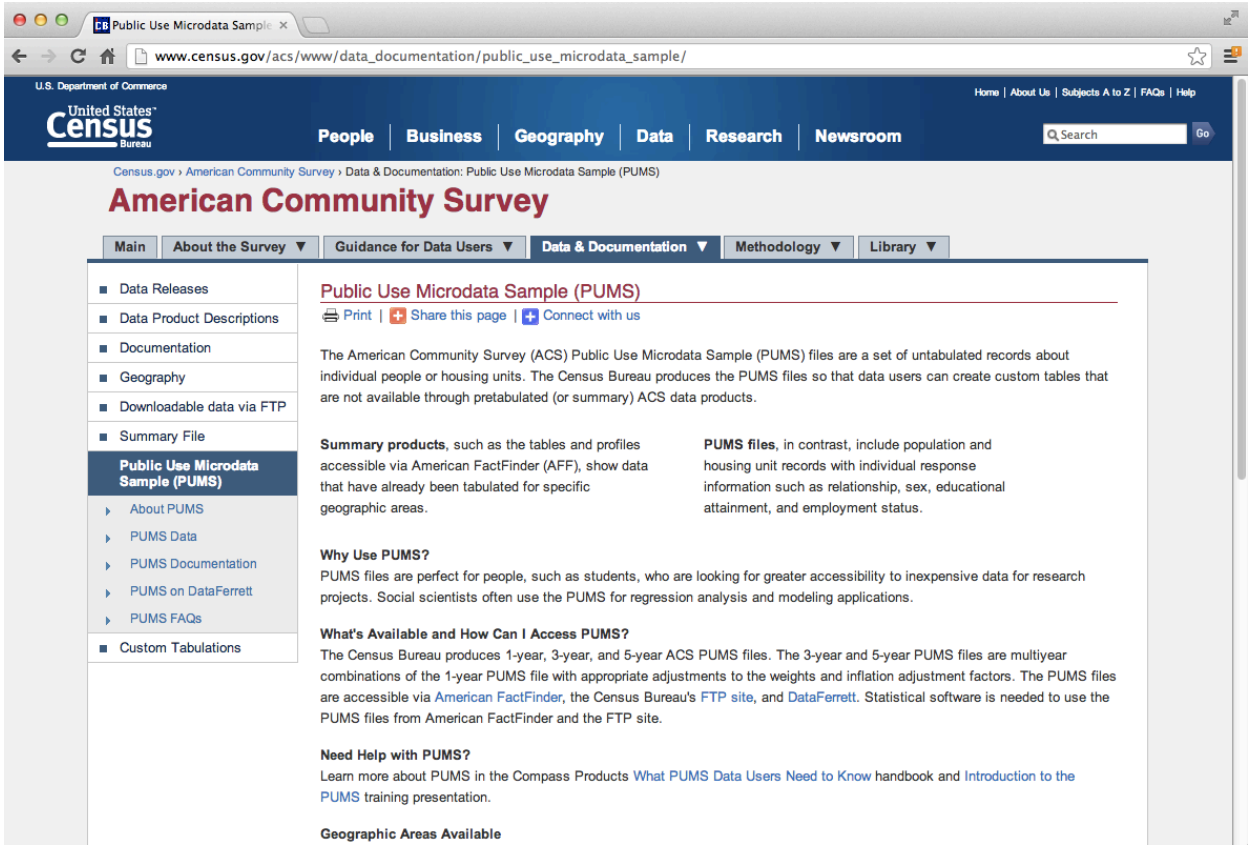
Exploratory graphs

- 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



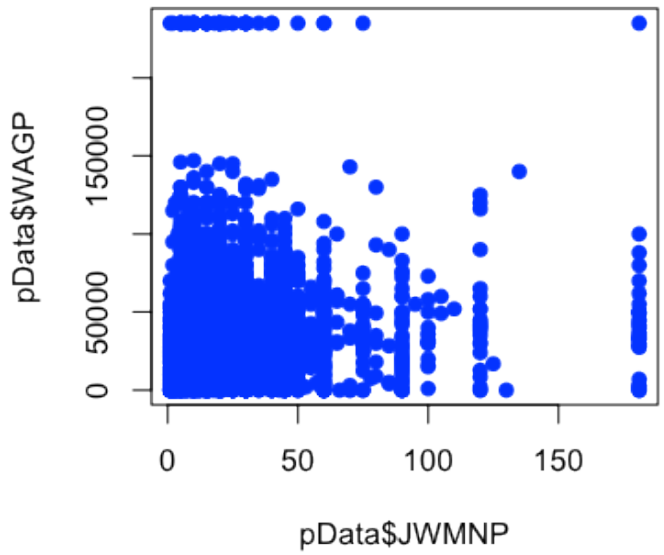

```
pData <- read.csv("./data/ss06pid.csv")
```

5/23

Scatterplots

- Important paramters: *x*,*y*,*type*,*xlab*,*ylab*,*xlim*,*ylim*,*cex*,*col*,*bg*
- See ?par for more

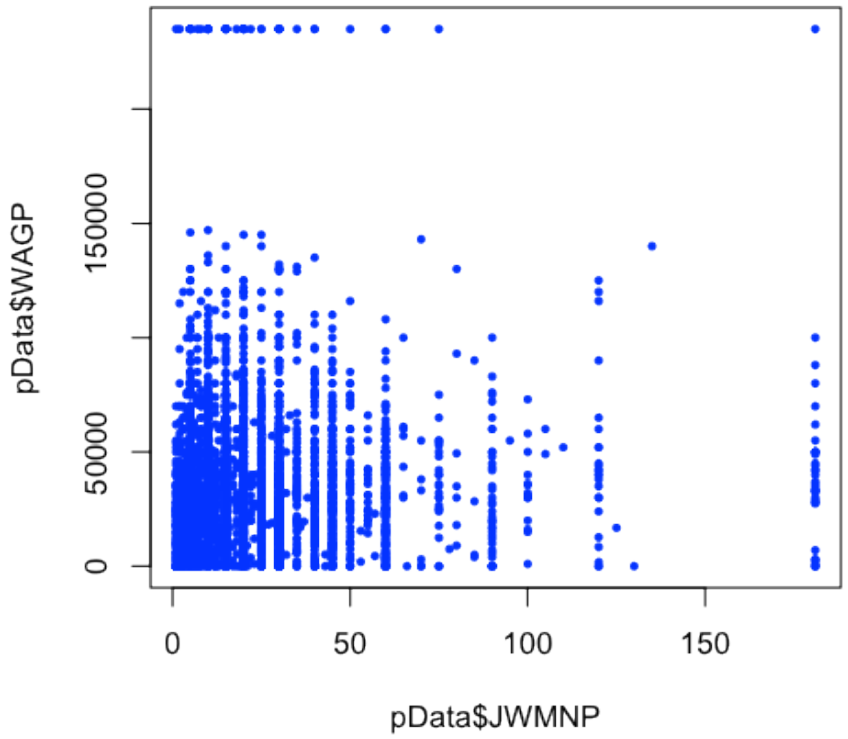
```
plot(pData$JWMNP,pData$WAGP,pch=19,col="blue")
```



6/23

Scatterplots - size matters

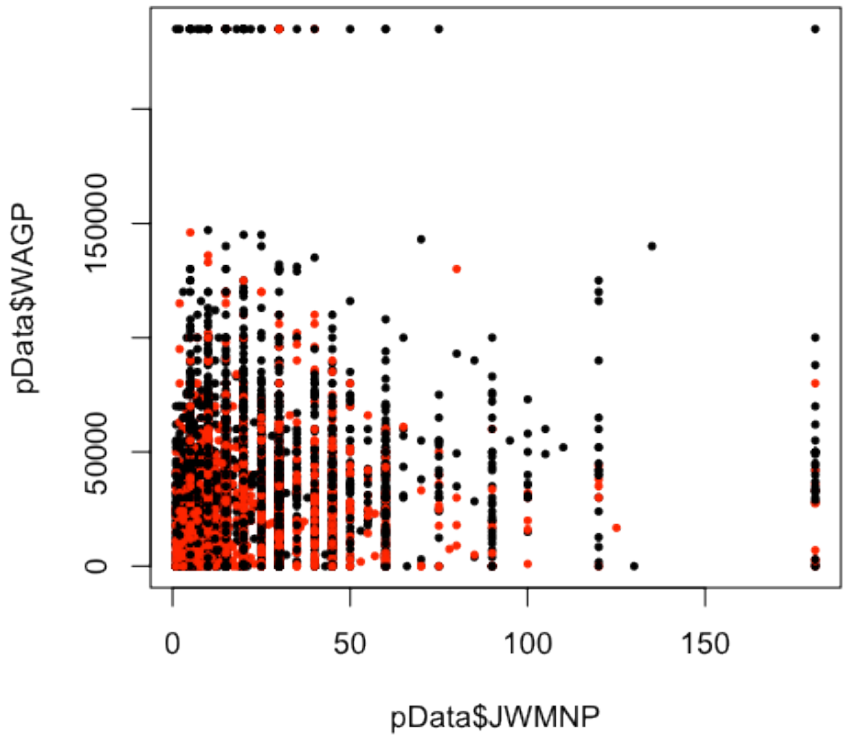
```
plot(pData$JWMNP,pData$WAGP,pch=19,col="blue",cex=0.5)
```



7/23

Scatterplots - using color

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



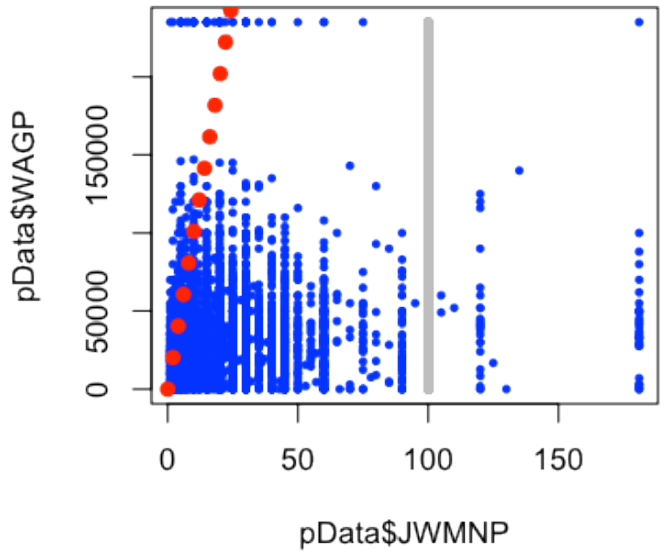
8/23

Scatterplots - using size

```
percentMaxAge <- pData$AGEP/max(pData$AGEP)
plot(pData$JWMNP,pData$WAGP,pch=19,col="blue",cex=percentMaxAge*0.5)
```

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",pch=19)
```



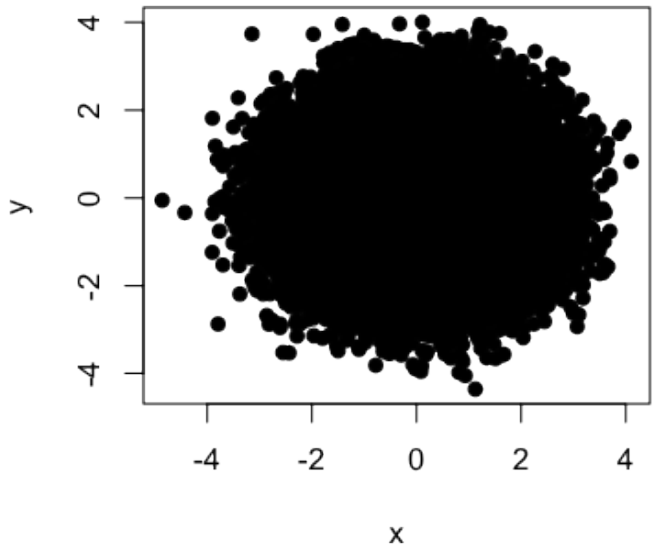
10/23

Scatterplots - numeric variables as factors

```
library(Hmisc)
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

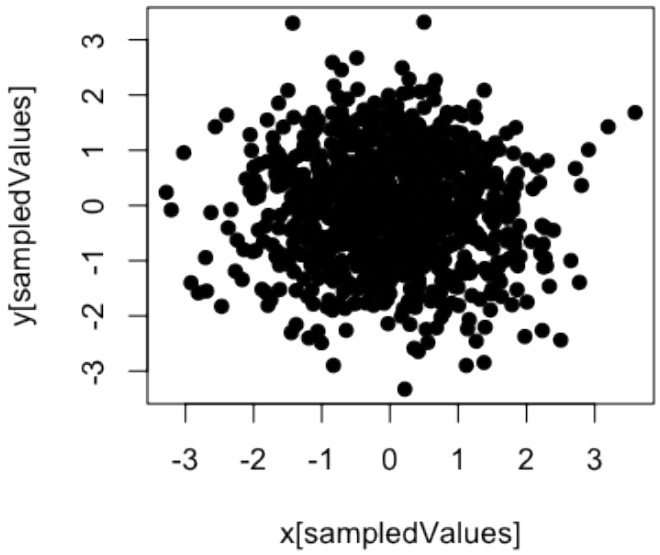
```
x <- rnorm(1e5)
y <- rnorm(1e5)
plot(x,y,pch=19)
```



12/23

If you have a lot of points - sampling

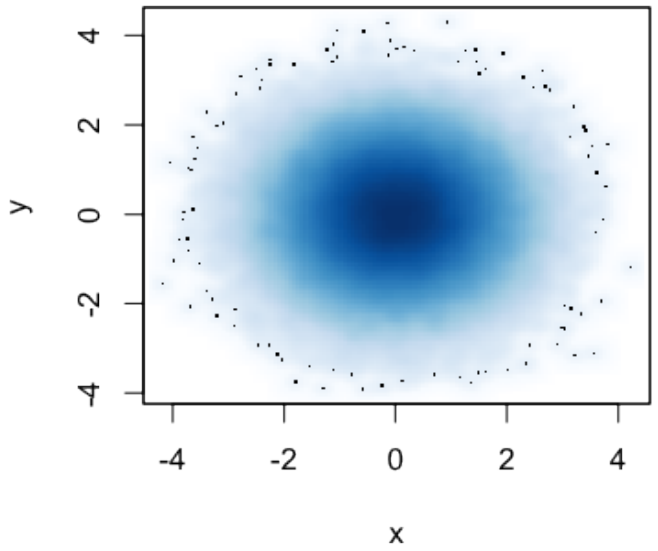
```
x <- rnorm(1e5)
y <- rnorm(1e5)
sampledValues <- sample(1:1e5,size=1000,replace=FALSE)
plot(x[sampledValues],y[sampledValues],pch=19)
```



13/23

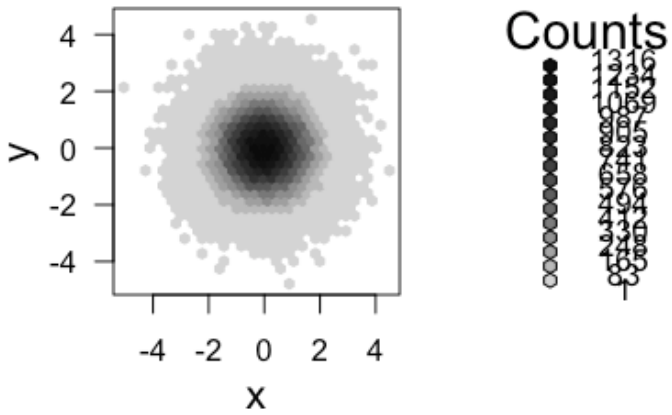
If you have a lot of points - smoothScatter

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



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

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

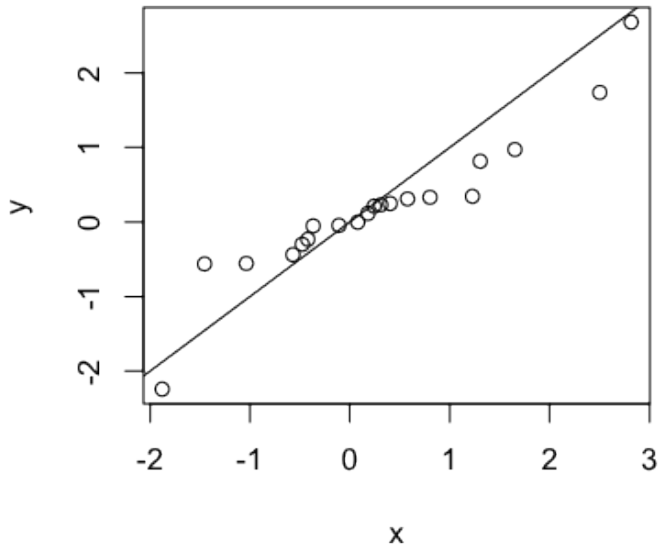


15/23

QQ-plots

- Important parameters: *x*,*y*

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

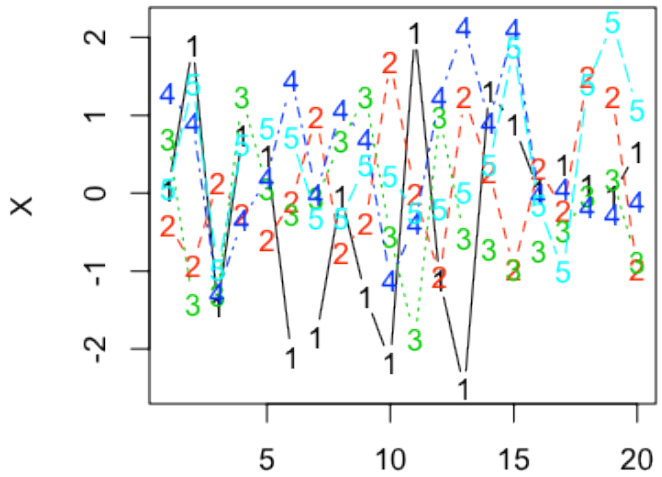


16/23

Matplot and spaghetti

- Important paramters: *x*, *y*, *lty*,*lwd*,*pch*,*col*

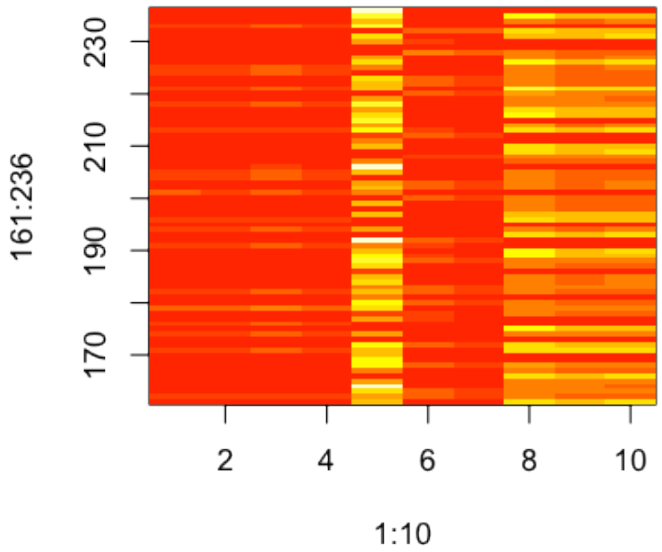
```
X <- matrix(rnorm(20*5),nrow=20)
matplot(X,type="b")
```



Heatmaps

- Important paramters: *x*,*y*,*z*,*col*

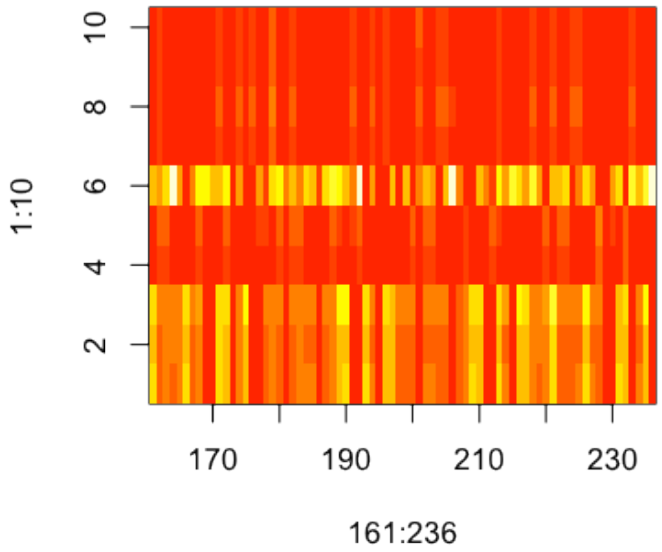
```
image(1:10,161:236,as.matrix(pData[1:10,161:236]))
```



18/23

Heatmaps - matching intuition

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

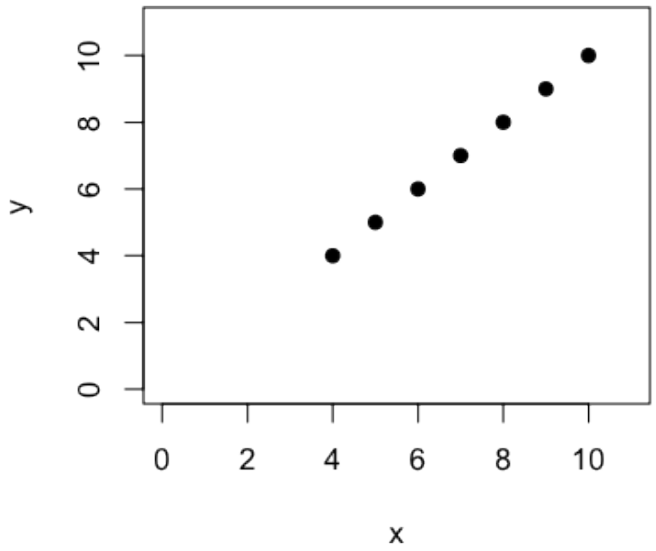


Maps - very basics

```
library(maps)
map("world")
lat <- runif(40,-180,180); lon <- runif(40,-90,90)
points(lat,lon,col="blue",pch=19)
```

Missing values and plots

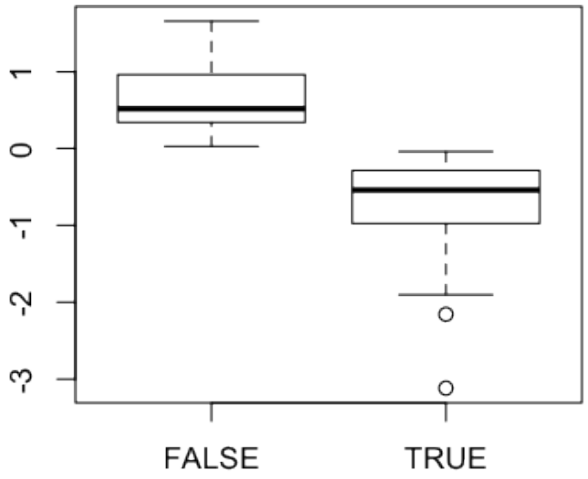
```
x <- c(NA,NA,NA,4,5,6,7,8,9,10)
y <- 1:10
plot(x,y,pch=19,xlim=c(0,11),ylim=c(0,11))
```



21/23

Missing values and plots

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



22/23

Further resources

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