M3

台灣人工智慧學校

資料處理方法

遺失值處理/資料轉換/異常檢測

吳漢銘
國立政治大學 統計學系

http://www.hmwu.idv.tw

資料處理方法- 大綱

- **主題1**
  - 具缺失值資料 (Missing Data)
  - 缺失機制 (Missingness Mechanism)
    - Missing by Design
    - Missing Completely at Random (MCAR)
    - Missing at Random (MAR)
    - Missing Not at Random (MNAR)
  - R Packages for Dealing With Missing Values: VIM, MICE
  - Visualizing the Pattern of Missing Data
  - Traditional Approaches to Handling Missing Data
  - Imputation Methods: KNN
  - Which Imputation Method?
- **主題2**
  - 為什麼要做資料轉換?
  - 常見的資料轉換方式
  - 對數轉換 (Log Transformation)
  - Box-Cox Transformation
  - 標準化 (Standardization)
  - 要使用哪一種資料轉換方式?
- **主題3**
  - 異常檢測的統計方法
  - Grubbs' Test for a Single Outlier

具缺失值資料 (Missing Data)

Missing data (missing values for certain variables for certain cases): **item non-response**.

When data are missing for a variable for all cases: **latent** or **unobserved**.

When data are missing for all variables for a given case: **unit non-response**.

	A	B	C	D	E	F	G
1	ID	C	Y	X1	X2	X3	X4
2	s1	1	78.3	69.6	74.3	NA	5.22
3	s2	2	77	69.9	72.54	NA	3.98
4	s3	3	72.2	65.7	69.74	NA	4.89
5	s4	1	33.4	NA	30.97	NA	21.54
6	s5	2	32.65	28.35	30.54	NA	9.82
7	s6	3	35.45	28.5	32.01	NA	19.81
8	s7	1	424	378	403.55	NA	12.98
9	s8	2	NA	NA	NA	NA	NA
10	s9	3	355	312.5	339.96	NA	14.14
11	s10	1	18.2	15.5	17.19	NA	13.93
12	s11	2	18.3	15.3	16.38	NA	6.92
13	s12	3	16.1	13.9	14.92	NA	10.15
14	s13	1	23.75	20.2	22.19	NA	32.81

缺失值的處理

- The missing values may give clues to systematic aspects of the problem.

- **How to deal with missing values:**
  - Use a global constant to fill the value will misguide the mining process. (例如: 缺考給0分; 影像訊號=前景-背景)
  - Use the attribute mean or median for all samples belonging to the same class as the given tuple.
  - **補值 (Missing value imputation)** (most popular)

缺失機制
Missingness Mechanism

- The presence of missing data can
  - effect the properties of the estimates
    (e.g. means, percentages, percentiles, variances, ratios, regression parameters, etc.).
  - affect inferences.
    (e.g., the properties of tests and confidence intervals.)

- **The missingness mechanism** (Little and Rubin, 1987)
  - The way in which the **probability of an item missing** depends on other observed or non-observed variables as well as on its own value.

- It helpful to classify missing values on the basis of the **stochastic mechanism** that produces them.

缺失機制
Missingness Mechanism

collected data

$$X = \{X_o, X_m\}$$

observed elements　　missing elements

The missingness indicator matrix R corresponds X,

and each element of R is 1 if the corresponding element of X is missing,

and 0 otherwise.

define the missingness mechanism as

the probability of R conditional on

the values of the observed and missing elements of X:

$$Pr(R|X_o, X_m)$$

Missing by Design
Missing Completely at Random

- **Missing by Design**
  - **Excluded** some participants from the analysis because they are not part of the population under investigation.
  - missingness codes: (i) refused to answer; (ii) answered don't know; (iii) had a valid skip or (iv) was skipped by an enumerator error.

- **Missing Completely at Random (MCAR)**
  - missingness is independent of their own unobserved values and the observed data.

$$Pr(R|X) = Pr(R)$$

  - *Example*: Miscoding or forgetting to log in answer.
  - **Imputation methods** rely on the missingness being of the MCAR type.

Missing at Random (MAR)
Missing Not at Random (MNAR)

- **Missing at Random (MAR)** $Pr(R|X) = Pr(R|X_o)$
  - missingness does not depend on their unobserved value but does dependent on the observed data.
  - ***Example 1***: male participants (observed data) are more likely to refuse to fill out the **depression survey**, but it does not depend on the level of their depression (unobserved value).
  - ***Example 2***: if men are more likely to tell you their weight than women, **weight** is MAR.
  - We can ignore missing data (= omit missing observations) if we have MAR or MCAR.

- **Missing Not at Random (MNAR)**
  - Missingness that depends on the missing value itself.
  - ***Example***: question about **income**, where the high rate of missing values (usually 20%~50%) is related to the value of the income itself (very high and very low values will not be answered).
  - MNAR data is a more serious issue. (not ignorable)

Some Notes

- Assuming data is **MCAR**, too much missing data can be a problem.
  - Usually a safe maximum threshold is 5% of the total for large datasets.
  - If missing data for a certain feature or sample is more than 5% then you probably should leave that feature or sample out.
- If some variable is missing almost 25% of the data points.
  - Consider either dropping it from the analysis or gather more measurements.
  - Keep the other variables are below the 5% threshold.
- For categorical variables, replacing categorical variables is usually not advisable.
- Some common practice include replacing missing categorical variables with the mode of the observed ones (questionable).

Other Special Values in R

- **NaN** : "not a number" which can arise for example when we try to compute the undeterminate 0/0.

```
> x <- c(1, 0, 10)
> x/x
[1]   1 NaN   1
> is.nan(x/x)
[1] FALSE  TRUE FALSE
```

- **Inf** which results from computations like 1/0.
- Using the functions **is.finite()** and **is.infinite()** we can determine whether a number is finite or not.

```
> 1/x
[1] 1.0 Inf 0.1
> is.finite(1/x)
[1]  TRUE FALSE  TRUE
>
> -10/x
[1]  -10 -Inf   -1
> is.infinite(-10/x)
[1] FALSE  TRUE FALSE
```

```
> exp(-Inf)
[1] 0
> 0/Inf
[1] 0
> Inf - Inf
[1] NaN
> Inf/Inf
[1] NaN
```

R Packages for Dealing With Missing Values

- **Amelia (Amelia II)**: A Program for Missing Data
- **hot.deck:** Multiple Hot-Deck Imputation
- **HotDeckImputation**: Hot Deck Imputation Methods for Missing Data
- **impute**: (Bioconductor) Imputation for Microarray Data
- **mi**: Missing Data Imputation and Model Checking
- **mice**: Multivariate Imputation by Chained Equations
- **missForest:** Nonparametric Missing Value Imputation using Random Forest
- **missMDA:** Handling Missing Values with Multivariate Data Analysis (e.g., imputePCA, imputeMCA,)
- **mitools**: Tools for Multiple Imputation of Missing Data
- **norm:** Analysis of Multivariate Normal Datasets with Missing Values
- **VIM**: Visualization and Imputation of Missing Values
- R packages support for missing values imputation.
  - **Hmisc:** Harrell Miscellaneous
  - **survey**: analysis of complex survey samples
  - **Zelig**: Everyone's Statistical Software
  - **rfImpute{randomForest}**: Imputations by randomForest
  - **imputation{rminer}**: Data Mining Classification and Regression Methods, Missing data imputation (e.g. substitution by value or hotdeck method).
  - **impute.svd{bcv}**: Cross-Validation for the SVD (Bi-Cross-Validation), Missing value imputation via a low-rank SVD approximation estimated by the EM algorithm.
  - **mlr**: Machine Learning in R provides several imputation methods. https://mlr-org.github.io/mlr-tutorial/release/html/index.html

Package "**imputation**" was removed from the CRAN. (Archived on 2014-01-14)

R Package: MICE

- `mice`: Multivariate Imputation by **Chained Equations** in R by Stef van Buuren.

- Imputing missing values on:
  - **Continuous data**: Predictive mean matching, Bayesian linear regression, Linear regression ignoring model error, Unconditional mean imputation etc.
  - **Binary data**: Logistic Regression, Logistic regression with bootstrap
  - **Categorical data** (More than 2 categories) - Polytomous logistic regression, Proportional odds model etc.
  - **Mixed data** (Can work for both Continuous and Categorical) - CART, Random Forest, Sample (Random sample from the observed values).

Source: http://www.listendata.com/2015/08/missing-imputation-with-mice-package-in.html

Generates Multivariate Imputations by Chained Equations (MICE)

```
mice(data, m = 5, method = vector("character", length = ncol(data)),
  predictorMatrix = (1 - diag(1, ncol(data))),
  visitSequence = (1:ncol(data))[apply(is.na(data), 2, any)],
  form = vector("character", length = ncol(data)),
  post = vector("character", length = ncol(data)), defaultMethod = c("pmm",
  "logreg", "polyreg", "polr"), maxit = 5, diagnostics = TRUE,
  printFlag = TRUE, seed = NA, imputationMethod = NULL,
  defaultImputationMethod = NULL, data.init = NULL, ...)
```

```
> methods(mice)
 [1] mice.impute.2l.norm      mice.impute.2l.pan       mice.impute.2lonly.mean
 [4] mice.impute.2lonly.norm  mice.impute.2lonly.pmm   mice.impute.cart
 [7] mice.impute.fastpmm      mice.impute.lda          mice.impute.logreg
[10] mice.impute
[13] mice.impute
[16] mice.impute
[19] mice.impute
[22] mice.impute
[25] mice.theme
see '?methods' f
> ? mice
```

Method	Description	Scale type	Default
pmm	Predictive mean matching	numeric	Y
norm	Bayesian linear regression	numeric	
norm.nob	Linear regression, non-Bayesian	numeric	
mean	Unconditional mean imputation	numeric	
2L.norm	Two-level linear model	numeric	
logreg	Logistic regression	factor, 2 levels	Y
polyreg	Multinomial logit model	factor, >2 levels	Y
polr	Ordered logit model	ordered, >2 levels	Y
lda	Linear discriminant analysis	factor	
sample	Random sample from the observed data	any	

Exploring Missing Data

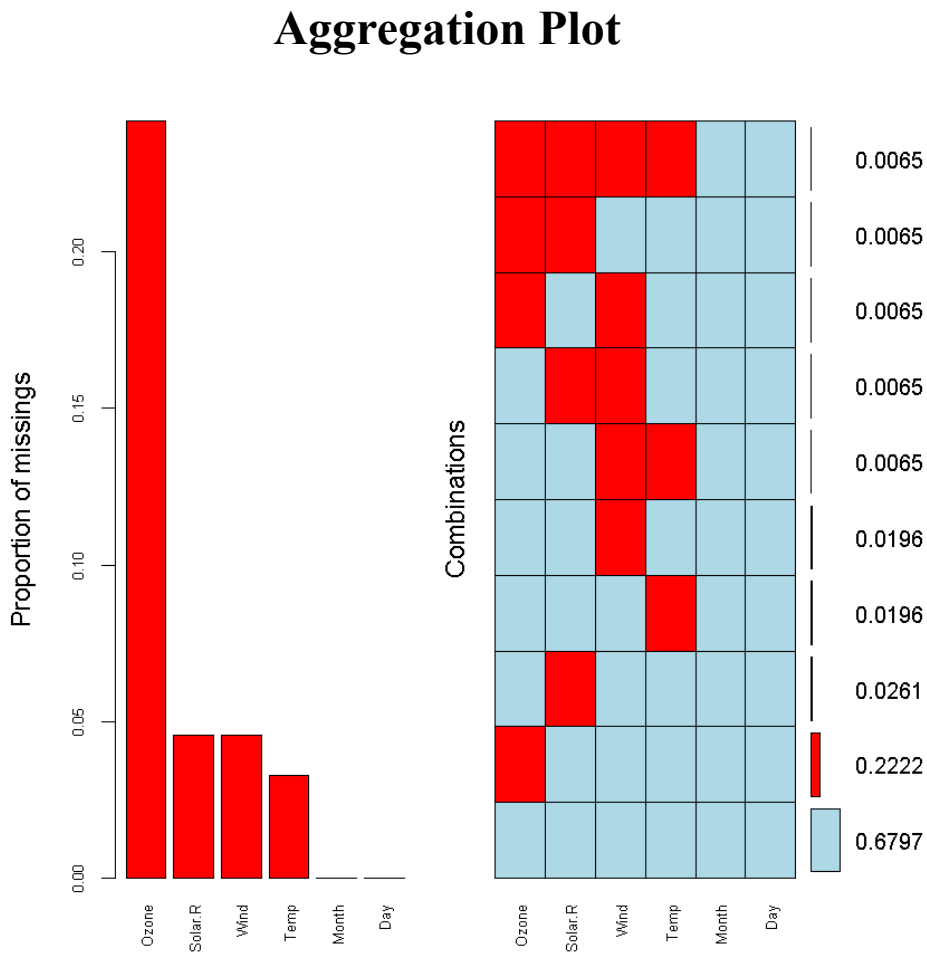
```
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
> dim(airquality)
[1] 153   6
> mydata <- airquality
> mydata[4:10, 3] <- rep(NA, 7)
> mydata[1:5, 4] <- NA
>
> # Use numerical variables as examples here.
> # Ozone is the variable with the most missing datapoints.
> summary(mydata)
     Ozone           Solar.R           Wind             Temp           Month            Day      
 Min.   :  1.00   Min.   :  7.0   Min.   : 1.700   Min.   :57.00   Min.   :5.000   Min.   : 1.0  
 1st Qu.: 18.00   1st Qu.:115.8   1st Qu.: 7.400   1st Qu.:73.00   1st Qu.:6.000   1st Qu.: 8.0  
 Median : 31.50   Median :205.0   Median : 9.700   Median :79.00   Median :7.000   Median :16.0  
 Mean   : 42.13   Mean   :185.9   Mean   : 9.806   Mean   :78.28   Mean   :6.993   Mean   :15.8  
 3rd Qu.: 63.25   3rd Qu.:258.8   3rd Qu.:11.500   3rd Qu.:85.00   3rd Qu.:8.000   3rd Qu.:23.0  
 Max.   :168.00   Max.   :334.0   Max.   :20.700   Max.   :97.00   Max.   :9.000   Max.   :31.0  
 NA's   :37       NA's   :7       NA's   :7        NA's   :5                                      
```

Sourec: http://www.r-bloggers.com/imputing-missing-data-with-r-mice-package/

Visualizing the Pattern of Missing Data

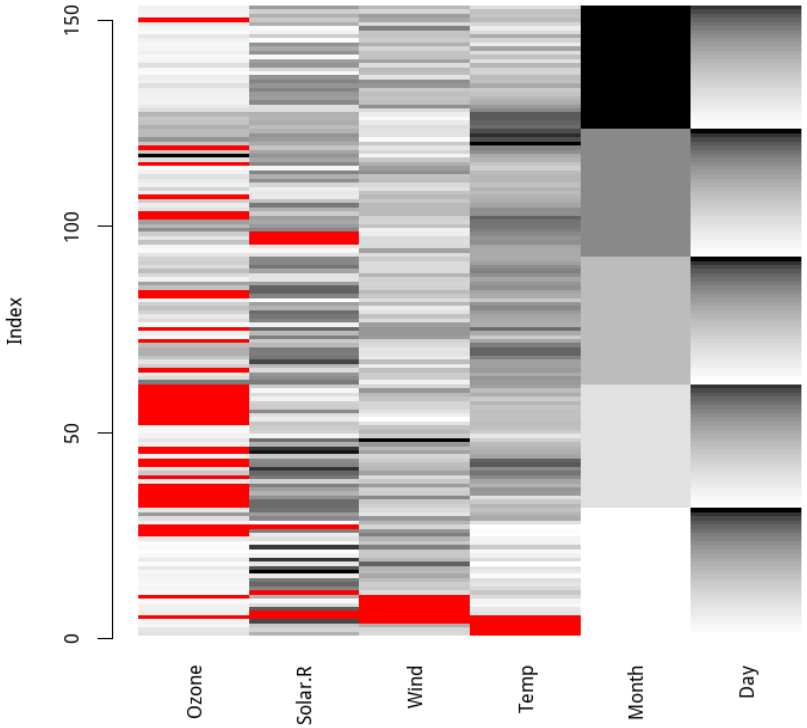
```
> library(mice)
> md.pattern(mydata)
    Month Day Temp Solar.R Wind Ozone
104     1   1    1       1    1     1  0
 34     1   1    1       1    1     0  1
  4     1   1    1       0    1     1  1
  3     1   1    1       1    0     1  1
  3     1   1    0       1    1     1  1
  1     1   1    1       0    1     0  2
  1     1   1    1       1    0     0  2
  1     1   1    1       0    0     1  2
  1     1   1    0       1    0     1  2
  1     1   1    0       0    0     0  4
        0   0    5       7    7    37 56
```

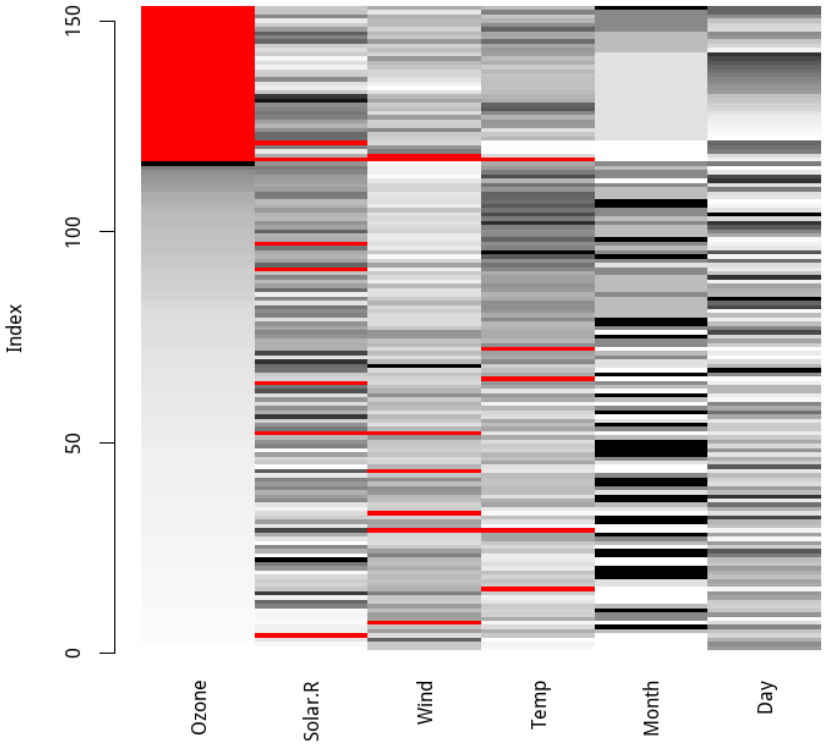
```
> library(VIM)
> mydata.aggrplot <- aggr(mydata,
col=c('lightblue','red'), numbers=TRUE,
prop = TRUE, sortVars=TRUE,
labels=names(mydata), cex.axis=.7, gap=3)

 Variables sorted by number of missings:
 Variable      Count
    Ozone 0.24183007
  Solar.R 0.04575163
     Wind 0.04575163
     Temp 0.03267974
    Month 0.00000000
      Day 0.00000000
```



Matrix Plot

```
> matrixplot(mydata)

Click in a column to sort by the corresponding variable.
To regain use of the VIM GUI and the R console, click outside the plot region.

Matrix plot sorted by variable 'Ozone'.
```





Number of Observations Per Patterns for All Pairs of Variables

V2	v	partial	complete
	x	all missing	partial
		x	v
		V1	

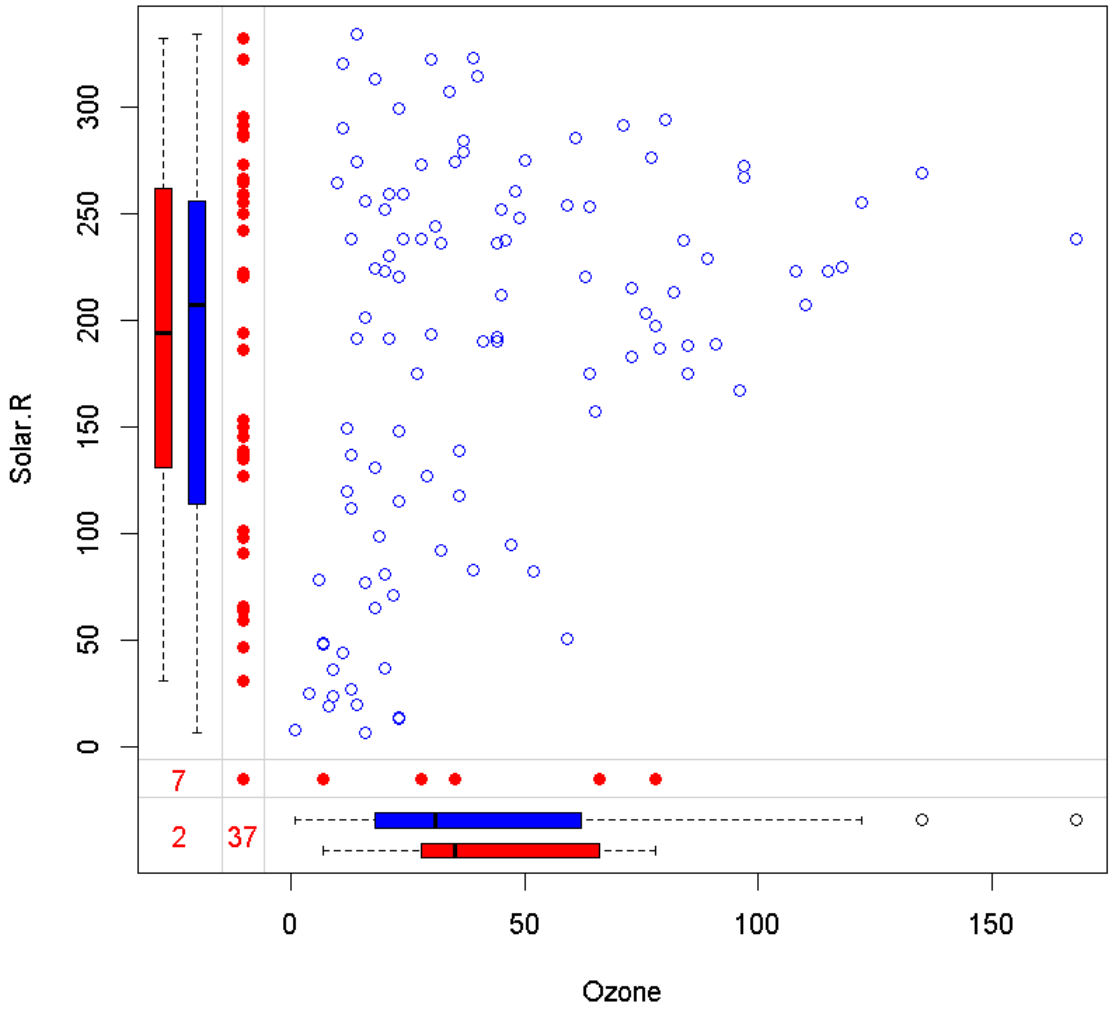
- `rr`: response-response, both variables are observed
- `rm`: response-missing, row observed, column missing
- `mr`: missing-response, row missing, column observed
- `mm`: missing-missing, both variables are missing

```
> md.pairs(mydata)
$rr
        Ozone Solar.R Wind Temp Month Day
Ozone     116     111  111  112   116 116
Solar.R   111     146  141  142   146 146
Wind      111     141  146  143   146 146
Temp      112     142  143  148   148 148
Month     116     146  146  148   153 153
Day       116     146  146  148   153 153

$rm
        Ozone Solar.R Wind Temp Month Day
Ozone       0       5    5    4     0   0
Solar.R    35       0    5    4     0   0
Wind       35       5    0    3     0   0
Temp       36       6    5    0     0   0
Month      37       7    7    5     0   0
Day        37       7    7    5     0   0
```

```
$mr
        Ozone Solar.R Wind Temp Month Day
Ozone       0      35   35   36    37  37
Solar.R     5       0    5    6     7   7
Wind        5       5    0    5     7   7
Temp        4       4    3    0     5   5
Month       0       0    0    0     0   0
Day         0       0    0    0     0   0

$mm
        Ozone Solar.R Wind Temp Month Day
Ozone      37       2    2    1     0   0
Solar.R     2       7    2    1     0   0
Wind        2       2    7    2     0   0
Temp        1       1    2    5     0   0
Month       0       0    0    0     0   0
Day         0       0    0    0     0   0
```

Marginplot

```
> marginplot(mydata[,c("Ozone", "Solar.R")], col = c("blue", "red"))
```



- The blue box plot located on the left and bottom margins shows the distribution of the non-missing datapoints.
- The red box plot on the left shows the distribution of Solar.R with Ozone missing.
- If our assumption of MCAR data is correct, then we expect the red and blue box plots to be very similar.

List-wise Deletion

- Also called the **complete case analysis**.
- The use of this method is only justified if the missing data generation mechanism is **MCAR**.

```
> mdata <- matrix(rnorm(15), nrow=5)
> mdata[sample(1:15, 4)] <- NA
> mdata <- as.data.frame(mdata)
> mdata
           V1         V2          V3
1 -0.62222501  1.0807983          NA
2  0.07124865  0.5216675 -0.08334454
3  1.70707399  0.1004917  0.88197789
4          NA -0.6595201 -0.08387860
5          NA  1.6138847          NA
> (x1 <- na.omit(mdata))
          V1        V2          V3
2 0.07124865 0.5216675 -0.08334454
3 1.70707399 0.1004917  0.88197789
> (x2 <- mdata[complete.cases(mdata),])
          V1        V2          V3
2 0.07124865 0.5216675 -0.08334454
3 1.70707399 0.1004917  0.88197789
```

```
> mdata[!complete.cases(mdata),]
         V1         V2         V3
1 -0.622225  1.0807983         NA
4        NA -0.6595201 -0.0838786
5        NA  1.6138847         NA
```

快速分析一下，得知資料大概狀況

Pairwise Deletion

- To compute a **covariance matrix**, each two cases will be used for which the values of both corresponding variables are available.
- This can result in covariance or correlation matrices which are not positive semi-definite, as well as NA entries if there are no complete pairs for the given pair of variables.

```
> mdata
           V1         V2          V3
1 -0.62222501  1.0807983          NA
2  0.07124865  0.5216675 -0.08334454
3  1.70707399  0.1004917  0.88197789
4          NA -0.6595201 -0.08387860
5          NA  1.6138847          NA
> cov(mdata)
   V1        V2 V3
V1 NA        NA NA
V2 NA 0.7694197 NA
V3 NA        NA NA
> cov(mdata, use = "all.obs")
Error in cov(mdata, use = "all.obs") :
missing observations in cov/cor
> cov(mdata, use = "complete.obs")
           V1          V2         V3
V1  1.3379623 -0.34448500  0.7895494
V2 -0.3444850  0.08869452 -0.2032852
V3  0.7895494 -0.20328521  0.4659237
```

```
> cov(mdata, use = "na.or.complete")
           V1          V2         V3
V1  1.3379623 -0.34448500  0.7895494
V2 -0.3444850  0.08869452 -0.2032852
V3  0.7895494 -0.20328521  0.4659237
> cov(mdata, use = "pairwise")
           V1          V2         V3
V1  1.4304107 -0.56002326 0.78954945
V2 -0.5600233  0.76941970 0.05468712
V3  0.7895494  0.05468712 0.31078774
```

Mean Substitution

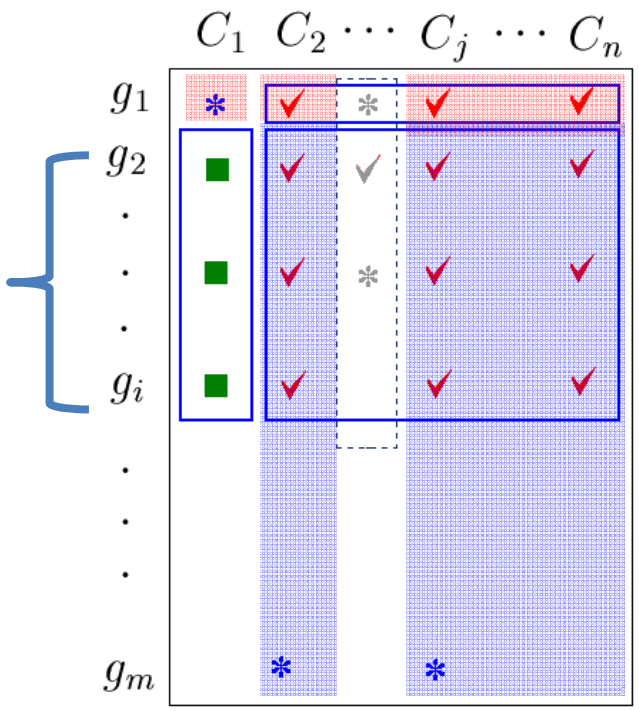
- A very simple but popular approach is to substitute means for the missing values.
- This method produces biased estimates and can severely distort the distribution of the variable in which missing values are substituted.
- Due to these **distributional problems**, it is often recommended to ignore missing values rather than impute values by mean substitution (Little and Rubin, 1989.)

```
mean.subst <- function(x) {
   x[is.na(x)] <- mean(x, na.rm = TRUE)
   x
}
```

```
> mdata
           V1         V2          V3
1 -0.62222501  1.0807983          NA
2  0.07124865  0.5216675 -0.08334454
3  1.70707399  0.1004917  0.88197789
4          NA -0.6595201 -0.08387860
5          NA  1.6138847          NA
> mdata.mip <- apply(mdata, 2, mean.subst)
> mdata.mip
               V1         V2          V3
[1,] -0.62222501  1.0807983  0.23825158
[2,]  0.07124865  0.5216675 -0.08334454
[3,]  1.70707399  0.1004917  0.88197789
[4,]  0.38536588 -0.6595201 -0.08387860
[5,]  0.38536588  1.6138847  0.23825158
```

K-Nearest Neighbour Imputation

- KNN imputation searches for the k-nearest observations (respective to the observation which has to be imputed) and replaces the missing value with the mean of the found *k* observations.
- It is recommended to use the (weighted) median instead of the arithmetic mean.
- **KNN** minimize data modeling assumptions and take advantage of the correlation structure of the data.



KNNimpute

Model:

$$\{g_{(k)}, k = 1, 2, \cdots, K\} = \underset{k}{\text{args}} \ \max_{i \in \mathrm{C}} \mathrm{Corr}(g_1, g_i)$$

$$\{g_{(k)}, k = 1, 2, \cdots, K\} = \underset{k}{\text{args}} \ \min_{i \in \mathrm{C}} \mathrm{Dist}(g_1, g_i)$$

C: Observed C_i's without missing values

Imputation:

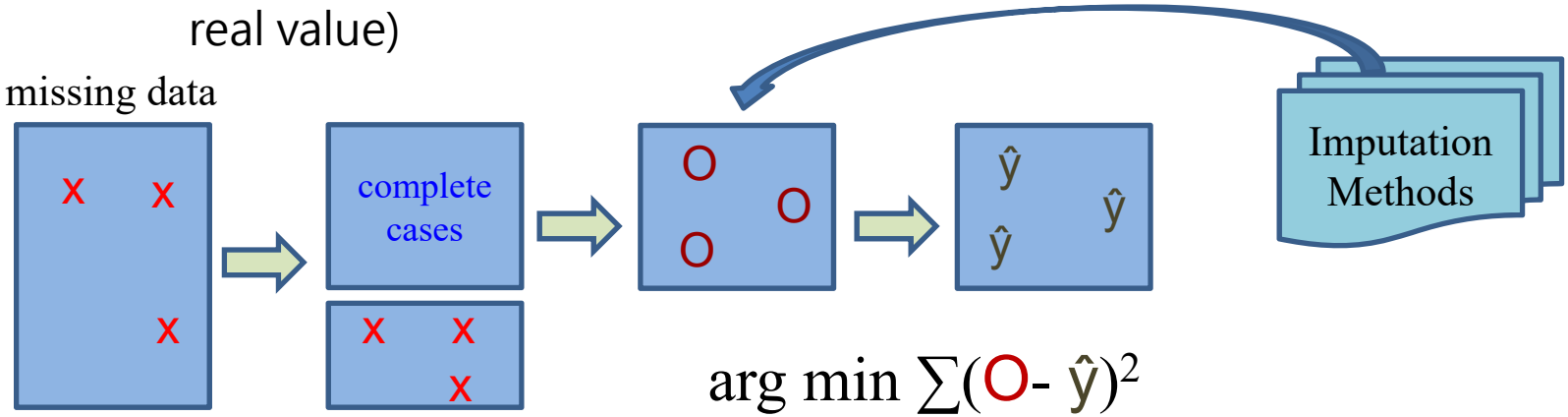
Average $\quad \widehat{C_1(g_1)} = \frac{1}{K} \sum_{k=1}^{K} C_1(g_k)$

Weighted Average $\quad \widehat{C_1(g_1)} = \frac{\sum_{k=1}^{K} w_k C_1(g_k)}{\sum_{k=1}^{K} w_k}$

$$w_k = \frac{1}{\sum_{j \in \mathrm{C}} [C_j(g_k) - C_1(g_1)]^2}$$

Which Imputation Method?

- KNN is the most widely-used.
- Characteristics of data that may affect choice of imputation method:
  - dimensionality.
  - percentage of values missing.
  - experimental design (time series, case/control, etc.)
  - patterns of correlation in data.
- Suggestion:
  - add (**same percentage**) artificial missing values to your (**complete cases**) data set.
  - impute them with various methods, see which is best (since you know the real value)



$$\arg\min \sum (O - \hat{y})^2$$

Classical (Numerical) Data Table

*j*th variable

*i*th subject
(*i*th sample)

UID	alpha0	alpha7	alpha14	alpha21	alpha28	alpha35	alpha42
YAR007C	-0.48	-0.42	0.87	0.92	0.67	-0.18	-0.35
YBL035C	-0.39	-0.58	1.08	1.21	0.52	-0.33	-0.58
YBR023C	0.87	0.25	-0.17	0.18	-0.13	-0.44	-0.13
YBR067C	1.57	1.03	1.22	0.31	0.16	-0.49	-1.02
YBR088C	-1.15	-0.86	1.21	1.62	1.12	0.16	-0.44
YBR278W	0.04	-0.12	0.31	0.16	0.17	-0.06	0.08
YCL055W	2.95	0.45	-0.4	-0.66	-0.59	-0.38	-0.76
YDL003W	-1.22	-0.74	1.34	1.5	0.63	0.29	-0.55
YDL055C	-0.73	-1.06	-0.79	-0.02	0.16	0.44	0.03
YDL102W	-0.58	-0.4	0.13	0.58	-0.09	0.02	-0.45
YDL164C	-0.5	-0.42	0.66	1.05	0.68	0.06	0.01
YDL197C	-0.86	-0.29	0.42	0.46	0.3	0.1	-0.63
YDL227C	-0.16	0.2877	0.17	-0.28	-0.02	-0.55	-0.04
YDR052C	-0.36	-0.03	-0.03	-0.08	-0.23	-0.25	-0.21
YDR097C	-0.72	-0.85	0.54	1.04	0.84	0.24	-0.64
YDR113C	-0.78	-0.52	0.26	0.2	0.48	0.48	0.27
YDR309C	0.6	-0.55	0.41	0.45	0.18	-0.66	-1.02
YDR356W	-0.2	-0.67	0.13	0.1	0.38	0.44	0.05
YER001W	-2.29	-0.635739	0.77	1.6	0.53	0.55	-0.38
YER070W	-1.46	-0.76	1.08	1.5	0.74	0.47	-0.7
YER095W	-0.57	0.42	1.03	1.35	0.64	0.42	-0.4
YGL163C	-0.11	0.13	0.41	0.6	0.23	0.31	0.19
YGL225W	-1.08	-0.99	-0.16	0.2	0.61	0.37	0.1
YGR109C	-1.79	0.9449	2.13	1.75	0.23	0.15	-0.66

transformation
for each row

transformation
for each column

transformation
for both rows
and columns

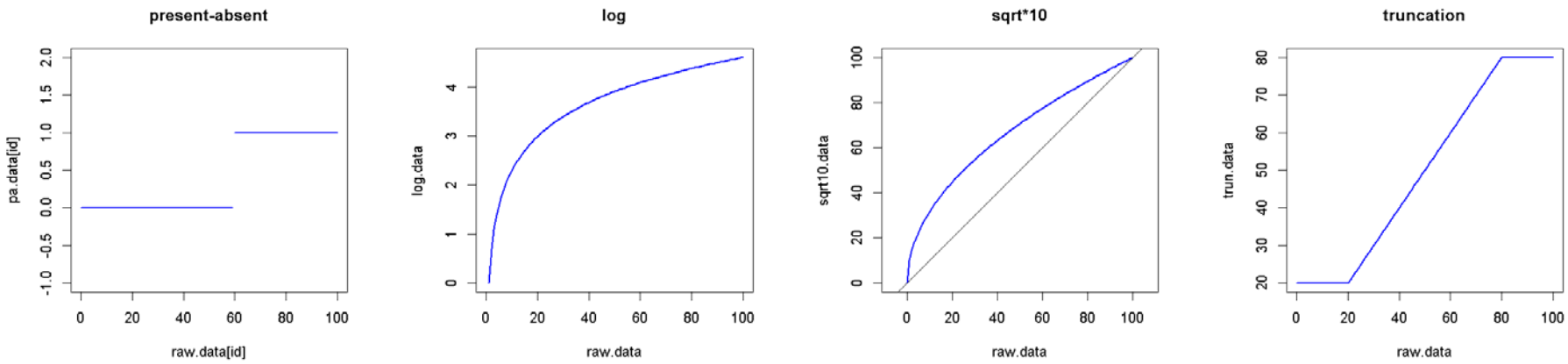
為什麼要做資料轉換?

- to make it more closely **the assumptions** of a statistical inference procedure,
- to make it **easier to visualize** (appearance of graphs),
- to improve **interpretability**,
- to make descriptors that have been measured in **different units comparable**,
- to make the relationships among **variables linear**,
- to modify the **weights** of the variables or objects (e.g. give the same length (or norm) to all object vectors)
- to **code** categorical variables into dummy binary variables.

常見的資料轉換方式

```
par(mfrow=c(1,4))
raw.data <- 0:100
pa.data <- ifelse(raw.data >= 60, 1, 0)
id <- which(pa.data==1)
plot(raw.data[id], pa.data[id], main="present-absent",
+ type="l", lwd=2, col="blue", ylim=c(-1, 2), xlim=c(0, 100))
points(raw.data[-id], pa.data[-id], type="l", lwd=2, col="blue")

log.data <- log(raw.data)
plot(raw.data, log.data, main="log", type="l", lwd=2, col="blue")

sqrt10.data <- sqrt(raw.data)*10
plot(raw.data, sqrt10.data, main="sqrt*10", type="l", lwd=2, col="blue", asp=1)
abline(a=0, b=1)

trun.data <- ifelse(raw.data >= 80, 80, ifelse(raw.data < 20, 20, raw.data))
plot(raw.data, trun.data, main="truncation", type="l", lwd=2, col="blue")
```

NOTE:
```
apply(raw.data.matrix, 2, log)
apply(raw.data.matrix, 2, function(x) sqrt(x)*10)
apply(raw.data.matrix, 2, myfun)
```



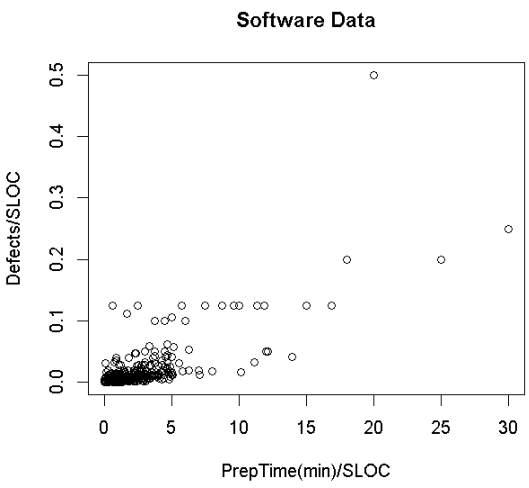
範例: Software Inspection Data

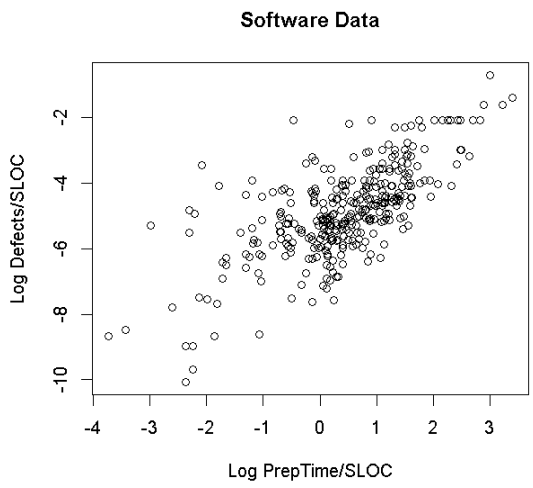
- The data were collected in response to efforts for process improvement in software testing by code inspection.
- The variables are normalized by the size of the inspection (the number of pages or SLOC (single lines of code)):
  - the preparation time in minutes (**prepage, prepsloc**),
  - the total work hours in minutes for the meeting (**mtgsloc**),
  - and the number of defects found (**defpage**, **defsloc**).

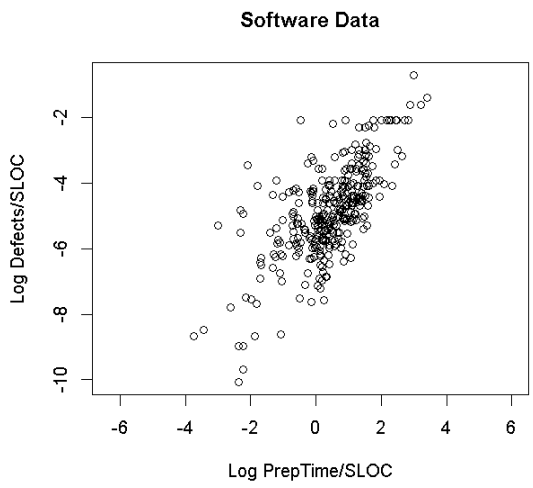
```
> library('R.matlab')
> data <- readMat("software.mat")
> print(data)
...
> str(data)
List of 5
 $ prepsloc: num [1:426, 1] 0.485 0.54 0.54 0.311 0.438 ...
 $ defsloc : num [1:426, 1] 0.005 0.002 0.002 0.00328 0.00278 ...
 $ mtgsloc : num [1:426, 1] 0.075 0.06 0.06 0.2787 0.0417 ...
 $ prepage : num [1:491, 1] 6.15 1.47 1.47 5.06 5.06 ...
 $ defpage : num [1:491, 1] 0.0385 0.0267 0.0133 0.0128 0.0385 ...
```

- **Interested in**: understanding the relationship between the inspection time and the number of defects found.

對數轉換 (Log Transformation)







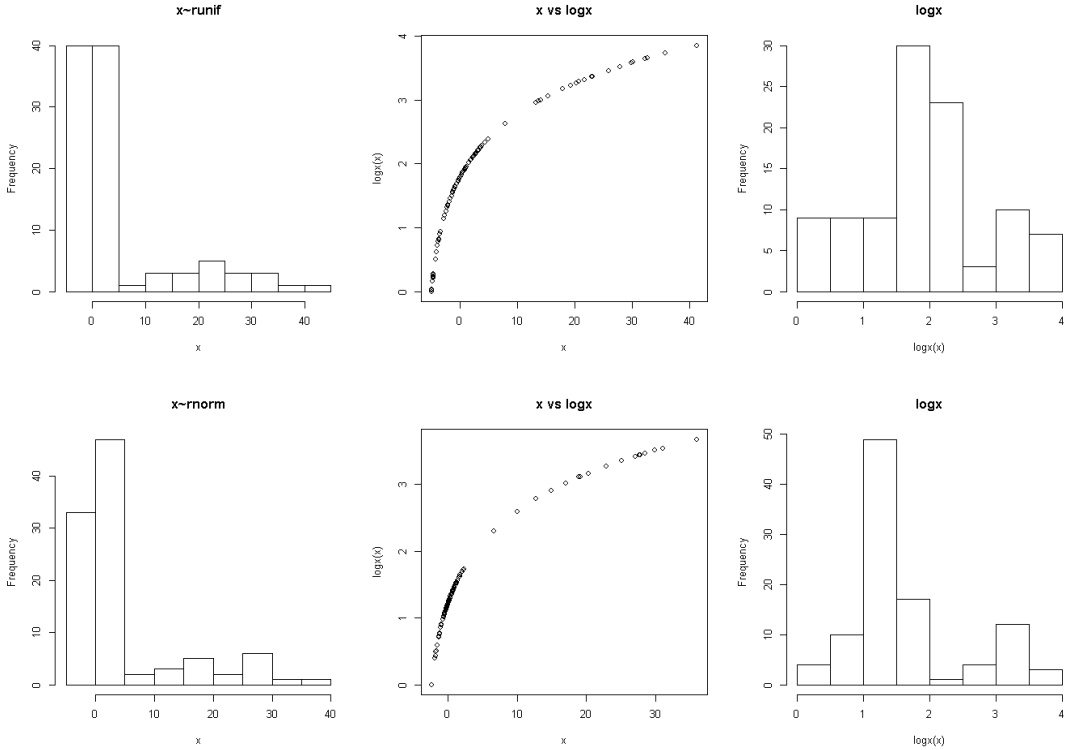

```
plot(data$prepsloc, data$defsloc, xlab="PrepTime(min)/SLOC", ylab="Defects/SLOC",
main="Software Data")

plot(log(data$prepsloc), log(data$defsloc), xlab="Log PrepTime/SLOC",
ylab="Log Defects/SLOC", main="Software Data")

plot(log(data$prepsloc), log(data$defsloc), xlab="Log PrepTime/SLOC",
ylab="Log Defects/SLOC", main="Software Data", asp=1)
```

對數轉換: How to Handle Negative Data Values?

Solution 1: Translate, then Transform

- log(x + 1 - min(x))

```
logx <- function(x){
   log(x + 1 - min(x))
}

x <- runif(80, min = -5, max = 5)
# x <- rnorm(80)
x <- c(x, rnorm(20, mean=20, sd=10))
par(mfrow=c(1, 3))
hist(x, main="x~runif")
plot(x, logx(x), main="x vs logx")
hist(logx(x), main="logx")
```



Solution 2: Use Missing Values

- A criticism of the previous method is that some practicing statisticians don't like to add an arbitrary constant to the data.
- They argue that a better way to handle negative values is to use missing values for the logarithm of a nonpositive number.

Box-Cox Transformations

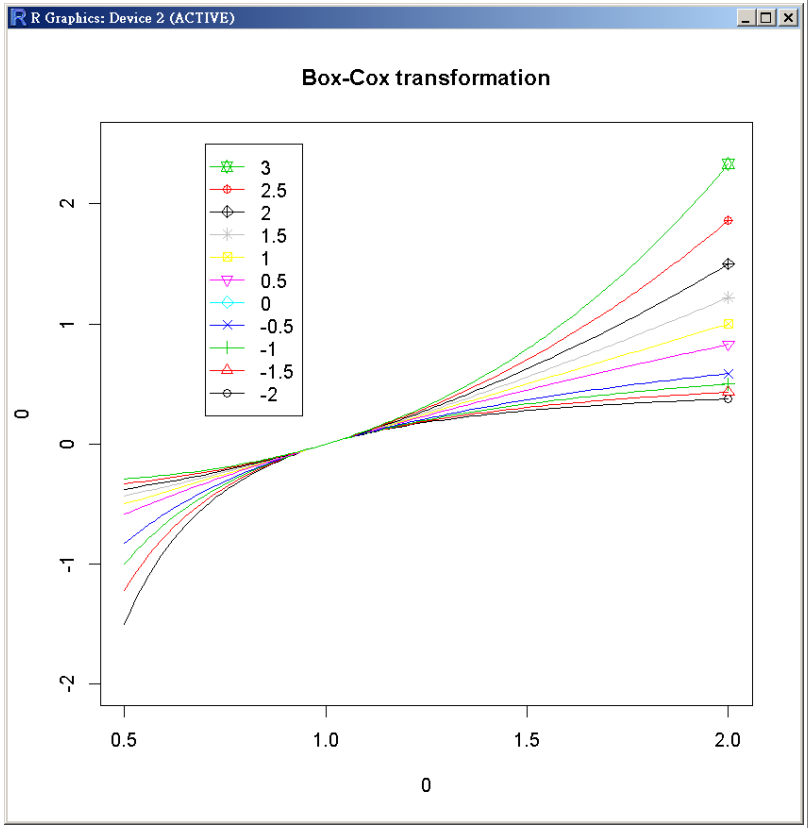
$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0; \\ \log y, & \text{if } \lambda = 0. \end{cases}$$

Box and Cox(1964)

- The aim of the Box-Cox transformations is to ensure the usual assumptions for Linear Model hold.

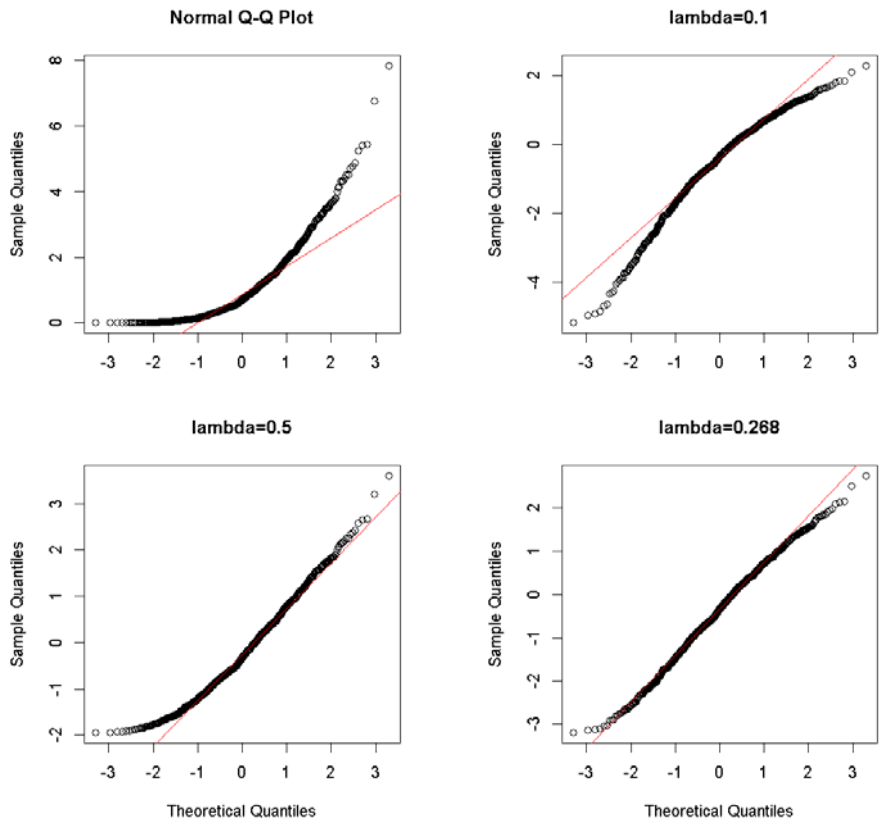
$$\boldsymbol{y} \sim \mathrm{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n)$$

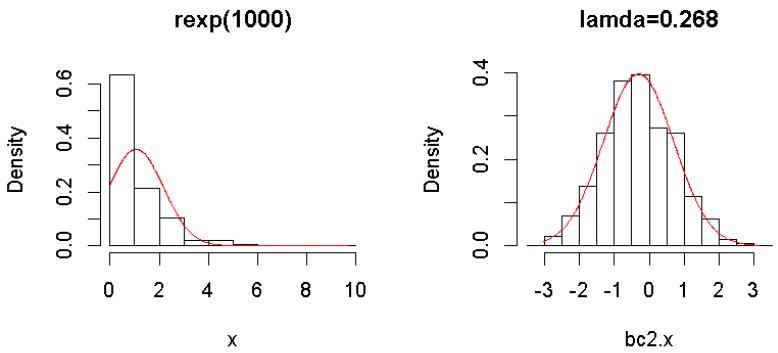
```
x <- seq(0.5, 2, length.out=100)
bc <- function(y, lambda){
    (y^lambda -1)/lambda
}
lambda <- seq(-2, 3, 0.5)
plot(0, 0, type="n", xlim=c(0.5, 2),
     ylim=c(-2, 2.5), main="Box-Cox transformation")
for(i in 1:length(lambda)){
   points(x, bc(x, lambda[i]), type="l", col=i)
   points(2, bc(2, lambda[i]), col=i, pch=i)
}
legend(0.7, 2.5, legend=as.character(rev(lambda)),
       lty=1, pch=length(lambda):1,
       col=length(lambda):1)
```



Box-Cox Transformations

```
x <- rexp(1000)
bc <- function(y, lambda){
    (y^lambda -1)/lambda
}
qqnorm(x); qqline(x, col="red")

bc1.x <- bc(x, 0.1)
qqnorm(bc1.x, main="lambda=0.1")
qqline(bc1.x, col="red")
bc3.x <- bc(x, 0.5)
qqnorm(bc3.x, main="lambda=0.5")
qqline(bc3.x, col="red")

bc2.x <- bc(x, 0.268)
qqnorm(bc2.x, main="lambda=0.268")
qqline(bc2.x, col="red")

hist(x, main="rexp(1000)")
hist(bc2.x, main="lambda=0.268")
```





$$\left(\Phi^{-1}\left(\frac{i-0.5}{n}\right), x_{(i)}\right), \qquad \text{for } i = 1, 2, \ldots, n,$$

Source: Box-Cox Transformations: An Overview, Pengfei Li, Department of Statistics, University of Connecticut, Apr 11, 2005

Modified Box-Cox Transformations

Manly(1971)

$$y(\lambda) = \begin{cases} \frac{e^{\lambda y}-1}{\lambda}, & \text{if } \lambda \neq 0; \\ y, & \text{if } \lambda = 0. \end{cases}$$

Negative y's could be allowed. The transformation was reported to be successful in transform unimodal skewed distribution into normal distribution, but is not quite useful for **bimodal** or **U-shaped distribution**.

John and Draper(1980) "Modulus Transformation"

$$y(\lambda) = \begin{cases} \text{Sign}(y)\frac{(|y|+1)^{\lambda}-1}{\lambda}, & \text{if } \lambda \neq 0; \\ \text{Sign}(y)\log(|y|+1), & \text{if } \lambda = 0, \end{cases}$$

$$\text{Sign}(y) = \begin{cases} 1, & \text{if } y \geq 0; \\ -1, & \text{if } y < 0. \end{cases}$$

Bickel and Doksum(1981)

$$y(\lambda) = \frac{|y|^{\lambda}\text{Sign}(y) - 1}{\lambda}, \quad \text{for } \lambda > 0,$$

Yeo and Johnson(2000)

$$y(\lambda) = \begin{cases} \frac{(y+1)^{\lambda}-1}{\lambda}, & \text{if } \lambda \neq 0,\ y \geq 0; \\ \log(y+1), & \text{if } \lambda = 0,\ y \geq 0; \\ \frac{(1-y)^{2-\lambda}-1}{\lambda-2}, & \text{if } \lambda \neq 2,\ y < 0; \\ -\log(1-y), & \text{if } \lambda = 2,\ y < 0. \end{cases}$$

Source: Box-Cox Transformations: An Overview, Pengfei Li, Department of Statistics, University of Connecticut, Apr 11, 2005

標準化 (Standardization)

- Standardization: (called z-score): the new variate Z will have a mean of zero and a variance of one. (also called centering and scaling.)

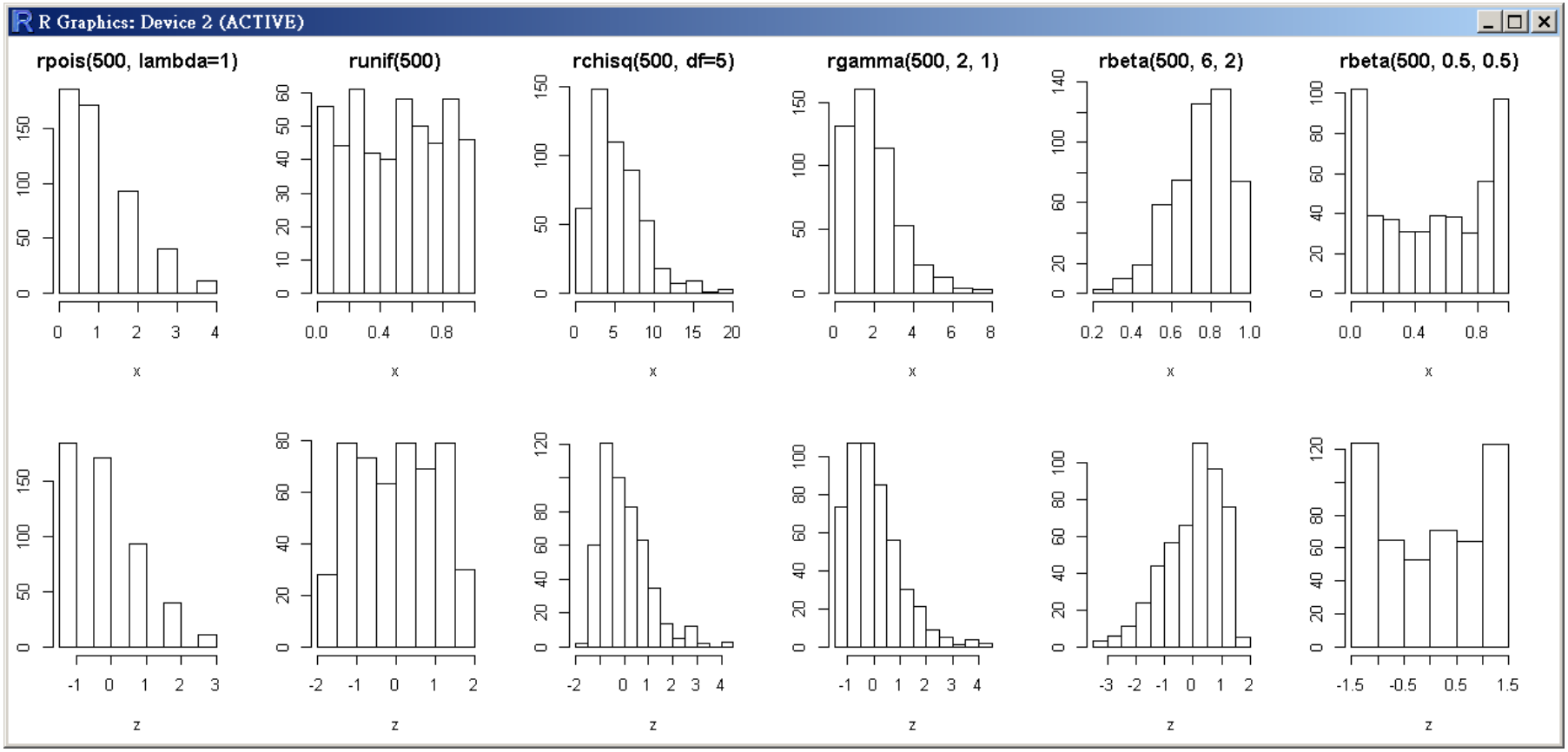
$$z_i = \frac{x_i - \bar{x}}{s}$$

- If the variables are measurements along a **different scale** or if the standard deviations for the variables are different from one another, then one variable might **dominate** the distance (or some other similar calculation) used in the analysis.

- Standardization is useful for comparing variables expressed in different units.

標準化 (Standardization)

Standardization makes no difference to the shape of a distribution.




```
x <- rpois(500, lambda=1)
hist(x, main="rpois(500, lambda=1)"); z <- scale(x); hist(z, main="")
```

範例: Standardization

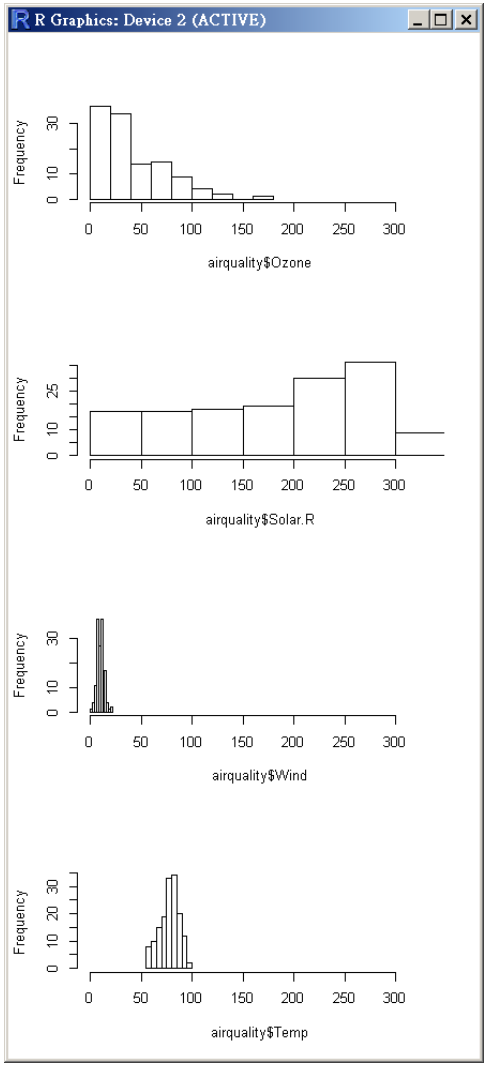
airquality {datasets}

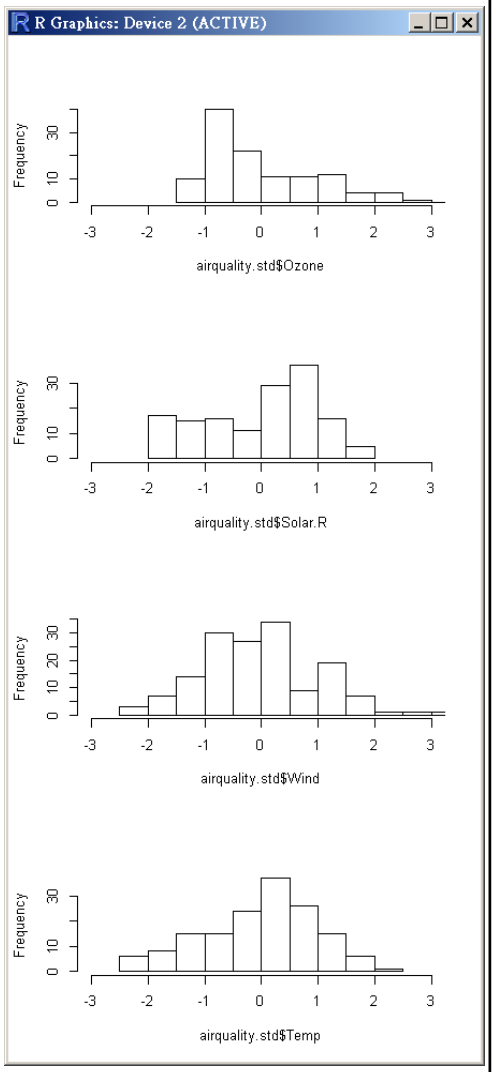
New York Air Quality Measurements: Daily air quality measurements in New York, May to September 1973.

A data frame with 154 observations on 6 variables.

[1] Ozone: Ozone (ppb)
[2] Solar.R: Solar R (lang)
[3] Wind: Wind (mph)
[4] Temp: Temperature (degrees F)
[5] Month: Month (1--12)
[6] Day: Day of month (1--31)

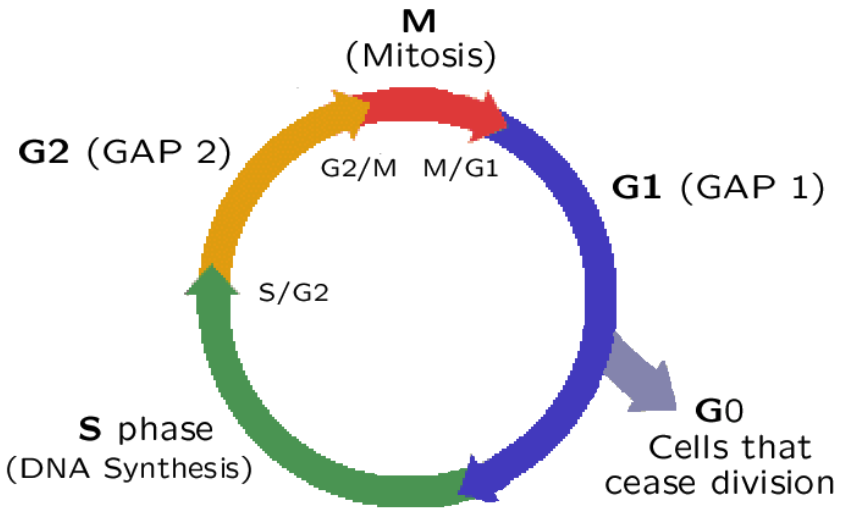
```
> head(airquality )
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
> r <- range(airquality[,1:4], na.rm = T)
> hist(airquality$Ozone , xlim = r)
> hist(airquality$Solar.R, xlim = r)
> hist(airquality$Wind, xlim = r)
> hist(airquality$Temp, xlim = r)
>
> airquality.std <- as.data.frame(
apply(airquality, 2, scale))
> r.std <- c(-3, 3)
> hist(airquality.std$Ozone, xlim = r.std)
> hist(airquality.std$Solar.R, xlim = r.std)
> hist(airquality.std$Wind, xlim = r.std)
> hist(airquality.std$Temp, xlim = r.std)
```

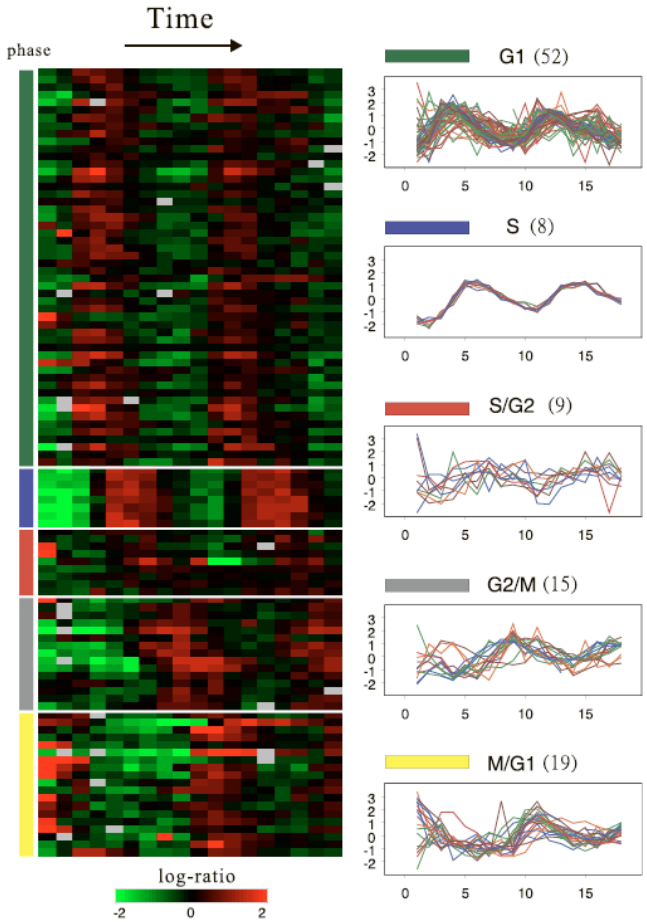




範例: Microarray Data of Yeast Cell Cycle

- Lu and Wu (2010)
  - Time course data: every 7 minutes and totally 18 time points.
  - Known genes: there are 103 cell cycle-regulated genes by traditional method in G1, S, S/G2, G2/M, or M/G1. (Remove NA' s: 79.)

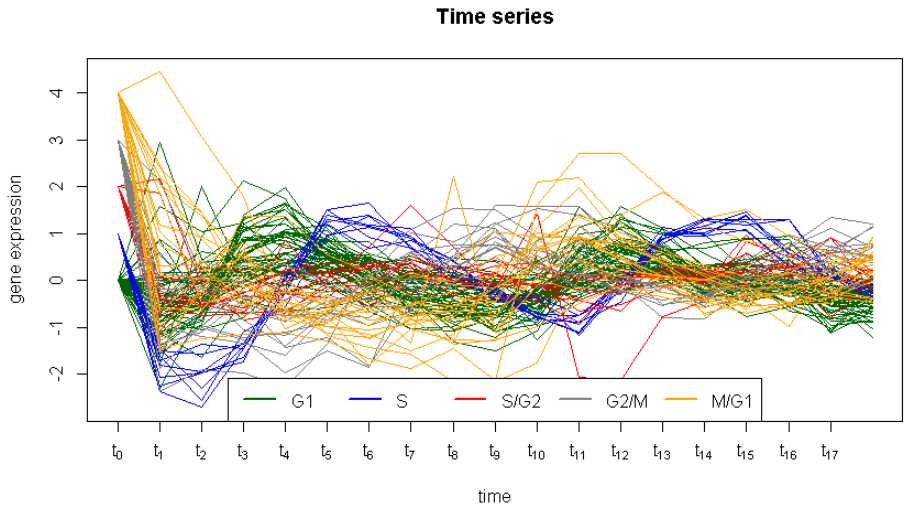


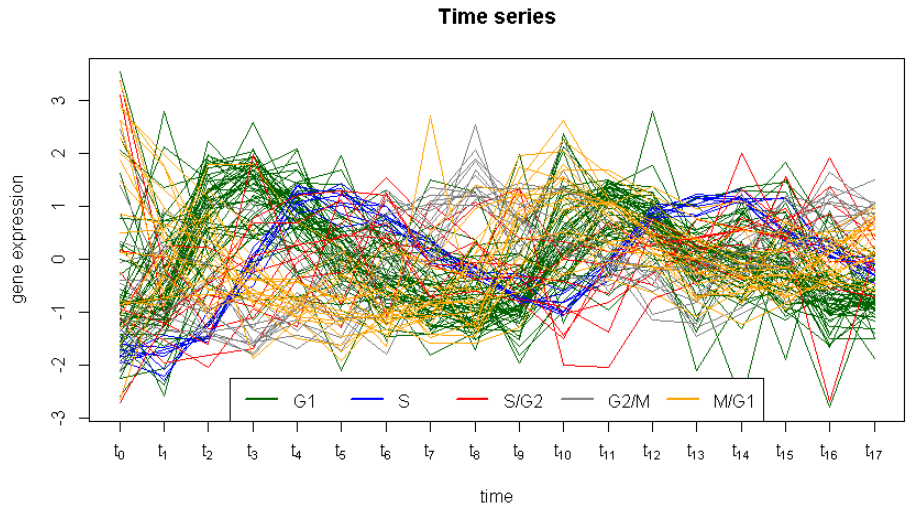


See also: Using R to draw a Heatmap from Microarray Data
http://www2.warwick.ac.uk/fac/sci/moac/people/students/peter_cock/r/heatmap/

Standardization in Time Series Microarray Gene Expression Experiments

```
cell.raw <- read.table("trad_alpha103.txt", row.names=1, header=T)
head(cell.raw)
cell.xdata <- t(scale(t(cell.raw[,2:19]), center=T, scale=T))
y.C <-  as.integer(cell.raw[,1])
table(y.C)
no.cluster <- length(unique(y.C))
cellcycle.color <- c("darkgreen", "blue", "red", "gray50", "orange")
p <- ncol(cell.raw) -1
ycolors <- cellcycle.color[y.C+1]
my.pch <- c(1:no.cluster)[y.C+1]
phase <- c("G1", "S", "S/G2", "G2/M", "M/G1")
matplot(t(cell.xdata), pch = 1:p, lty=1, type = "l", ylab="gene expression",
            col=ycolors, xlab="time", main="Time series", xaxt="n")
time.label <- parse(text=paste("t[",0:p,"]",sep=""))
axis(1, 1:(p+1), time.label)
legend("bottom", legend=phase, col=cellcycle.color, lty=1, horiz = T, lwd=2)
```





The data map for 103 cell cycle-regulated genes and the plots of time courses for each phase. Each expression profile is normalized as mean equals zero and variance 1.

範例: Crab Data

crabs {MASS}

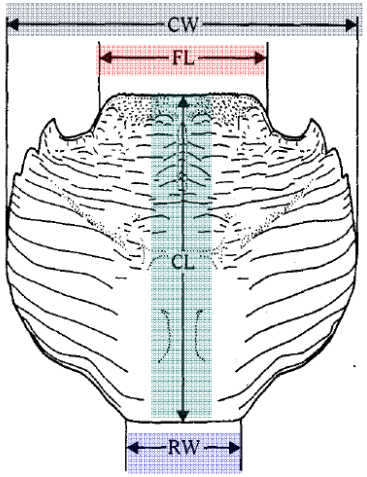
Morphological Measurements on Leptograpsus Crabs

Description: The crabs data frame has 200 rows and 8 columns, describing 5 morphological measurements on **50 crabs each of two colour forms and both sexes**, of the species Leptograpsus variegatus (紫岩蟹) collected at Fremantle, W. Australia.

This data frame contains the following columns:

sp: species - "B" or "O" for blue or orange.
sex: “M” or “F” for male or female
index: 1:50 within each of the four groups.
FL: carapace frontal lobe (lip) size (mm).
RW: carapace rear width (mm).
CL: carapace length (mm).
CW: carapace width (mm).
BD: body depth (mm).

```
> library(MASS)
> data(crabs)
```

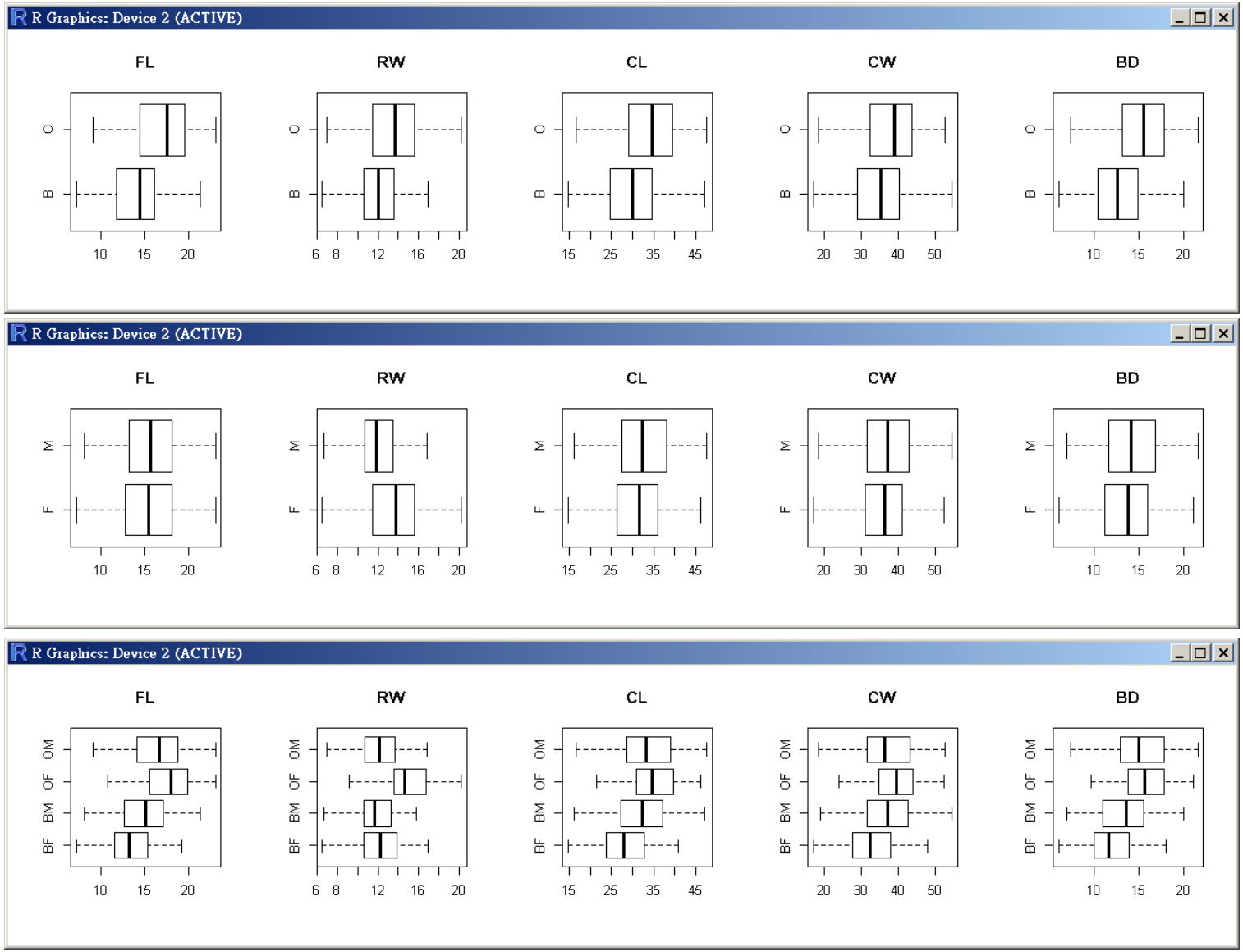


Aust. J. Zool. 1974, 22, 417-25

http://www.qm.qld.gov.au/Find+out+about/Animals+of+Queensland/Crustaceans/Common+marine+crustaceans/Crabs/Purple+Swift-footed+Shore+Crab#.VhPWYiurFhs

範例: Crab Data

```
boxplot(crabs$FL~crabs$sp, main="FL", horizontal=T)
```

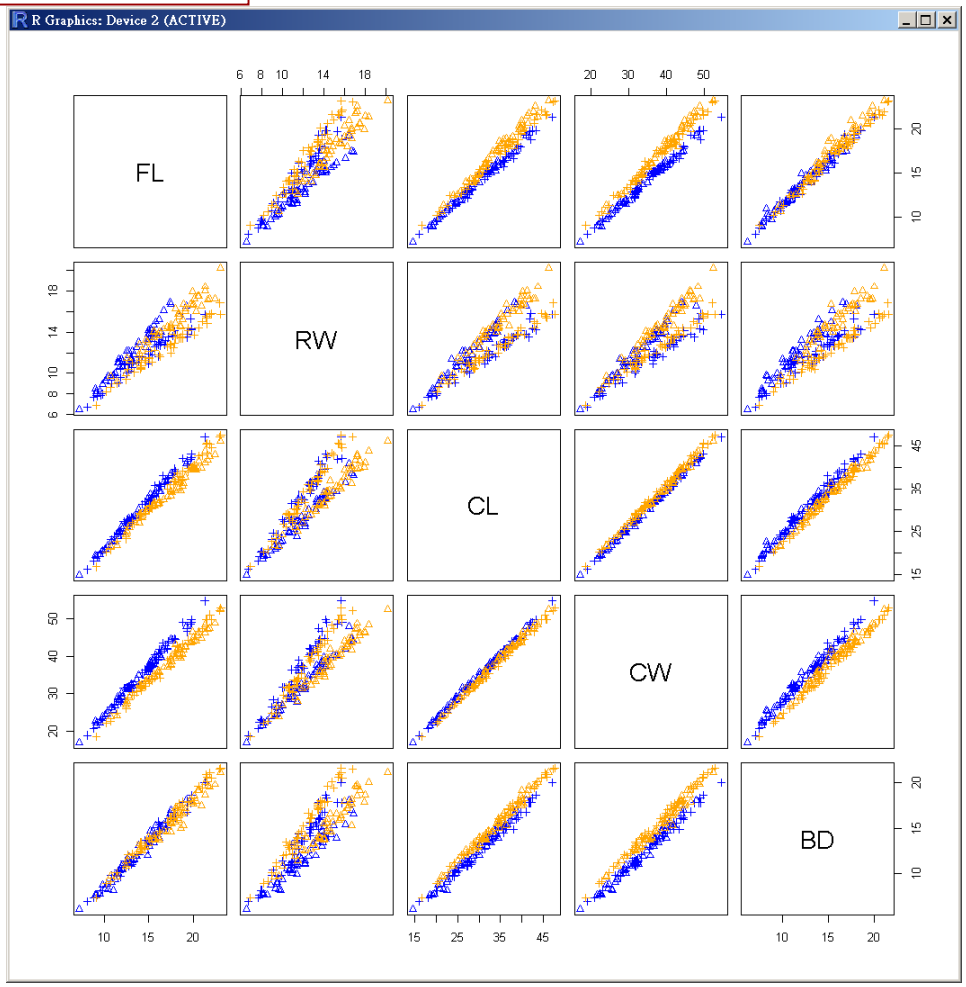


範例: Crab Data

```
# tri: F,  cross: M
pairs(crabs[,4:8],
pch=as.integer(crabs$sex)+1,
col=c("blue","orange")[as.integer(crabs$sp)])
```

- The analysis of ratios of body measurements is deeply ingrained in the taxonomic literature.
- Whether for plants or animals, certain ratios are commonly indicated in identification keys, diagnoses, and descriptions.

(Hannes Baur and Christoph Leuenberger, Analysis of Ratios in Multivariate Morphometry, Systematic Biology 60(6), 813-825.)

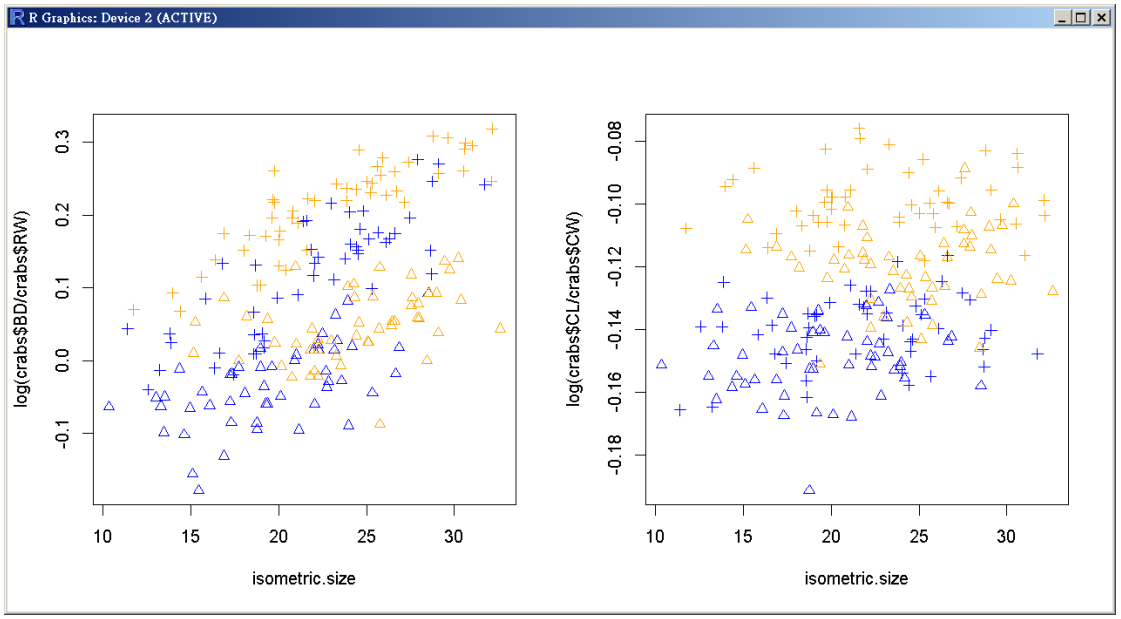


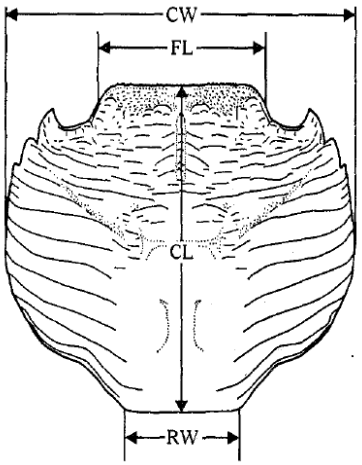
範例: Crab Data

- The use of **ratios of measurements** (i.e., of body proportions), has a long tradition and is deeply ingrained in morphometric taxonomy.

Three size vectors have been commonly proposed in the literature:

(a) isometric size
(the arithmetic mean of x),
(b) allometric size,
(c) shape-uncorrelated size.

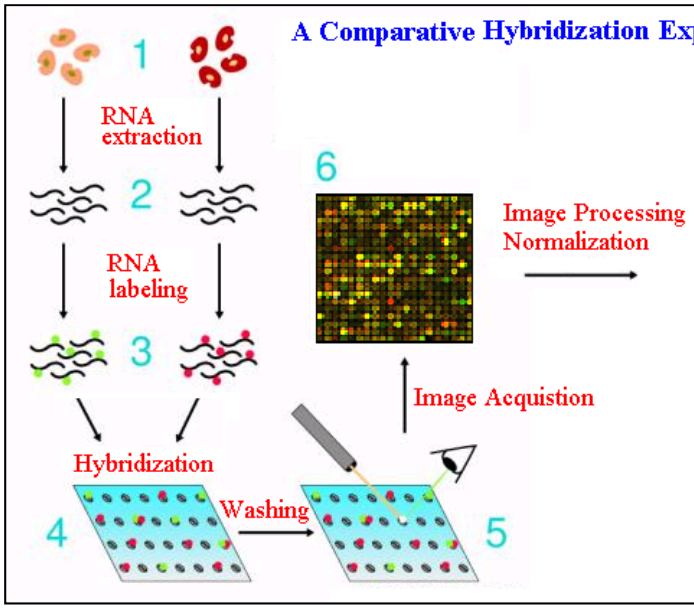





```
par(mfrow=c(1,2))
mp <- as.integer(crabs$sex)+1
mc <- c("blue","orange")[as.integer(crabs$sp)]
isometric.size <- apply(crabs[,4:8], 1, mean)
plot(isometric.size,  log(crabs$BD/crabs$RW), pch=mp, col=mc)
plot(isometric.size, log(crabs$CL/crabs$CW), pch=mp, col=mc)
```

範例: cDNA Microarray Gene Expression Data

微陣列資料統計分析 Statistical Microarray Data Analysis
http://www.hmwu.idv.tw/index.php/mada



cDNA Microarray Data

	A	B	C	D
1	UNIQID	Gene Name Description	Array 1	Array 2
2	588029	588029:Hs.79:ACY1	0.645	0.375
3	190929	190929:Hs.247565:RHO	0.615	0.210
4	246550	246550:Hs.293548	0.585	0.665
5	32553	32553:Hs.101248	0.825	0.230
6	446172	446172:	0.570	0.495
7	417978	417978:Hs.268874	0.495	1.835
...				
12000	366879	366879:Hs.169341	1.835	0.300

Matrix of genes (rows) and samples (columns)

Why Normalization?

Non-biological factor can contribute to the variability of data, in order to reliably compare data from multiple probe arrays, differences of non-biological origin must be minimized. (Remove the systematic bias in the data).

- Within-Array Normalization
- Between-Array Normalization
- Paired-slides Normalization
- ...

要使用哪一種資料轉換方式?

- Use a transformation that other researchers **commonly use in your field**.
- Guidance for how data should be transformed, or whether a transformation should be applied at all, should come from the particular statistical analysis to be performed.
- The main criterions in choosing a transformation:
  - what works with the data?
  - what makes physical (biological, economic, whatever) sense.
- If you have a **large** number of observations, compare the effects of different transformations on the normality and the homoscedasticity of the variable.

http://www.biostathandbook.com/transformation.html

異常檢測的統計方法 (Outliers Detection in R)

- **Graphical techniques**: index plot, Boxplot side-by-side, scatterplot, heatmap and so on.
- **R packages:**
  - **`outliers`**: Tests for outliers
    - A collection of some tests commonly used for identifying outliers.
    https://cran.r-project.org/web/packages/outliers/index.html
    - Grubbs' test (Grubbs 1969 and Stefansky 1972) is used to detect outliers in a univariate data set. It is based on the assumption of normality. That is, you should first verify that your data can be reasonably approximated by a normal distribution before applying the Grubbs' test.
  - **`extRemes`**: Extreme Value Analysis.
  - **`in2extRemes`**: Into the extRemes Package, GUI to some of the functions in the package extRemes. (http://www.assessment.ucar.edu/toolkit/)
  - **`extremevalues`**: Univariate Outlier Detection
  - **Extreme Value Analysis(EVA) packages in R: `evd, evdbayes, evir, fExtremes, lmom, SpatialExtremes, texmex, extRemes, ismev, texmex, ismev`**

- **Robust approaches to data with outliers**
  - Robustify the classical algorithm by replacing the sample mean vector and covariance matrix with the robust location and scatter estimators.

See also: Chapter 7, Outlier Detection, RDataMining-book-2015

Outliers Detection in R

- Outliers detection in R:
  https://statsandr.com/blog/outliers-detection-in-r/

- **Descriptive statistics**:
  Minimum and maximum, Histogram, Boxplot, Percentiles

Hampel filter

Another method, known as Hampel filter, consists of considering as outliers the values outside the interval (I) formed by the median, plus or minus 3 median absolute deviations (MAD):[1]

$$I = [median - 3 \cdot MAD; median + 3 \cdot MAD]$$

- **Statistical tests**: Grubbs's test, Dixon's test, Rosner's test

Grubbs' Test for a Single Outlier

- **Assumption**: the data (without any outliers) are approximately normally distributed.

- **Hypothesis**:
  - H0: There are no outliers in the data set
  - Ha: There is exactly one outlier in the data set

- **Test Statistic**: ESD (extreme studentized deviate)

$$ESD = max_{i=1,\ldots,n} \frac{|X_i - \bar{X}|}{s}$$

- **Critical Region**: For the two-sided test, the hypothesis of no outliers is rejected if

$$ESD > \frac{n-1}{\sqrt{n}} \sqrt{\frac{t^2}{n-2+t^2}}$$

where t is short for $t_{n-2,p}$ and $p = 1 - \alpha/(2n)$.

Grubbs, Frank (February 1969), Procedures for Detecting Outlying Observations in Samples, Technometrics, 11(1), pp. 1-21.

Robust Statistical Methods

CRAN Task View: Robust Statistical Methods
https://cran.r-project.org/web/views/Robust.html

- Robust Location and Scatter Estimators
  - Median, MAD (median of the absolute deviations from the median)
  - M-estimator (Huber, 1964; Maronna, 1976)
  - Stahel-Donoho estimator (Stahel, 1981; Donoho, 1982)
  - MVE (minimum volume ellipsoid), MCD (minimum covariance determinant) (Rousseeuw, 1983, 1984, 1985)
  - S-estimator (Davis, 1987)
  - Depth weighted and maximum depth estimators (Zuo, Cui and He, 2004)

MVE (minimum volume ellipsoid)

- Affine equivariant with high breakdown points.
- The existing efficient algorithm for computation.
- Readily available implementations.
- Ability to Identify extreme values.

$$\text{Outlier values} \sim 2 \times \sqrt{\chi^2_{0.975,p}} + N(0,1)$$

```
> qchisq(0.975,5)
[1] 12.83250
```