B01-2

R資料處理方法(II)

遺失值處理
離群值處理(異常檢測)
資料轉換

吳漢銘
國立政治大學 統計學系

http://www.hmwu.idv.tw

資料處理方法 (II)- 大綱

- 主題1: 遺失值處理
  - 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, Regression, MICE
  - 哪一種補值方法較好?
- 主題2: 離群值處理 (異常檢測)
  - 異常檢測的統計方法
  - Grubbs' Test for a Single Outlier
  - Robust Methods
- 主題3: 資料轉換
  - 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?

具遺失(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

遺失值的處理

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

- **如何處理遺失值:**
  - 不處理，換分析演算法。
  - 刪除法。
  - 用一全域值做填補: 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)
  - 用平均或中位數等統計量做填補: Use the attribute mean or median for all samples belonging to the same class as the given tuple.
  - 補值法 (Missing value imputation) (most popular)

遺失機制 (Missingness Mechanism)

- 若資料出現遺失值:
  - 計算及演算法無法進行。
  - 影響估計量的性質。
    (e.g. means, percentages, percentiles, variances, ratios, regression parameters, etc.).
  - 影響統計推論。
    (e.g., the properties of tests and confidence intervals.)

- 遺失機制 (**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

- 依設計產生的遺失 (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.

- 完全隨機遺失 (Missing Completely at Random, MCAR)
  - missingness is independent of their own unobserved values and the observed data.

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

  - 例: 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)

- **隨機遺失 (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.
  - *例 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).
  - *例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.

- **非隨機遺失 (Missing Not at Random, MNAR)**
  - Missingness that depends on the missing value itself.
  - *例*: 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)

一些注意事項

- 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.
- 類別變數的補值(categorical variable): replacing categorical variables is usually not advisable.
  - Some common practice include replacing missing categorical variables with the mode of the observed ones (questionable).
- 我的資料有需要做補值嗎?
- 補值後的資料不可改變「原資料結構」!
- 常聽到:「資料補值後，分類演算法的正確率提昇了」?!

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)

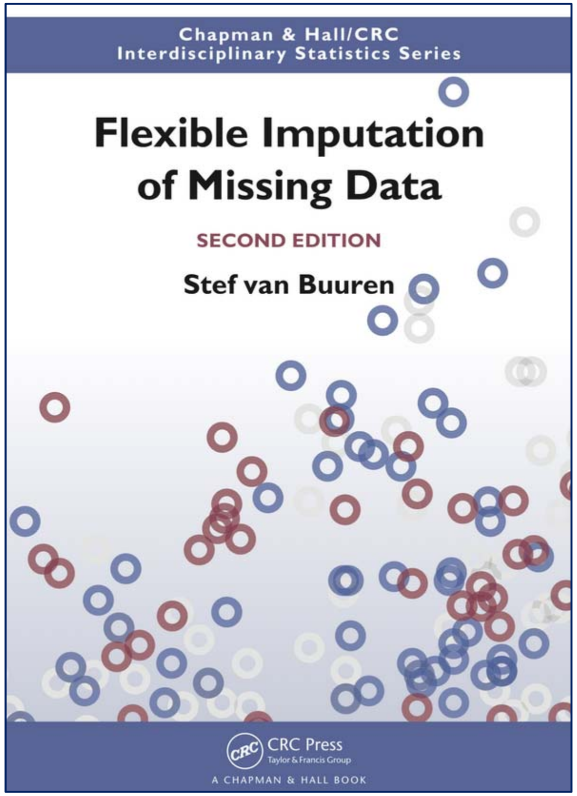
R Package: mice

- `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).

電子書

Flexible Imputation of Missing Data



https://stefvanbuuren.name/fimd

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

探索具遺失值資料 (Exploring Missing Data)

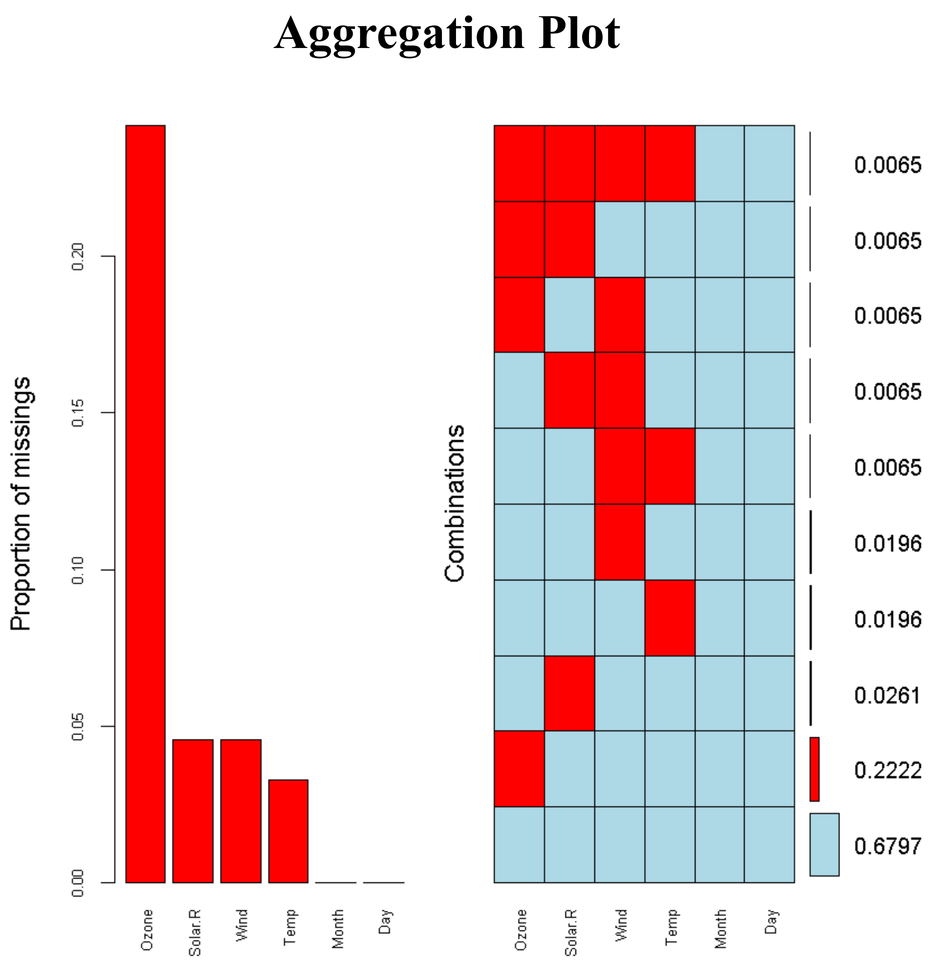
```
> head(airquality)
  Ozone Solar.R Wind Temp Month Day
1    41     190  7.4   67     5   1
2    36     118  8.0   72     5   2
3    12     149 12.6   74     5   3
4    18     313 11.5   62     5   4
5    NA      NA 14.3   56     5   5
6    28      NA 14.9   66     5   6
> dim(airquality)
[1] 153   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

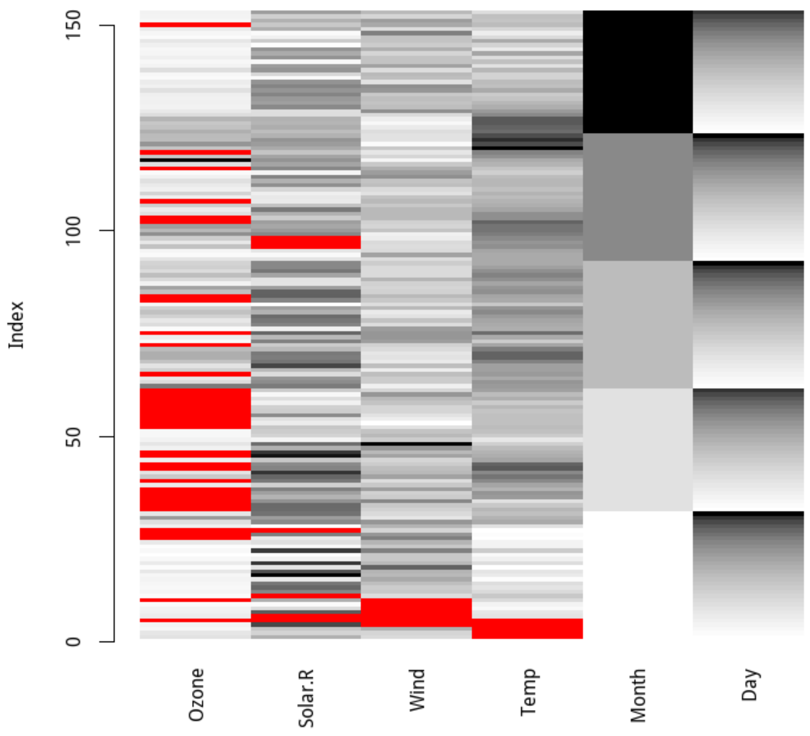
```
> library(mice)
> md.pattern(mydata)
    Month Day Temp Solar.R Wind Ozone
104     1   1    1       1    1     1  0
 34     1   1    1       1    1     0  1
  4     1   1    1       0    1     1  1
  3     1   1    1       1    0     1  1
  3     1   1    0       1    1     1  1
  1     1   1   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

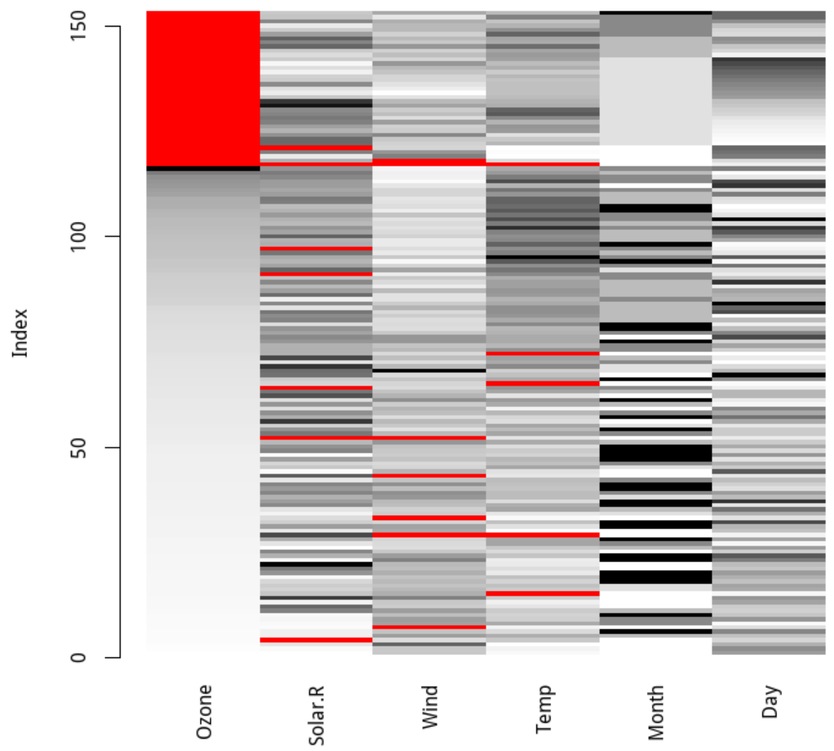
```
> library(VIM)
> mydata.aggrplot <- aggr(mydata,
col=c('lightblue','red'), numbers=TRUE,
prop = TRUE, sortVars=TRUE,
labels=names(mydata), cex.axis=.7, gap=3)

 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
```



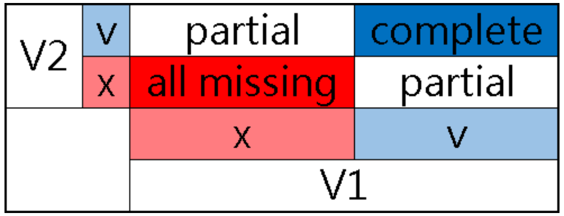
Matrix Plot

```
> 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



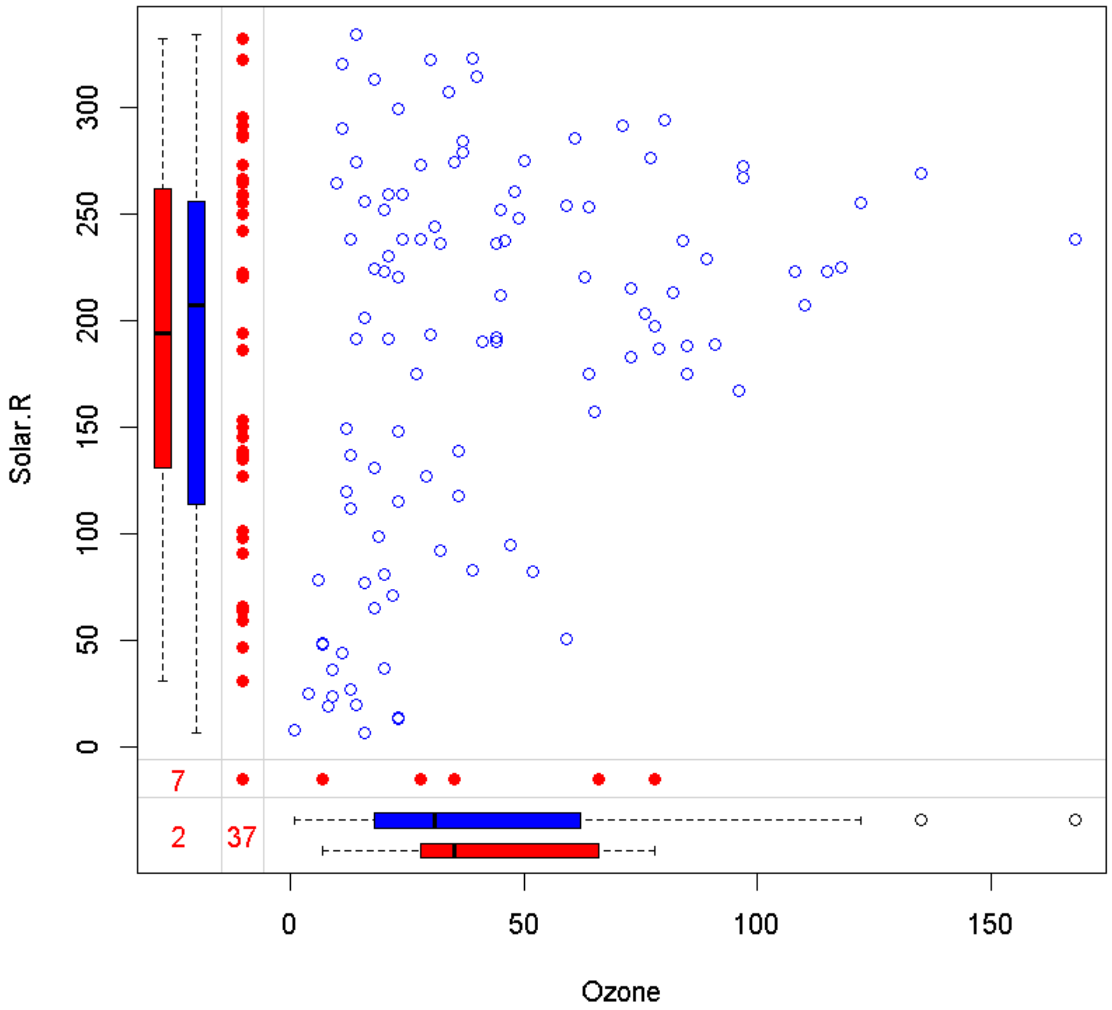
- `rr`: response-response, both variables are observed
- `rm`: response-missing, row observed, column missing
- `mr`: missing-response, row missing, column observed
- `mm`: missing-missing, both variables are missing

```
> 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.

列表刪除法 (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

成對刪除法 (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
```

以平均值補值 (Mean Substitution)

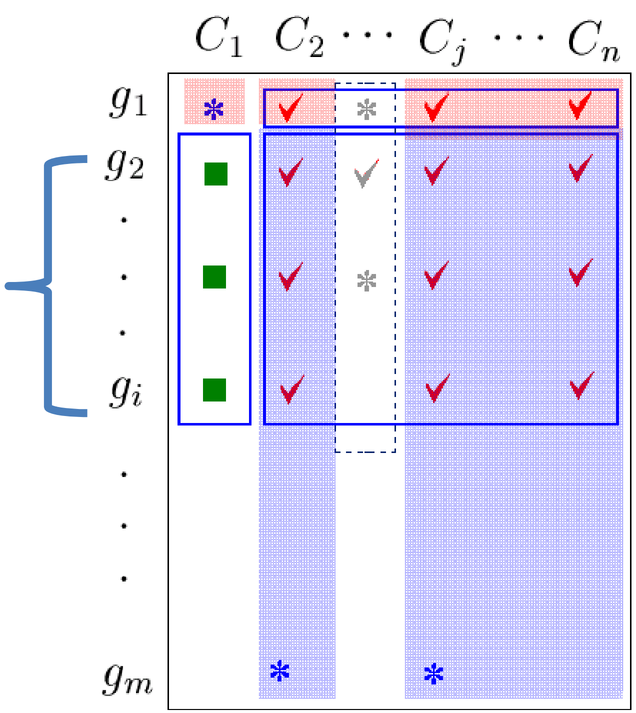
- A very simple but popular approach is to substitute means for the missing values.
- This method produces biased estimates and can severely distort the distribution of the variable in which missing values are substituted.
- Due to these **distributional problems**, it is often recommended to ignore 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
```

KNN 補值法 (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 $\quad \widehat{C_1(g_1)} = \frac{\sum_{k=1}^{K} w_k C_1(g_k)}{\sum_{k=1}^{K} w_k}$

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

kNN {VIM}: k-Nearest Neighbour Imputation

Description

k-Nearest Neighbour Imputation based on a variation of the Gower Distance for numerical, categorical, ordered and semi-continous variables.

Usage

```
kNN(data, variable = colnames(data), metric = NULL, k = 5,
  dist_var = colnames(data), weights = NULL, numFun = median,
  catFun = maxCat, makeNA = NULL, NAcond = NULL, impNA = TRUE,
  donorcond = NULL, mixed = vector(), mixed.constant = NULL,
  trace = FALSE, imp_var = TRUE, imp_suffix = "imp", addRandom = FALSE,
  useImputedDist = TRUE, weightDist = FALSE)
```

```
> names(airquality)
[1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"
> airquality.imp.median <- kNN(airquality[1:4], k=5)
> head(airquality.imp.median)
  Ozone Solar.R Wind Temp Ozone_imp Solar.R_imp Wind_imp Temp_imp
1    41     190  7.4   67     FALSE       FALSE    FALSE    FALSE
2    36     118  8.0   72     FALSE       FALSE    FALSE    FALSE
3    12     149 12.6   74     FALSE       FALSE    FALSE    FALSE
4    18     313 11.5   62     FALSE       FALSE    FALSE    FALSE
5    35      92 14.3   56      TRUE        TRUE    FALSE    FALSE
6    28     242 14.9   66     FALSE        TRUE    FALSE    FALSE
```

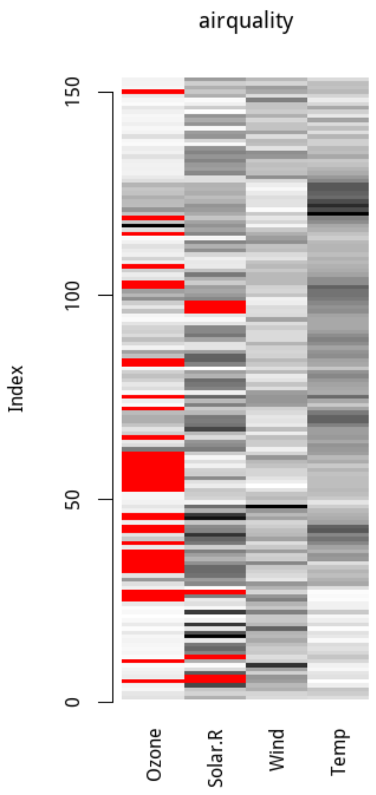

- Gower JC, 1971, A General Coefficient of Similarity and Some of Its Properties. Biometrics, 857–871.
- Alexander Kowarik and Matthias Templ, 2016, Imputation with the R Package VIM, Journal of Statistical Software, Volume 74, Issue 7.

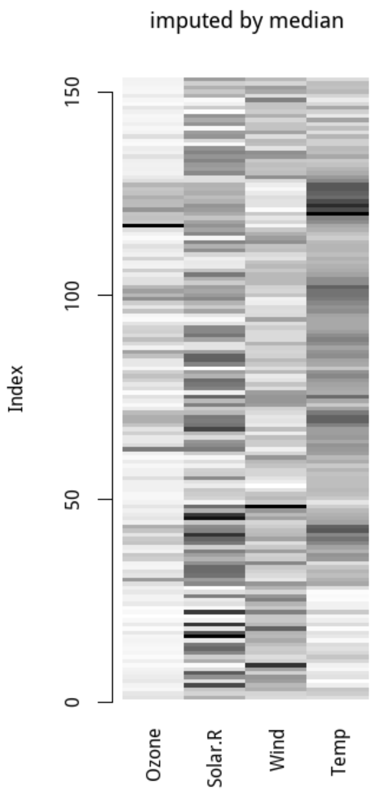
kNN {VIM}: k-Nearest Neighbour Imputation

```
> matrixplot(airquality[1:4], interactive = F, main="airquality")
> matrixplot(airquality.imp.median[1:4], interactive = F, main="imputed by median")
```

```
> airquality.imp.mean <- kNN(airquality[1:4], k = 5, metric = dist, numFun = mean)
> airquality.imp.tmean <- kNN(airquality[1:4], k = 5, numFun = trim_mean)
```

```
trim_mean <- function(x){
   mean(x, trim = 0.1)
}
```

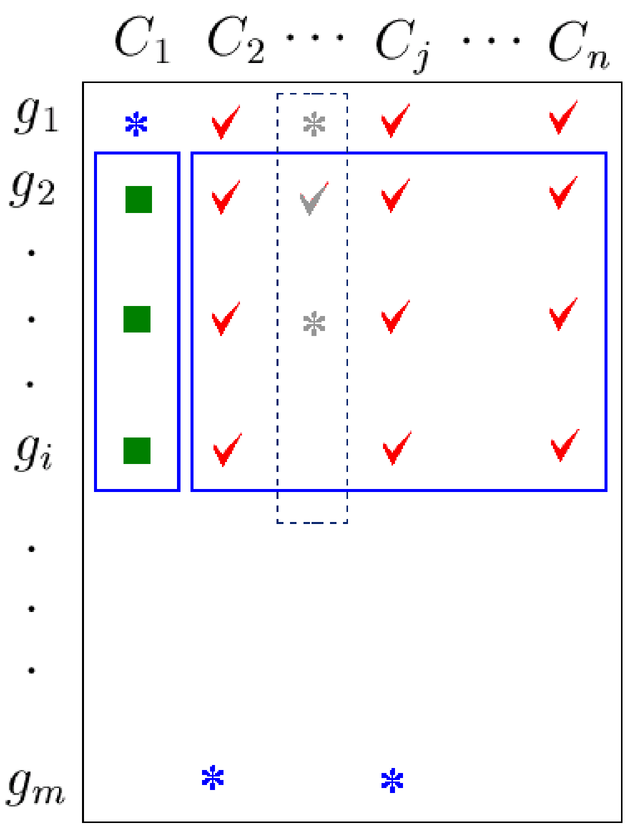





```
> airquality.imp.mean <- kNN(airquality[1:4], k=5, metric=dist, numFun=mean)
Warning messages:
1: In `[<-.data.table`(`*tmp*`, indexNA2s[, variable[j]], variable[j],  :
  Coerced 'double' RHS to 'integer' to match the column's type; may have trur
```

迴歸補值法 (Regression Methods)

- Using fitted regression values to replace missing values.
- The model must be chosen so that it does not yields invalid fitted values. e.g., negative values.
- This technique might be more accurate than simply substituting a measure of central tendency, since the imputed value is based on other input variables.



Regression

Model:

$$C_1 = \beta_0 + \sum_{j \in \mathrm{C}} \beta_j C_j$$

C: Observed C_i's without missing values

Imputation:

$$\widehat{C_1(g_1)} = \hat{\beta}_0 + \sum_{j \in \mathrm{C}} \hat{\beta}_j C_j(g_1)$$

regressionImp {VIM}: Regression Imputation

Description

Impute missing values based on a regression model.

Usage

```
regressionImp(formula, data, family = "AUTO", robust = FALSE,
  imp_var = TRUE, imp_suffix = "imp", mod_cat = FALSE)
```

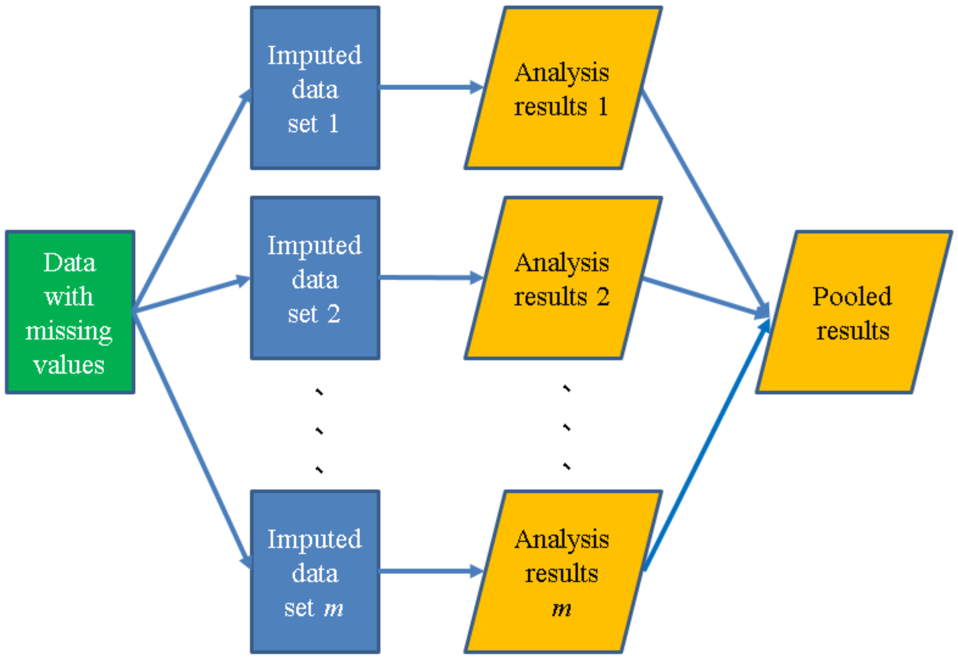
```
> airquality.imp.lm <- regressionImp(Ozone ~ Wind + Temp, data=airquality)
Error in regressionImp_work(formula = formula, data = data, family = family,  :
  找不到物件 'nLev'
>
> data(sleep)
> summary(sleep)
    BodyWgt            BrainWgt            NonD            Dream            Sleep
 Min.   :   0.005   Min.   :   0.14   Min.   : 2.100   Min.   :0.000   Min.   : 2.60
 1st Qu.:   0.600   1st Qu.:   4.25   1st Qu.: 6.250   1st Qu.:0.900   1st Qu.: 8.05
 Median :   3.342   Median :  17.25   Median : 8.350   Median :1.800   Median :10.45
 Mean   : 198.790   Mean   : 283.13   Mean   : 8.673   Mean   :1.972   Mean   :10.53
 3rd Qu.:  48.203   3rd Qu.: 166.00   3rd Qu.:11.000   3rd Qu.:2.550   3rd Qu.:13.20
 Max.   :6654.000   Max.   :5712.00   Max.   :17.900   Max.   :6.600   Max.   :19.90
                                      NA's   :14       NA's   :12      NA's   :4
      Span              Gest             Pred            Exp            Danger
 Min.   :  2.000   Min.   : 12.00   Min.   :1.000   Min.   :1.000   Min.   :1.000
 1st Qu.:  6.625   1st Qu.: 35.75   1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000
 Median : 15.100   Median : 79.00   Median :3.000   Median :2.000   Median :2.000
 Mean   : 19.878   Mean   :142.35   Mean   :2.871   Mean   :2.419   Mean   :2.613
 3rd Qu.: 27.750   3rd Qu.:207.50   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
 Max.   :100.000   Max.   :645.00   Max.   :5.000   Max.   :5.000   Max.   :5.000
 NA's   :4         NA's   :4
```

regressionImp {VIM}: Regression Imputation

```
> sleep.imp.lm <- regressionImp(Dream + NonD ~ BodyWgt + BrainWgt, data=sleep)
> summary(sleep.imp.lm)
    BodyWgt            BrainWgt             NonD              Dream              Sleep
 Min.   :   0.005   Min.   :   0.14   Min.   :-11.733   Min.   :-0.6897   Min.   : 2.60
 1st Qu.:   0.600   1st Qu.:   4.25   1st Qu.:  6.525   1st Qu.: 1.0000   1st Qu.: 8.05
 Median :   3.342   Median :  17.25   Median :  8.500   Median : 1.9312   Median :10.45
 Mean   : 198.790   Mean   : 283.13   Mean   :  8.335   Mean   : 1.9326   Mean   :10.53
 3rd Qu.:  48.203   3rd Qu.: 166.00   3rd Qu.: 10.550   3rd Qu.: 2.2750   3rd Qu.:13.20
 Max.   :6654.000   Max.   :5712.00   Max.   : 17.900   Max.   : 6.6000   Max.   :19.90
                                                                          NA's   :4
      Span              Gest             Pred            Exp             Danger
 Min.   :  2.000   Min.   : 12.00   Min.   :1.000   Min.   :1.000   Min.   :1.000
 1st Qu.:  6.625   1st Qu.: 35.75   1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000
 Median : 15.100   Median : 79.00   Median :3.000   Median :2.000   Median :2.000
 Mean   : 19.878   Mean   :142.35   Mean   :2.871   Mean   :2.419   Mean   :2.613
 3rd Qu.: 27.750   3rd Qu.:207.50   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:4.000
 Max.   :100.000   Max.   :645.00   Max.   :5.000   Max.   :5.000   Max.   :5.000
 NA's   :4         NA's   :4
 Dream_imp        NonD_imp
 Mode :logical   Mode :logical
 FALSE:50        FALSE:48
 TRUE :12        TRUE :14
 NA's :0         NA's :0
```

多重補值法、多重插補法 (Multiple Imputation)

- 多重插補法是迴歸插補法的一種，也是模型基礎法的延伸，它是目前插補法中最受推崇的主流方法。
- Multiple imputation requires three steps
  - Imputation: impute the missing entries of the incomplete data sets *m* times. Imputed values are drawn for a distribution (that can be different for each missing entry). This step results is *m* complete data sets.
  - Analysis: Analyze each of the *m* completed data sets. This step results in *m* analyses.
  - Pooling: Integrate the *m* analysis results into a final result.



- Rubin (1987) has shown that if the method to create imputations is 'proper', then the resulting inferences will be statistically valid.

Multiple Imputation Online: www.multiple-imputation.com

Rubin, D.B. (1987), Multiple Imputation for Nonresponse in Surveys, New York: John Wiley & Sons, Inc.
Little, R.J.A. and Rubin, D.B. (1987), Statistical Analysis with Missing Data, New York: John Wiley & Sons, Inc.

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.logreg.boot  mice.impute.mean         mice.impute.norm
[13] mice.impute.norm.boot    mice.impute.norm.nob     mice.impute.norm.predict
[16] mice.impute.passive      mice.impute.pmm          mice.impute.polr
[19] mice.impute.polyreg      mice.impute.quadratic    mice.impute.rf
[22] mice.impute.ri           mice.i
[25] mice.theme
see '?methods' for accessing help an
> ? 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	

van Buuren, S., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R. Journal of Statistical Software, 45(3), 1–67. https://doi.org/10.18637/jss.v045.i03

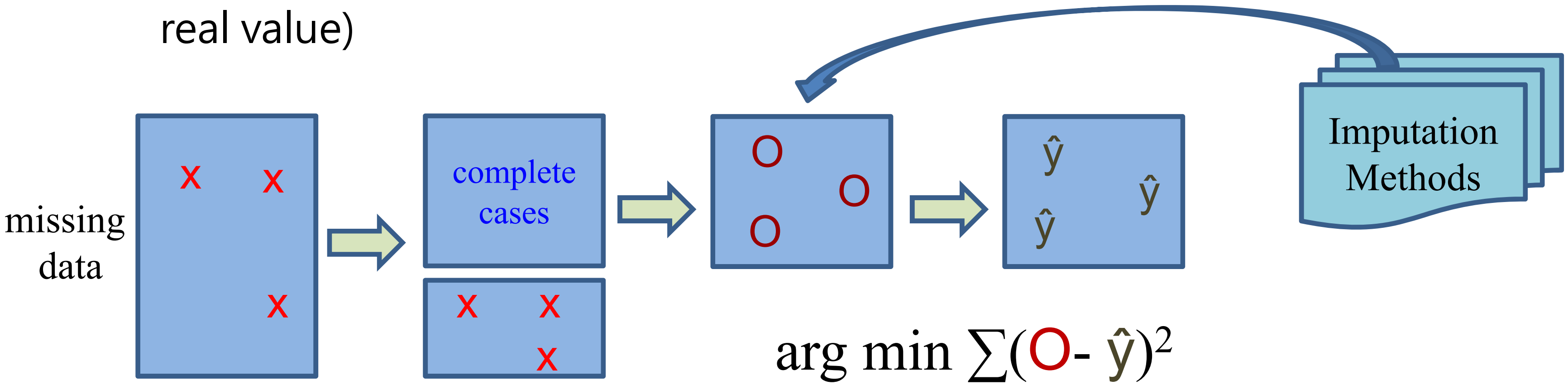
Quick Tutorial on MICE Package

```
> # Generate 10% missing values at Random
> iris.mis <- prodNA(iris, noNA = 0.1)  # library(missForest)
> # Check missing values introduced in the data
> summary(iris.mis)
> iris.mis <- subset(iris.mis, select = -c(Species))
> summary(iris.mis)
>
> #  A tabular form of missing value present in each variable
> library(mice)
> md.pattern(iris.mis)
> # Visualization
> library(VIM)
> mice_plot <- aggr(iris.mis, col=c('navyblue','yellow'), numbers=TRUE, sortVars=TRUE,
                  labels=names(iris.mis), cex.axis=.7,
                  gap=3, ylab=c("Missing data","Pattern"))

> #  Imputation
> imputed.Data <- mice(iris.mis, m = 5, maxit = 50, method = 'pmm', seed = 500)
> summary(imputed.Data)
> # Check imputed values
> imputed.Data$imp$Sepal.Width
> # Get complete data (2nd out of 5)
> completeData <- complete(imputed.Data, 2)
> # Build predictive model
> fit <- with(data = imputed.Data, exp = lm(Sepal.Width ~ Sepal.Length + Petal.Width))
> # Combine results of all 5 models
> combine <- pool(fit)
> summary(combine)
```

Source: http://www.analyticsvidhya.com/blog/2016/03/tutorial-powerful-packages-imputing-missing-values/

哪一種補值方法較好?

- 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.
- 建議:
  - 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)

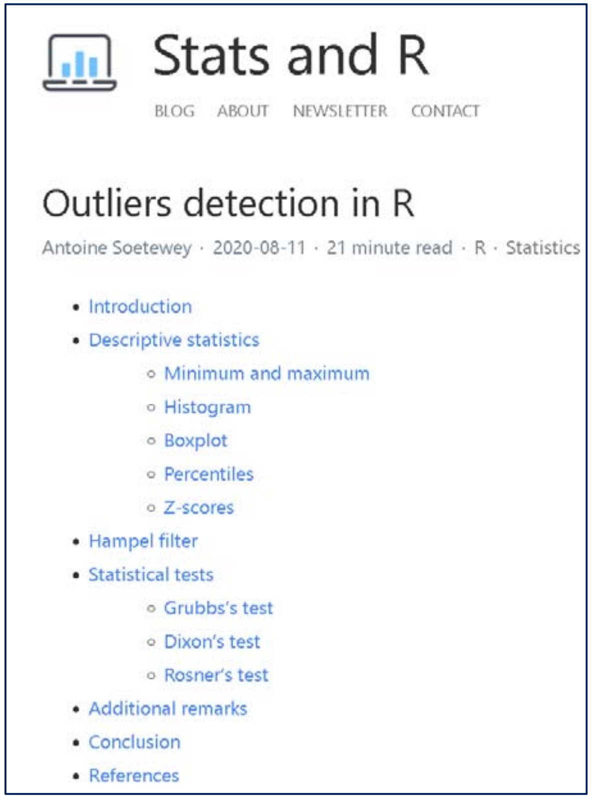


異常檢測的統計方法 (Outliers Detection in R)

- **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.
  - **`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: https://statsandr.com/blog/outliers-detection-in-r/


Stats and R

BLOG ABOUT NEWSLETTER CONTACT

Outliers detection in R

Antoine Soetewey · 2020-08-11 · 21 minute read · R · Statistics

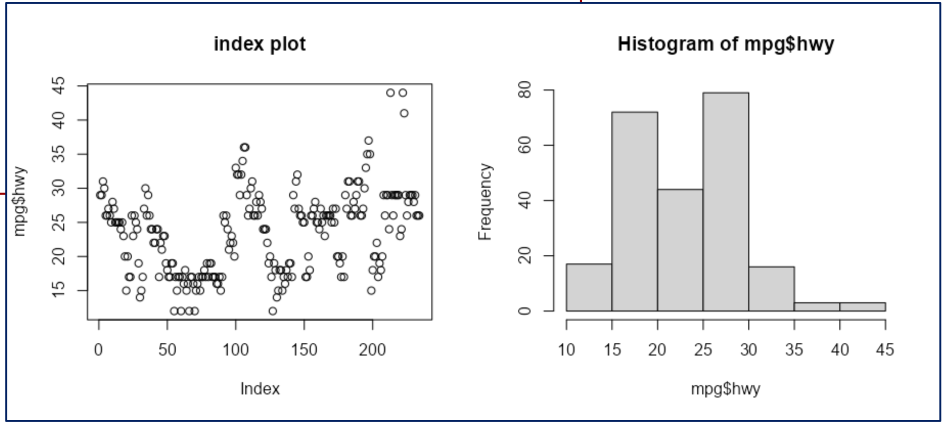
- Introduction
- Descriptive statistics
  - Minimum and maximum
  - Histogram
  - Boxplot
  - Percentiles
  - Z-scores
- Hampel filter
- Statistical tests
  - Grubbs's test
  - Dixon's test
  - Rosner's test
- Additional remarks
- Conclusion
- References

R package: ou1tliers
Statistlcal Tests

- `chisq.out.test`: Chi-squared test for outlier
- `cochran.test`: Test for outlying or inlying variance
- `dixon.test`: Dixon tests for outlier
- `grubbs.test`: Grubbs tests for one or two outliers in data sample.

- Dixon, W.J. (1950). Analysis of extreme values. Ann. Math. Stat. 21, 4, 488-506.
- Dixon, W.J. (1951). Ratios involving extreme values. Ann. Math. Stat. 22, 1, 68-78.
- Snedecor, G.W., Cochran, W.G. (1980). Statistical Methods (seventh edition). Iowa State University Press, Ames, Iowa.
- Grubbs, F.E. (1950). Sample Criteria for testing outlying observations. Ann. Math. Stat. 21, 1, 27-58.


```
> library(outliers)
> dim(mpg)
[1] 234  11
> head(mpg, 3)
# A tibble: 3 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    2008     4 manual(m6) f        20    31 p     compact
>
> summary(mpg$hwy)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  12.00   18.00   24.00   23.44   27.00   44.00
> par(mfrow = c(1, 2))
> plot(mpg$hwy, main = "index plot")
> hist(mpg$hwy)
```



mpg {ggplot2}

Fuel economy data from 1999 to 2008 for 38 popular models of cars

hwy: highway miles per gallon

Grubbs' Test for a Single Outlier

- **Assumption**: the data (without any outliers) are approximately normally distributed.
- **Hypothesis**:
  - H_0: There are no outliers in the data set
  - H_1: 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

The **Grubbs test** detects one outlier at a time (highest or lowest value), so the null and alternative hypotheses are as follows:

if we want to test the highest value

- H_0: The highest value is not an outlier
- H_1: The highest value is an outlier

if we want to test the lowest value.

- H_0: The lowest value is not an outlier
- H_1: The lowest value is an outlier

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

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.

R package: outliers
StatistIcal Tests

```
> test <- grubbs.test(mpg$hwy)
> test

         Grubbs test for one outlier

data:  mpg$hwy
G = 3.45274, U = 0.94862, p-value = 0.05555
alternative hypothesis: highest value 44 is an outlier
>
>
> test <- grubbs.test(mpg$hwy, opposite = TRUE)
> test

         Grubbs test for one outlier

data:  mpg$hwy
G = 1.92122, U = 0.98409, p-value = 1
alternative hypothesis: lowest value 12 is an outlier

>
>
> dixon.test(mpg$hwy)
Error in dixon.test(mpg$hwy) : Sample size must be in range 3-30
>
```

> # The p-value is 0.056. At the 5% significance level, we do not reject the hypothesis that the highest value 44 is not an outlier.

> # At the 5% significance level, we do not reject the hypothesis that the lowest value 12 is not an outlier.

穩健統計方法 (Robust Statistical Methods)

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

CRAN Task View: Robust Statistical Methods

Maintainer:	Martin Maechler
Contact:	Martin.Maechler at R-project.org
Version:	2023-04-05
URL:	https://CRAN.R-project.org/view=Robust
Source:	https://github.com/cran-task-views/Robust/
Contributions:	Suggestions and improvements for this task view are very welcome and can be made through issues or pull requests on GitHub or via e-mail to the maintainer address. For further details see the Contributing guide.
Citation:	Martin Maechler (2023). CRAN Task View: Robust Statistical Methods. Version 2023-04-05. URL https://CRAN.R-project.org/view=Robust.
Installation:	The packages from this task view can be installed automatically using the ctv package. For example, `ctv::install.views("Robust", coreOnly = TRUE)` installs all the core packages or `ctv::update.views("Robust")` installs all packages that are not yet installed and up-to-date. See the CRAN Task View Initiative for more details.

Robust (or "resistant") methods for statistics modelling have been available in S from the very beginning in the 1980s; and then in R in package `stats`. Examples are `median()`, `mean(*, trim =. )`, `mad()`, `IQR()`, or also `fivenum()`, the statistic behind `boxplot()` in package `graphics`) or `lowess()` (and `loess()`) for robust nonparametric regression, which had been complemented by `runmed()` in 2003. Much further important functionality has been made available in recommended (and hence present in all R versions) package MASS (by Bill Venables and Brian Ripley, see *the* book Modern Applied Statistics with S). Most importantly, they provide `rlm()` for robust regression and `cov.rob()` for robust multivariate scatter and covariance.

Robust Location and Scatter Estimators

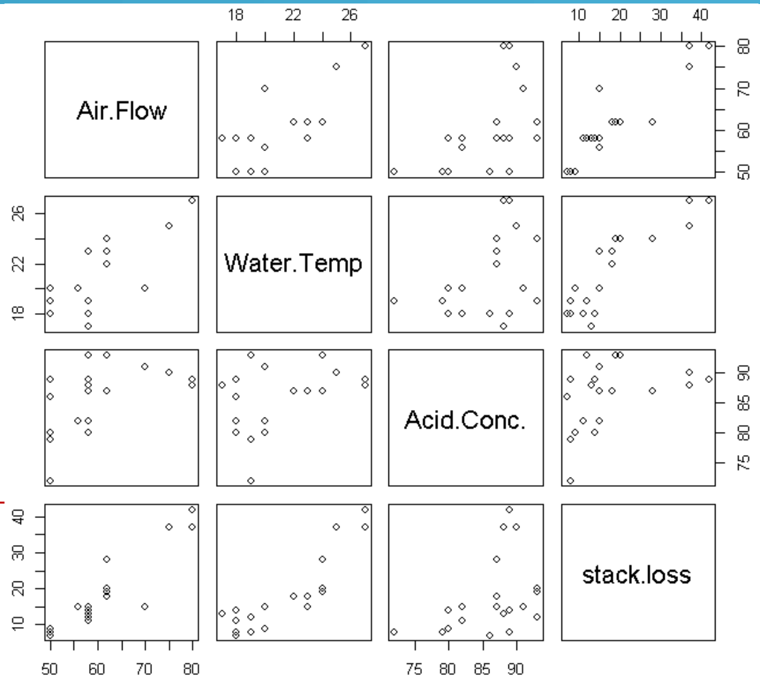
- Median, MAD (median of the absolute deviations from the median)
- M-estimator (Huber, 1964; Maronna, 1976)
- Stahel-Donoho estimator (Stahel, 1981; Donoho, 1982)
- MVE (minimum volume ellipsoid), MCD (minimum covariance determinant) (Rousseeuw, 1983, 1984, 1985)
- S-estimator (Davis, 1987)
- Depth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

MVE (minimum volume ellipsoid)

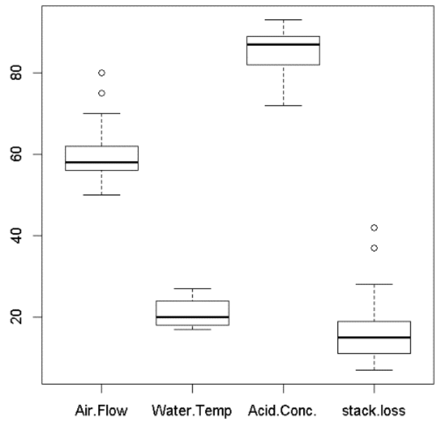
- Affine equivariant with high breakdown points.
- The existing efficient algorithm for computation.
- Readily available implementations.
- Ability to Identify extreme values.

Brownlee's Stack Loss Plant Data

- `stackloss {datasets}`, Operational data of a plant for the oxidation of ammonia to nitric acid.
  - Air.Flow: Flow of cooling air
  - Water.Temp: Cooling Water Inlet Temperature
  - Acid.Conc.: Concentration of acid [per 1000, minus 500]
  - stack.loss: Stack loss

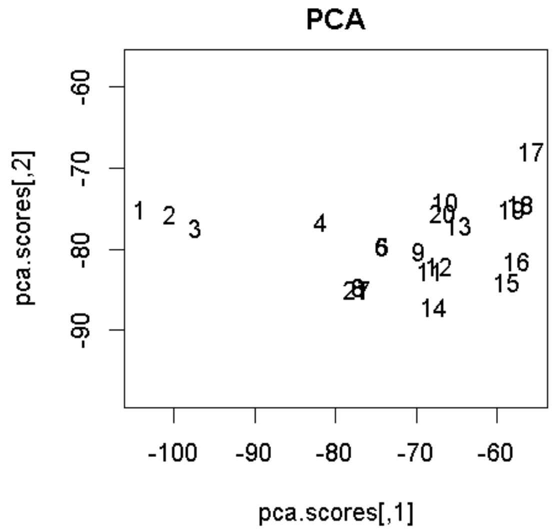



```
> data(stackloss)
> dim(stackloss)
[1] 21  4
> head(stackloss, 4)
  Air.Flow Water.Temp Acid.Conc. stack.loss
1       80         27         89         42
2       80         27         88         37
3       75         25         90         37
4       62         24         87         28
> summary(stackloss)
    Air.Flow       Water.Temp      Acid.Conc.       stack.loss
 Min.   :50.00   Min.   :17.0   Min.   :72.00   Min.   : 7.00
 1st Qu.:56.00   1st Qu.:18.0   1st Qu.:82.00   1st Qu.:11.00
 Median :58.00   Median :20.0   Median :87.00   Median :15.00
 Mean   :60.43   Mean   :21.1   Mean   :86.29   Mean   :17.52
 3rd Qu.:62.00   3rd Qu.:24.0   3rd Qu.:89.00   3rd Qu.:19.00
 Max.   :80.00   Max.   :27.0   Max.   :93.00   Max.   :42.00
```

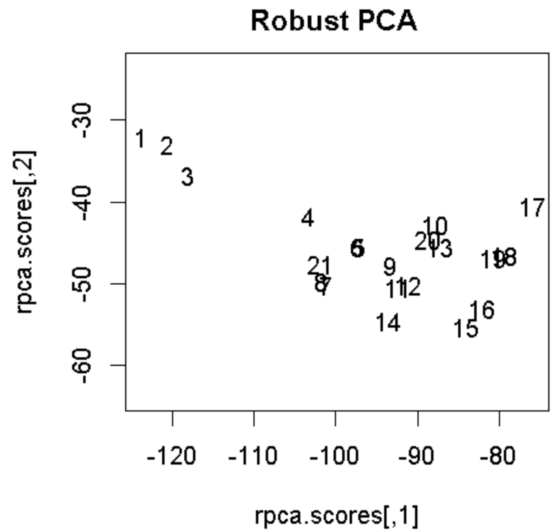


Robust PCA

```
> library(MASS)
> cov(stackloss)
            Air.Flow Water.Temp Acid.Conc. stack.loss
Air.Flow    84.05714  22.657143  24.571429   85.76429
Water.Temp  22.65714   9.990476   6.621429   28.14762
Acid.Conc.  24.57143   6.621429  28.714286   21.79286
stack.loss  85.76429  28.147619  21.792857  103.46190
> cov.mve(stackloss)$cov
             Air.Flow Water.Temp Acid.Conc. stack.loss
Air.Flow    21.600000   6.657143  11.285714  18.228571
Water.Temp   6.657143   6.066667   4.690476   7.900000
Acid.Conc.  11.285714   4.690476  23.095238   9.642857
stack.loss  18.228571   7.900000   9.642857  17.828571
```




```
> par(mfrow=c(2,1))
> library(MASS)
> ccov <- cov(stackloss)
> pca.scores <- as.matrix(stackloss) %*%
eigen(ccov)$vectors[,1:2]
> plot(pca.scores, main="PCA", asp=1, type="n")
> text(pca.scores, label=1:nrow(stackloss))
> rcov <- cov.mve(stackloss)$cov
> rpca.scores <- as.matrix(stackloss) %*%
                 eigen(rcov)$vectors[,1:2]
> plot(rpca.scores, main="Robust PCA", asp=1, type="n")
> text(rpca.scores, label=1:nrow(stackloss))
```



資料轉換
Classical (Numerical) Data Table

*j*th variable

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

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

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.

在統計學和機器學習的資料分析中，進行資料轉換的主要原因有以下幾點：

1. **正規化和標準化**：許多機器學習算法在處理數據時，對數據的尺度和分佈有一定的假設。例如，許多算法假設數據遵循正態分佈，或者所有特徵都在同一尺度上。透過資料轉換，我們可以將數據轉換為符合這些假設的形式，從而提高模型的性能。
2. **處理偏態數據**：在實際的數據集中，我們經常會遇到偏態（skewed）數據。這種數據的特點是，其分佈不均，有一邊的尾部特別長。這種情況下，一些統計測量（如均值和方差）可能會被拉向長尾的方向，導致對數據的理解偏差。透過資料轉換，我們可以將偏態數據轉換為更接近正態分佈的形式，從而使得統計測量更為準確。
3. **線性化關係**：許多統計和機器學習模型都假設數據中的變數之間存在線性關係。但在實際數據中，這種線性關係可能並不存在。透過資料轉換，我們可以將非線性關係轉換為線性關係，從而使得這些模型可以更好地擬合數據。
4. **處理異常值和離群值**：在實際數據中，我們經常會遇到異常值和離群值。這些值可能會對模型的訓練產生不良影響。透過資料轉換，我們可以將這些異常值和離群值轉換為更為合理的值，從而提高模型的穩定性和性能。

資料離散化 (Data Discretization)

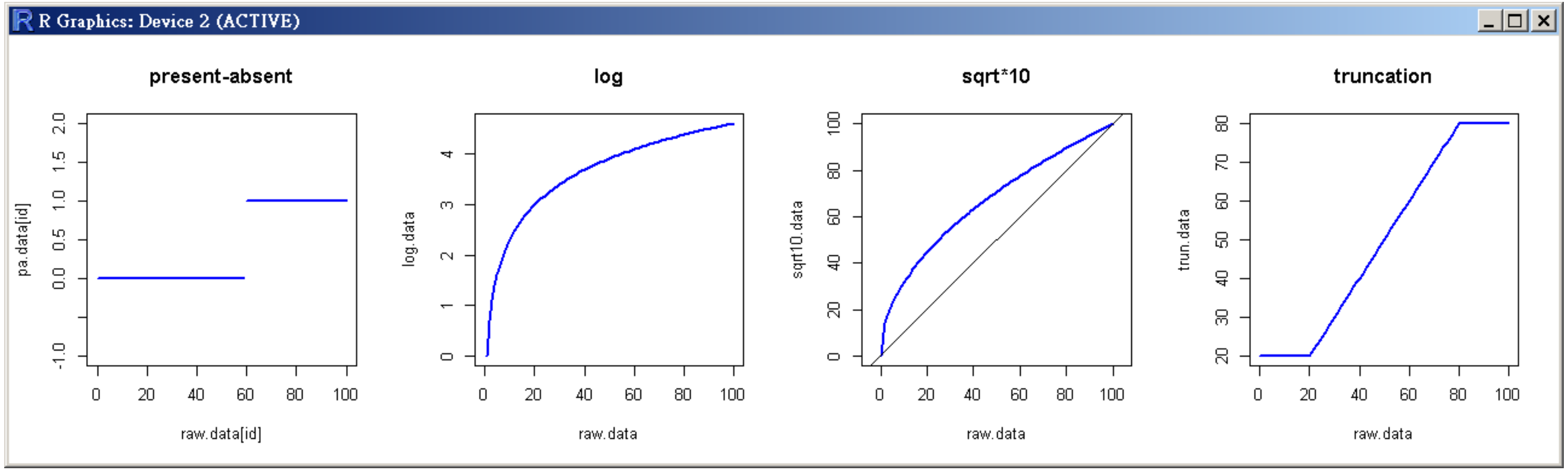
- Data discretization transforms numeric data by mapping values to interval or concept labels.
- **by binning**: This is a top-down unsupervised splitting technique based on a specified number of bins.
- **by histogram analysis**: In this technique, a histogram partitions the values of an attribute into disjoint ranges called buckets or bins. It is also an unsupervised method.
- **by cluster analysis**: In this technique, a clustering algorithm can be applied to discretize a numerical attribute by partitioning the values of that attribute into clusters or groups.
- **by decision tree analysis**: Here, a decision tree employs a top-down splitting approach; it is a supervised method. To discretize a numeric attribute, the method selects the value of the attribute that has minimum entropy as a split-point, and recursively partitions the resulting intervals to arrive at a hierarchical discretization.
- **by correlation analysis**: This employs a bottom-up approach by finding the best neighboring intervals and then merging them to form larger intervals, recursively. It is supervised method.

常見資料轉換方法 (1)

```
> 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")
```

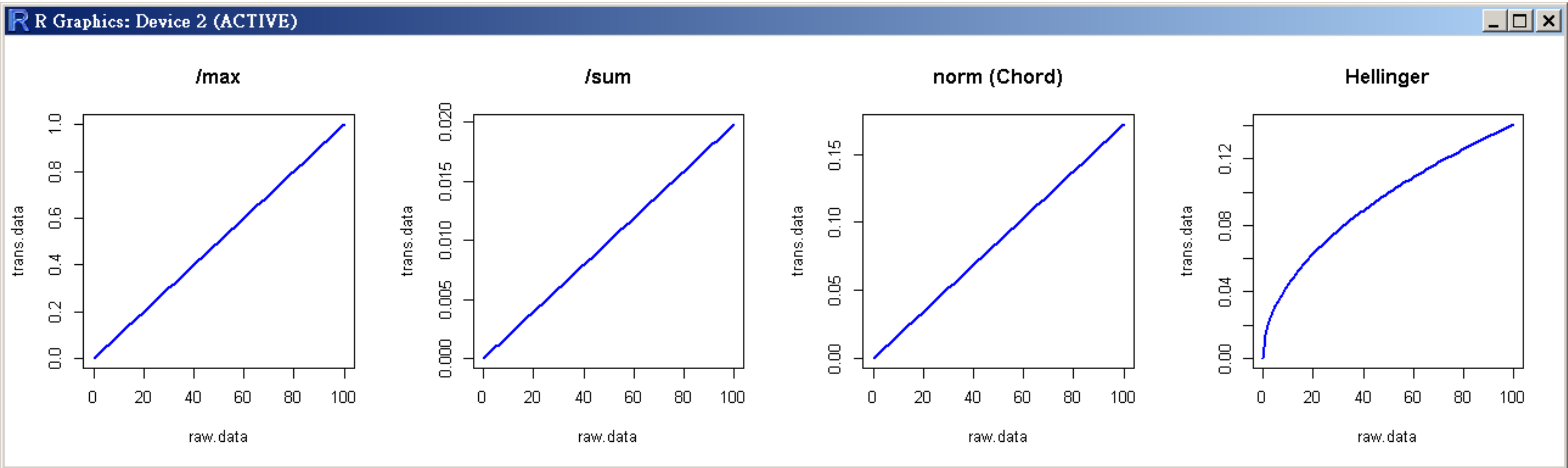
NOTE: `apply(raw.data.matrix, 2, log)`

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



常見資料轉換方法 (2)

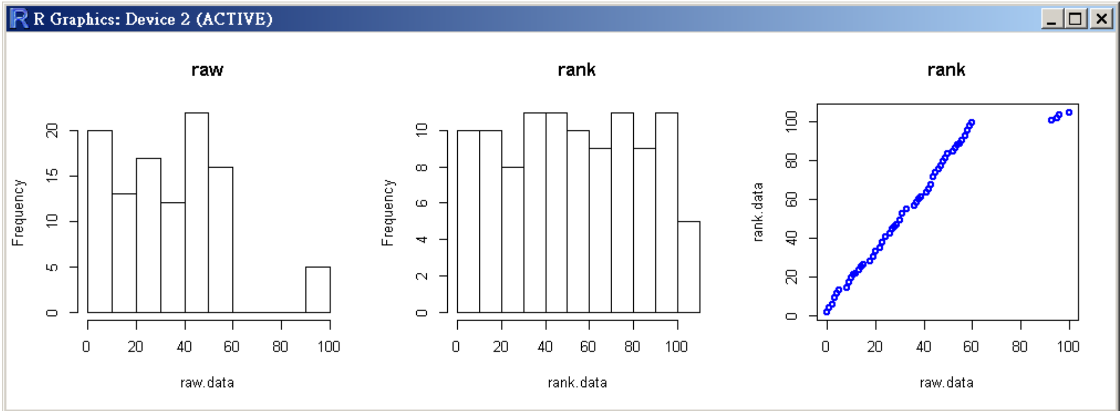
```
> par(mfrow = c(1,4))
> raw.data <- 0:100
> trans.data <- raw.data/max(raw.data)
> plot(raw.data, trans.data, main="/max", type="l", lwd=2, col="blue")
>
> trans.data <- raw.data/sum(raw.data)  #Species profile transformation
> plot(raw.data, trans.data, main="/sum", type="l", lwd=2, col="blue")
>
> trans.data <- raw.data/sqrt(sum(raw.data^2)) #length of 1, Chord transformation
> plot(raw.data, trans.data, main="norm (Chord)", type="l", lwd=2, col="blue")
>
> trans.data <- sqrt(raw.data/sum(raw.data)) #Hellinger transformation
> plot(raw.data, trans.data, main="Hellinger", type="l", lwd=2, col="blue")
```

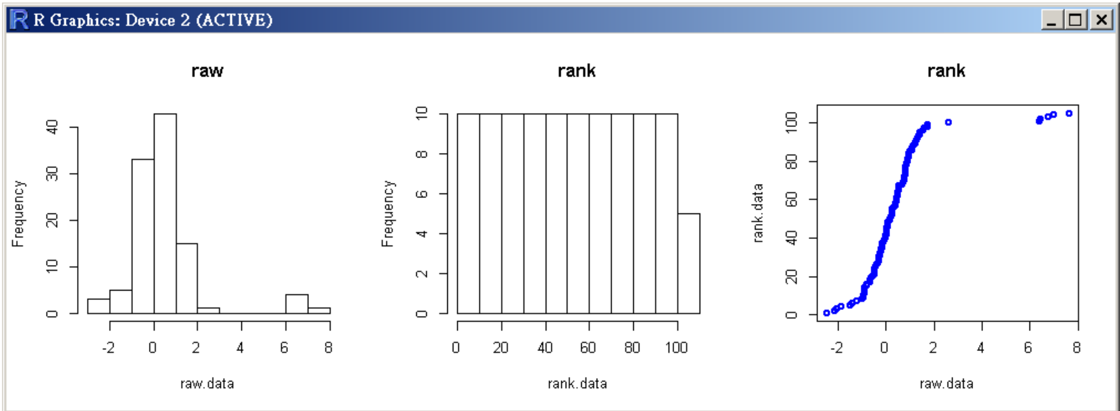


Other Transformations for community composition data: Chi-square distance transformation, Chi-square metric transformation

常見資料轉換方法 (3)

```
> par(mfrow=c(1,3)); set.seed(12345)
> raw.data <- c(sample(0:60, 100, replace=T), sample(90:100, 5, replace=T))
> rank.data <- rank(raw.data)
> hist(raw.data, main="raw")
> hist(rank.data, main="rank")
> plot(raw.data, rank.data, main="rank", lwd=2, col="blue")
```





常見資料轉換方法 (4)

- 倒數轉換(Reciprocal Transformation)
- The Square Root Transformation
- 指數函數: $f(x) = a^x$
- 對數函數$f(x) = \log_b(x)$
- Sigmoid函數 $\boldsymbol{f(x)} = \frac{\mathbf{1}}{\mathbf{1+e^{-x}}}$
- tanh函數 $\boldsymbol{f(x)} = \frac{e^{ax} - e^{-ax}}{e^{ax} + e^{-ax}}$

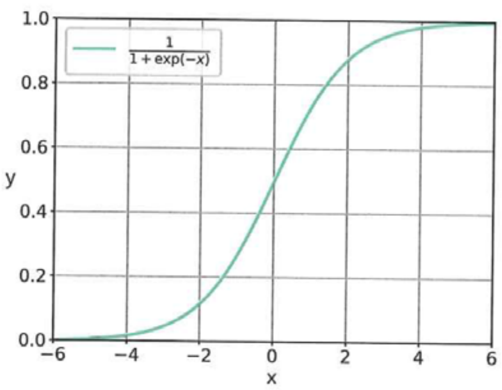


圖 5-9 *Sigmoid* 函數的圖形

https://en.wikipedia.org/wiki/Activation_function

WIKIPEDIA
The Free Encyclopedia

Activation function

Article Talk

From Wikipedia, the free encyclopedia

For the formalism used to approximate the influence of an extracellular elec
function. For a linear system's transfer function, see transfer function.

In artificial neural networks, the **activation function** of a node defines the output

Table of activation functions [edit]

The following table compares the properties of several activation functions that are functions of one fold x from the previous layer

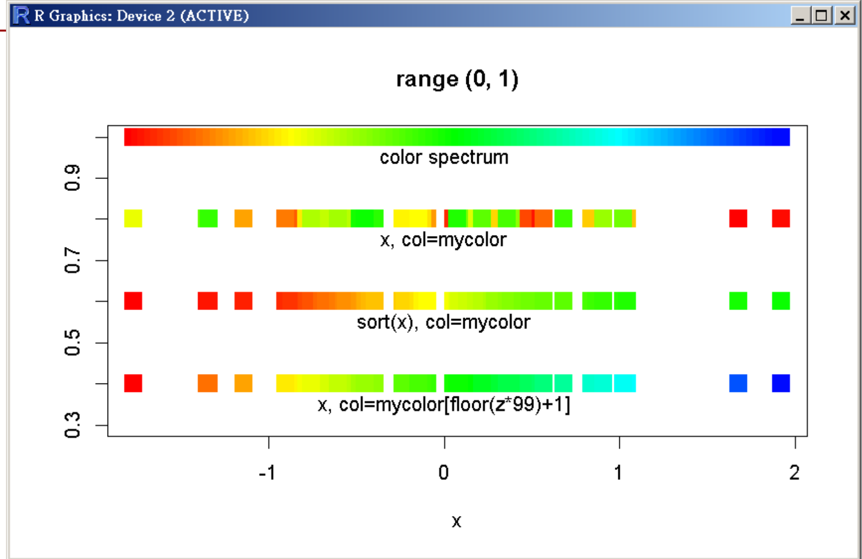
Name	Plot	Function, $g(x)$	Derivative of g, $g'(x)$	Range
Identity		x	1	$(-\infty, \infty)$
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$	0	$\{0, 1\}$
Logistic, sigmoid, or soft step		$\sigma(x) \doteq \frac{1}{1+e^{-x}}$	$g(x)(1-g(x))$	$(0, 1)$
Hyperbolic tangent (tanh)		$\tanh(x) \doteq \frac{e^x - e^{-x}}{e^x + e^{-x}}$	$1 - g(x)^2$	$(-1, 1)$
Rectified linear unit (ReLU)[8]		$(x)^+ \doteq \begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases} = \max(0, x) = x\mathbf{1}_{x>0}$	$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x > 0 \\ \text{undefined} & \text{if } x = 0 \end{cases}$	$[0, \infty)$

正規化、常規化 (Normalization): Transformation Using the Range，[0, 1]

- use the range of the variable as the divisor:
  - $z=(x-\min(x))/(\max(x)-\min(x))$, is **bounded by zero and one**, with at least one observed value at each of the end points.

```
x <- rnorm(50)
mycolor <- rainbow(150)[1:100]
z <- (x-min(x))/(max(x)-min(x))
plot(x, rep(1, length(x)), main="range (0, 1)", type="n", ylab="", ylim=c(0.3,1))
points(c(seq(min(x), max(x), length.out=100)), rep(1, 100), col=mycolor, cex=2, pch=15)
text(0, 0.95, "color spectrum")
points(x, rep(0.8, length(x)), col=mycolor, cex=2, pch=15)
text(0, 0.75, "x, col=mycolor")
points(sort(x), rep(0.6, length(x)), col=mycolor, cex=2, pch=15)
text(0, 0.55, "sort(x), col=mycolor")
points(x, rep(0.4, length(x)), col=mycolor[floor(z*99)+1], cex=2, pch=15)
text(0, 0.35, "x, col=mycolor[floor(z*99)+1]")
```

- The transformed variate is a linear function of the other one, so data standardized using these transformations will result in identical Euclidean distances.



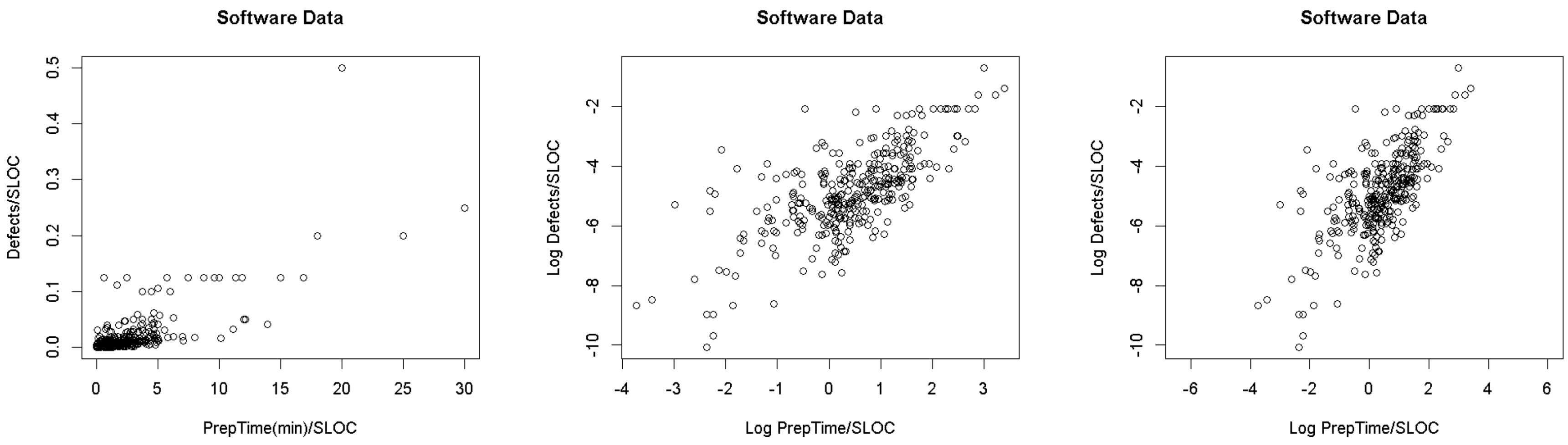
範例: Software Inspection Data

- 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)



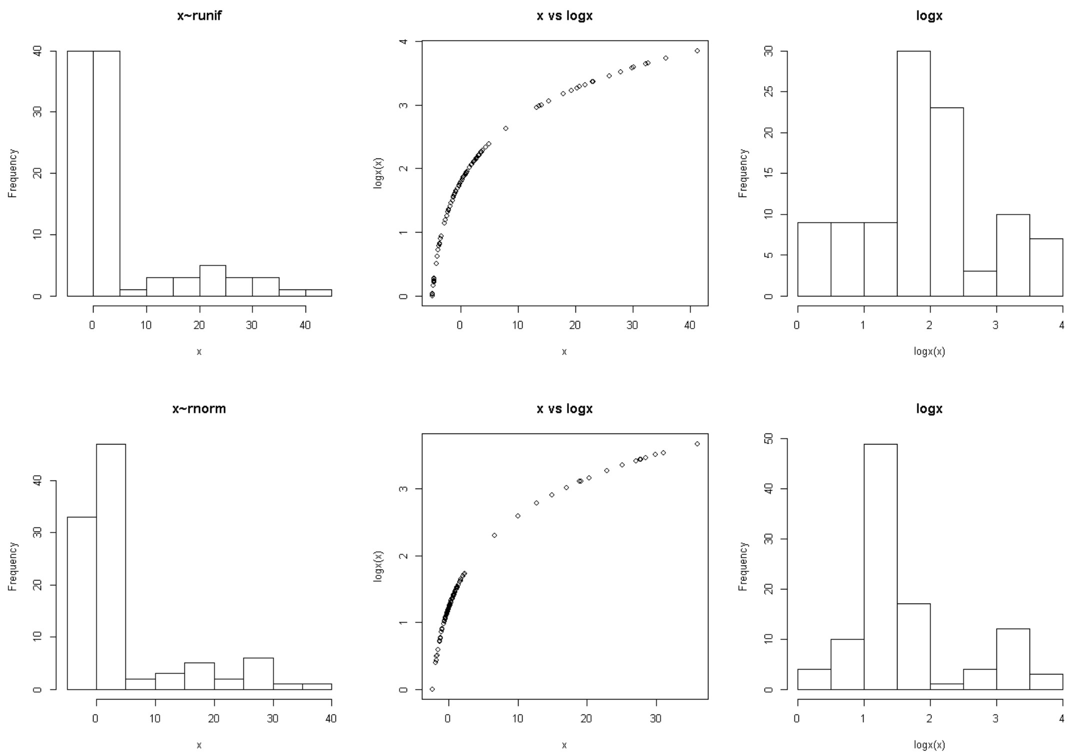

```
plot(data$prepsloc, data$defsloc, xlab="PrepTime(min)/SLOC", ylab="Defects/SLOC",
main="Software Data")

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

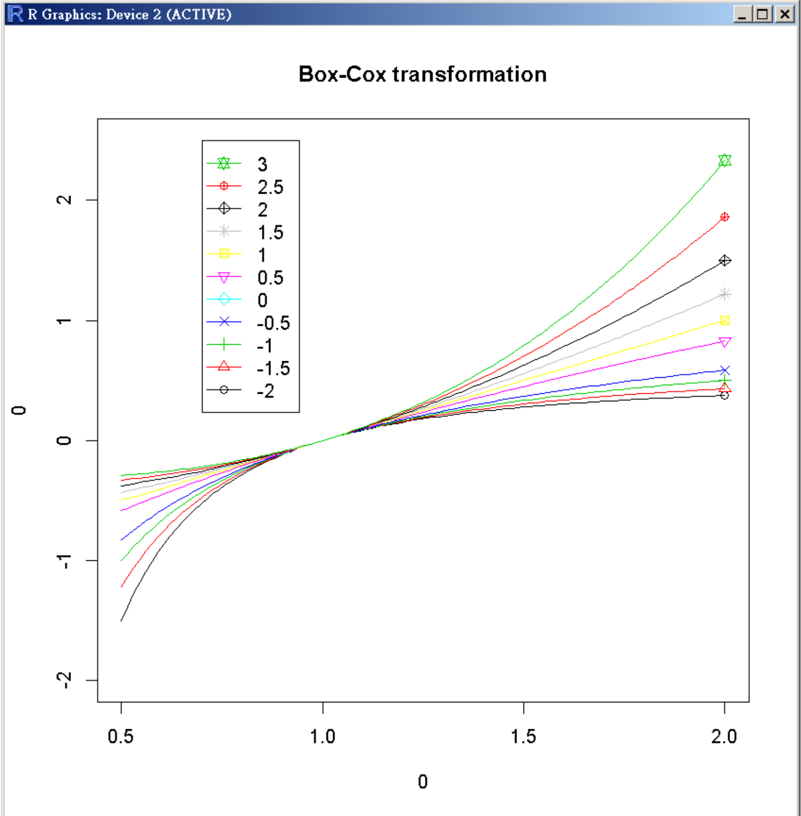
$$y(\lambda) = \begin{cases} \frac{y^\lambda - 1}{\lambda}, & \text{if } \lambda \neq 0; \\ \log y, & \text{if } \lambda = 0. \end{cases}$$

Box and Cox(1964)

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

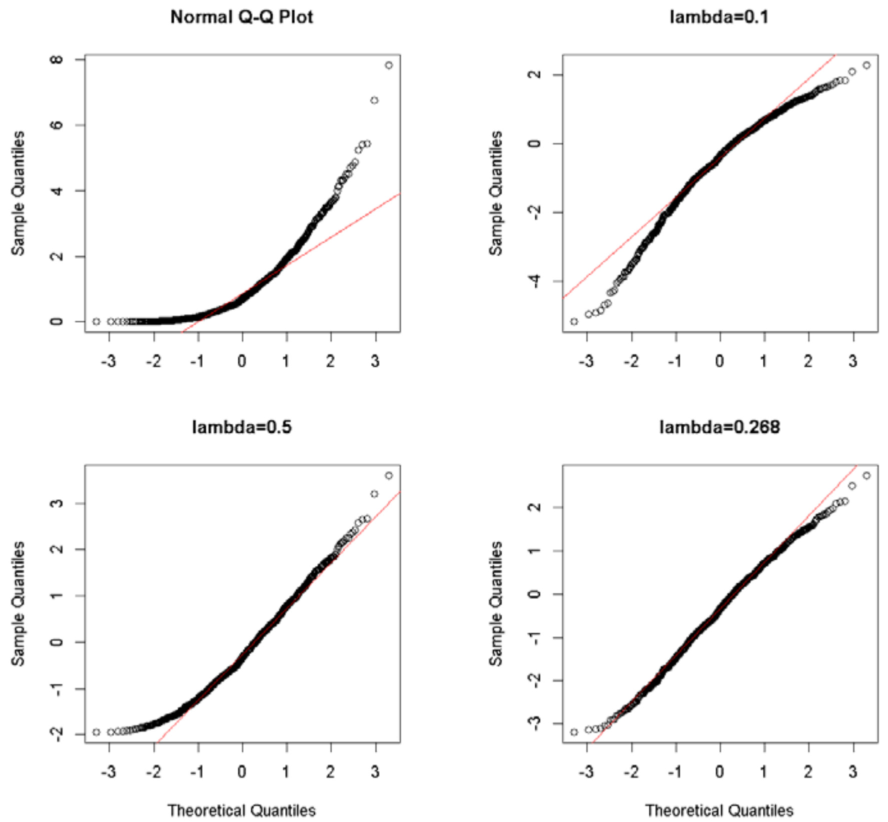
$$\boldsymbol{y} \sim \mathrm{N}(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n)$$

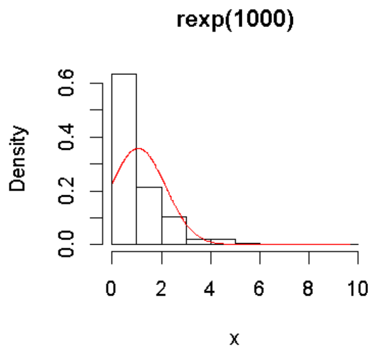
```
x <- seq(0.5, 2, length.out=100)
bc <- function(y, lambda){
    (y^lambda -1)/lambda
}
lambda <- seq(-2, 3, 0.5)
plot(0, 0, type="n", xlim=c(0.5, 2),
     ylim=c(-2, 2.5), main="Box-Cox transformation")
for(i in 1:length(lambda)){
   points(x, bc(x, lambda[i]), type="l", col=i)
   points(2, bc(2,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

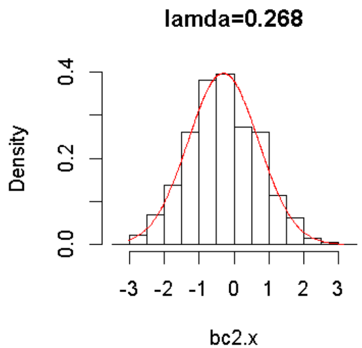


Box-Cox Transformations

```
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")
```







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

可估計Box-Cox Transformation Parameter的套件及指令，via input values (代值法)、MLE(最大概似估計法)、Normality Tests (常態檢定): `boxcoxnc {AID}`、`boxcox {MASS}`、`powerTransform {car}`、`find_lambda {rust}`、`BoxCox.lambda {forecast}`

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.)

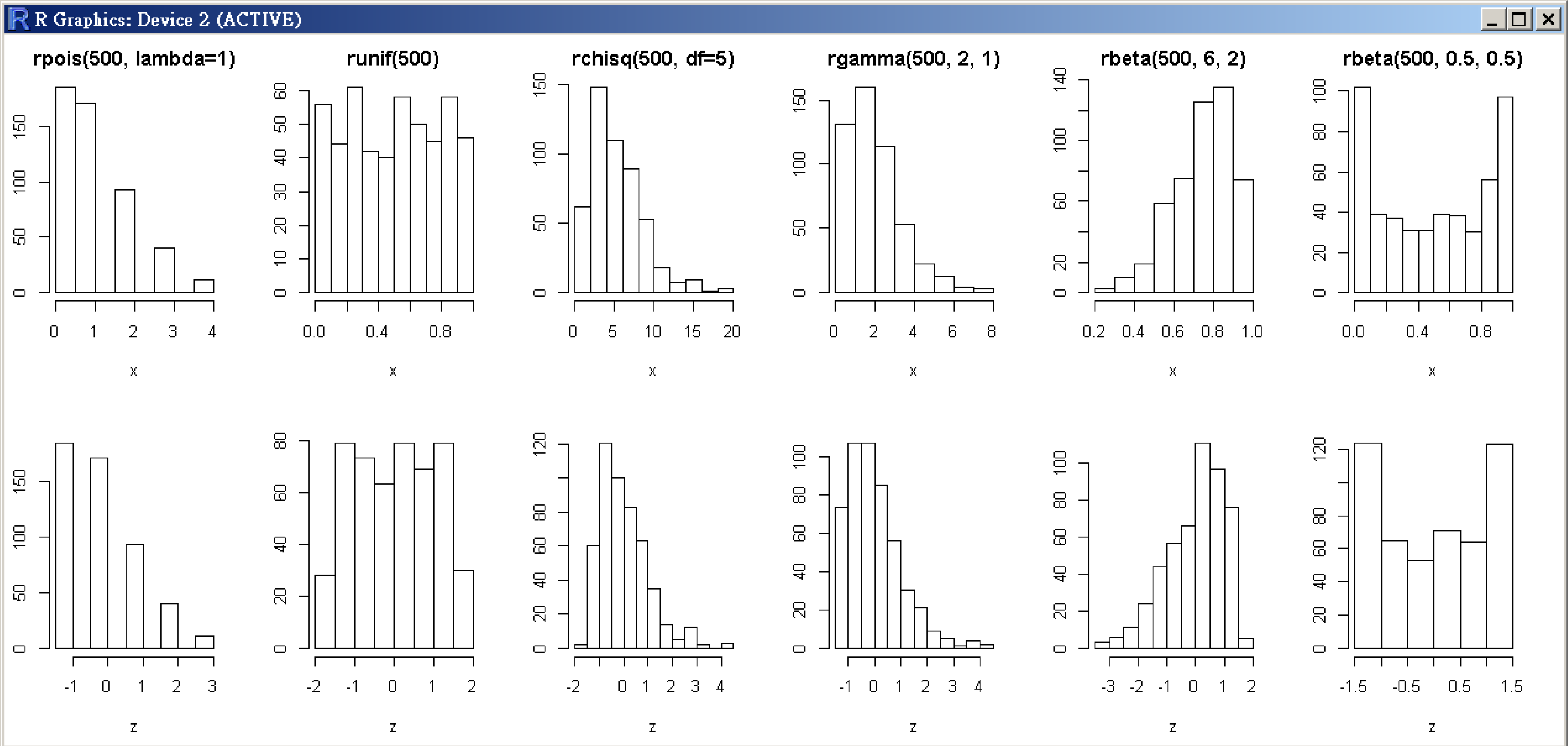
$$z_i = \frac{x_i - \bar{x}}{s}$$

- 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

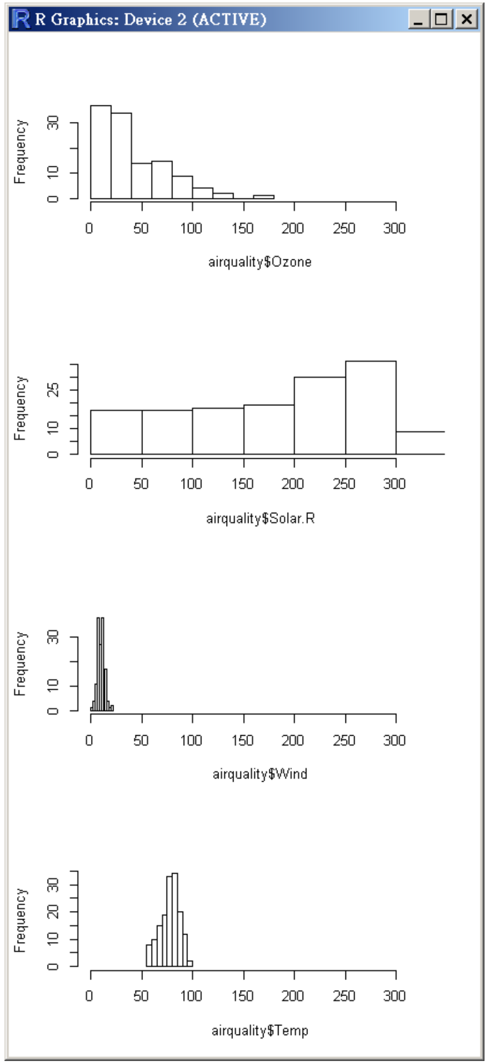
airquality {datasets}

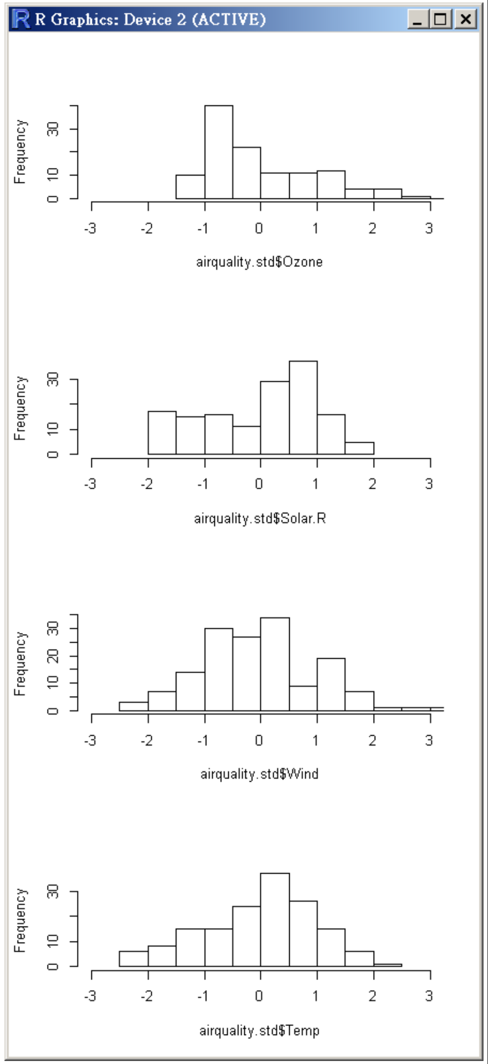
New York Air Quality Measurements: Daily air quality measurements in New York, May to September 1973.

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: Crab Data

crabs {MASS}

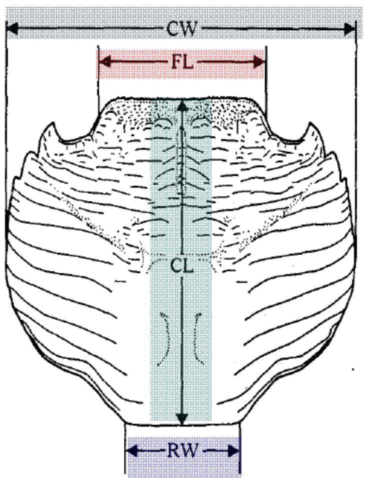
Morphological Measurements on Leptograpsus Crabs

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)
```

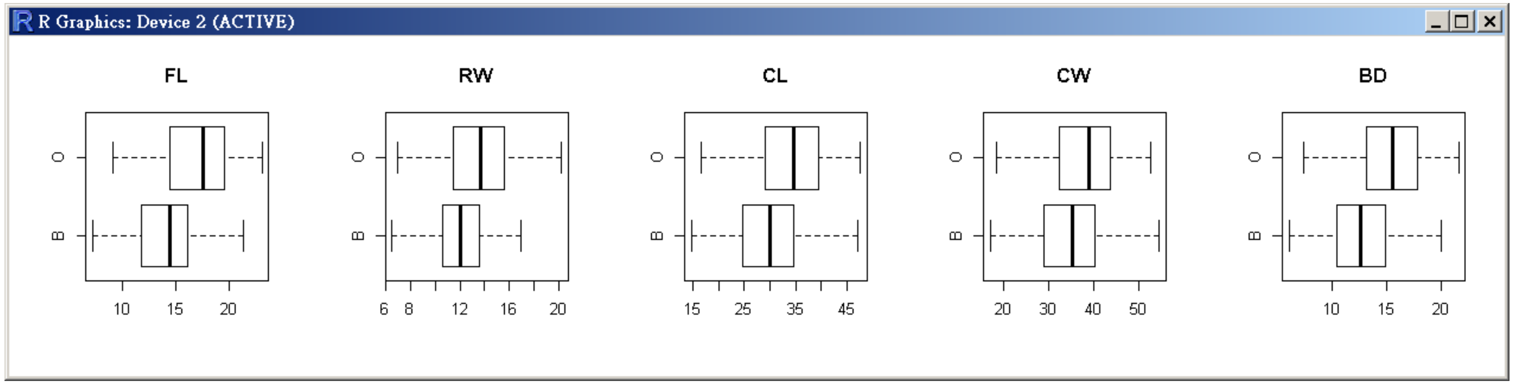


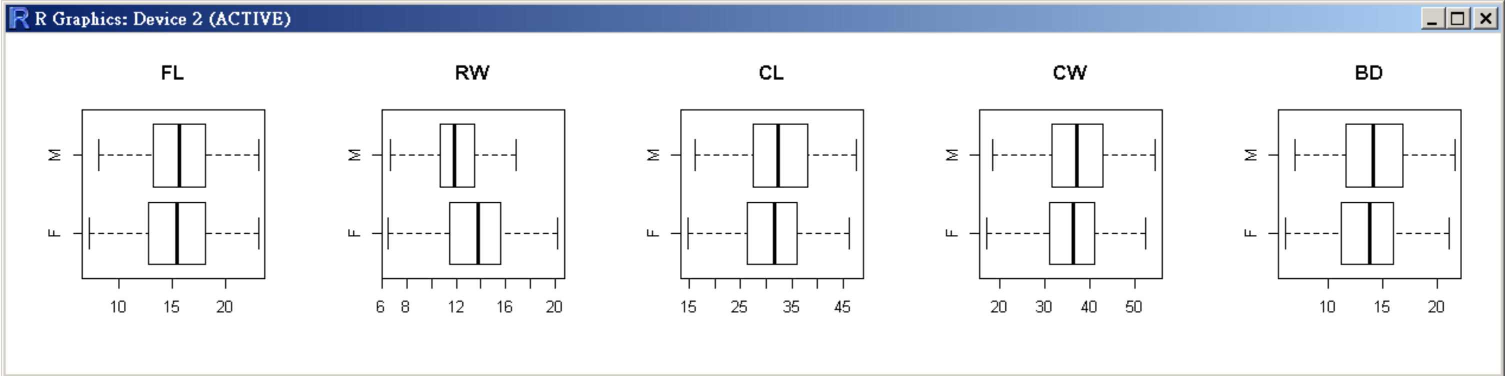
Aust. J. Zool. 1974, 22, 417-25

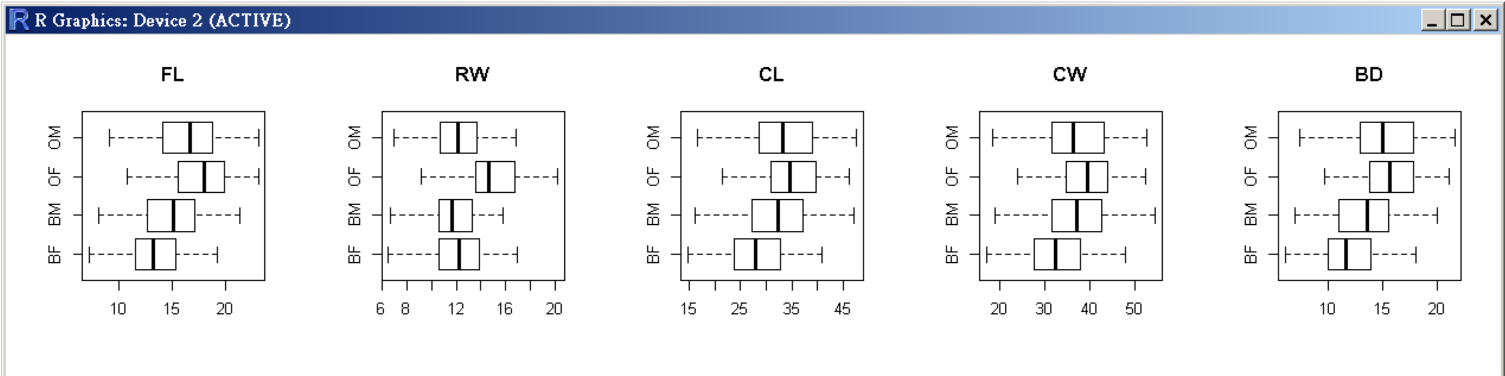
http://www.qm.qld.gov.au/Find+out+about/Animals+of+Queensland/Crustaceans/Common+marine+crustaceans/Crabs/Purple+Swift-footed+Shore+Crab#.VhPWYiurFhs

範例: Crab Data

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





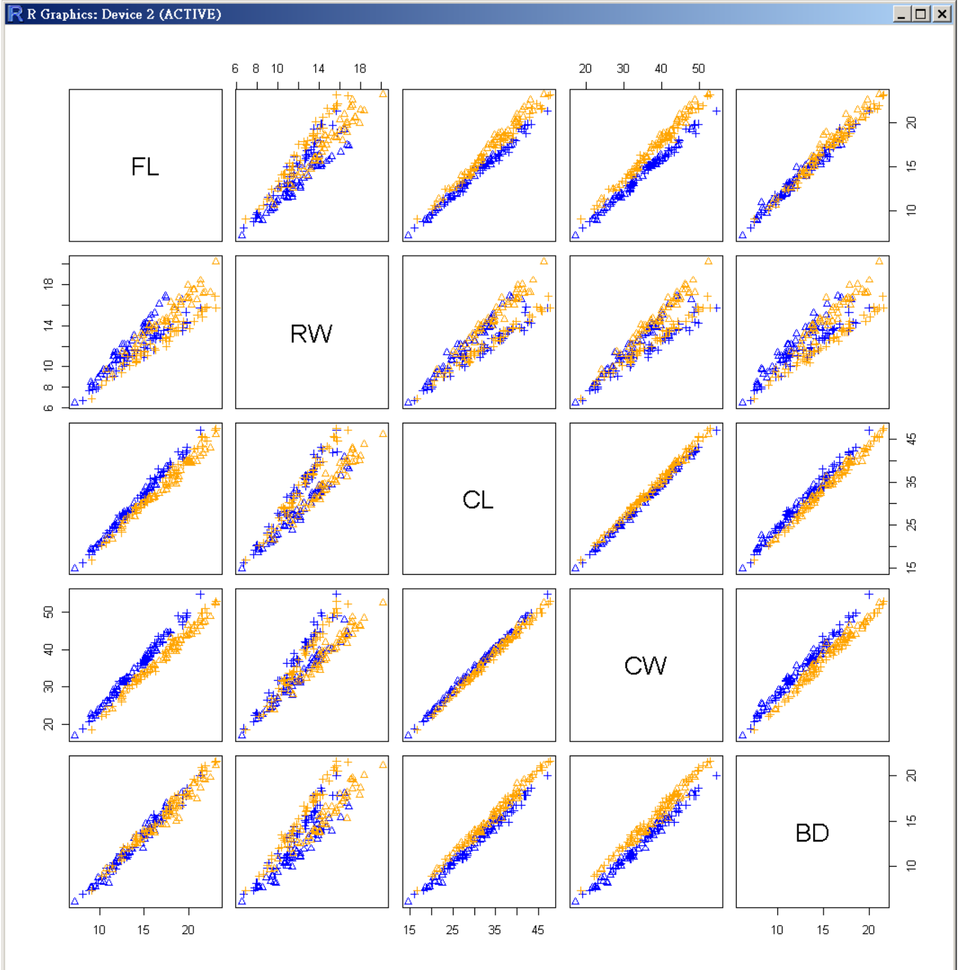


範例: Crab Data

```
# 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.

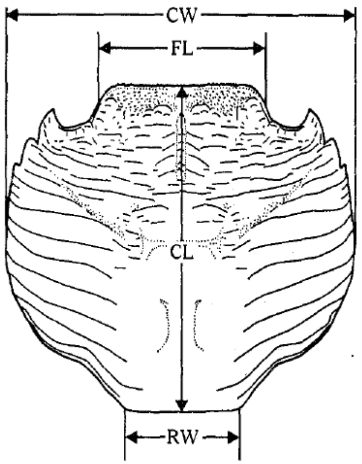
(Hannes Baur and Christoph Leuenberger, Analysis of Ratios in Multivariate Morphometry, Systematic Biology 60(6), 813-825.)

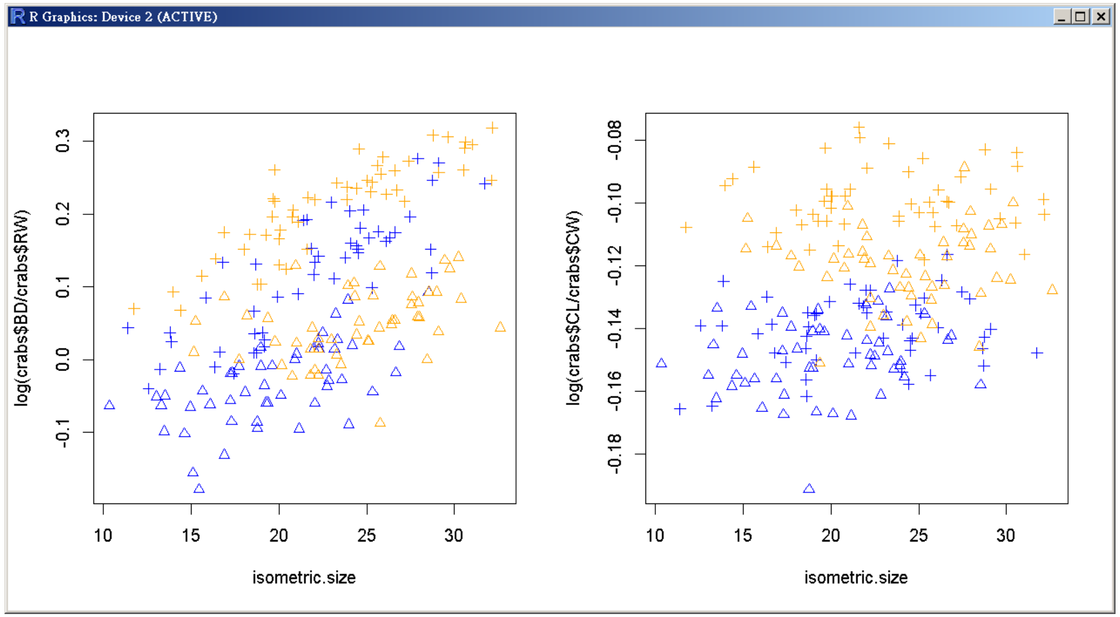


範例: Crab Data

- 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.

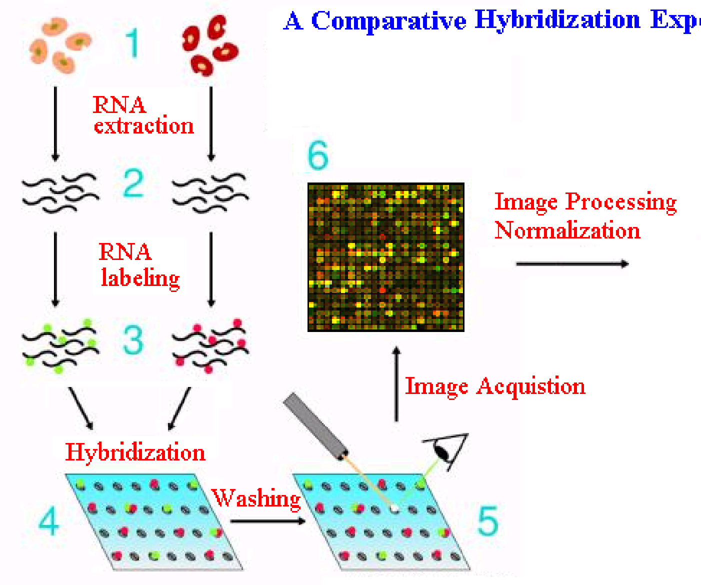





```
par(mfrow=c(1,2))
mp <- as.integer(crabs$sex)+1
mc <- c("blue","orange")[as.integer(crabs$sp)]
isometric.size <- apply(crabs[,4:8], 1, mean)
plot(isometric.size,  log(crabs$BD/crabs$RW), pch=mp, col=mc)
plot(isometric.size, log(crabs$CL/crabs$CW), pch=mp, col=mc)
```

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

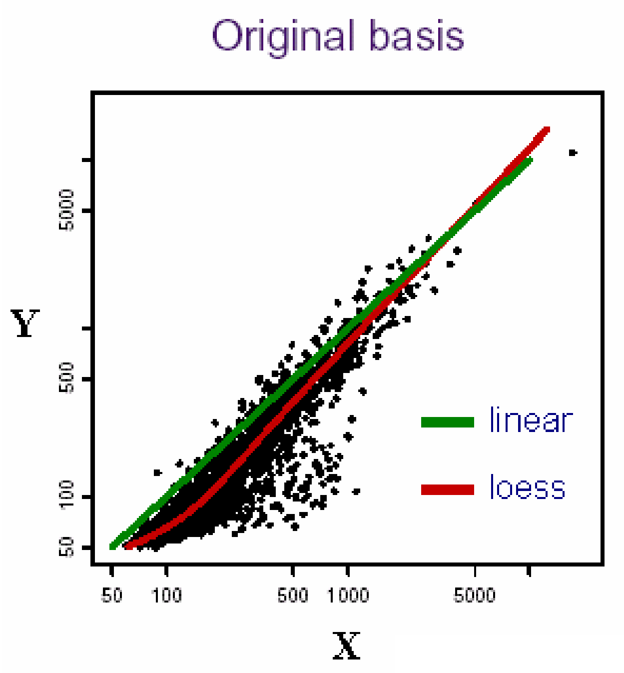
Matrix of genes (rows) and samples (columns)

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
- ...

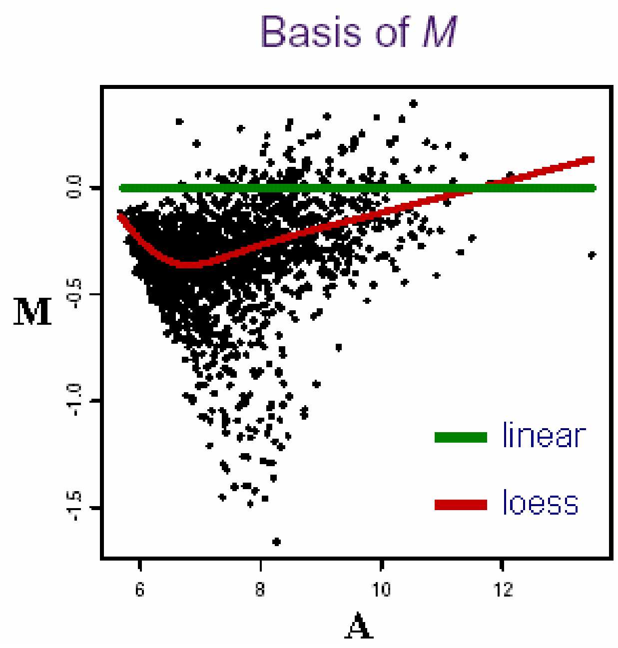
MA plot and Loess (Lowess) Normalization



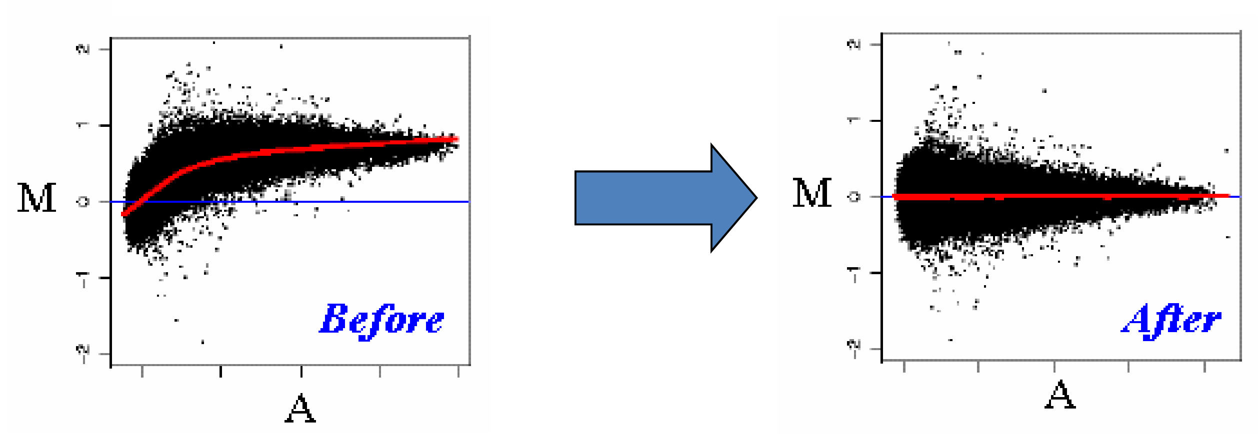
$$M = \log_2\left(\frac{Y}{X}\right)$$

$$A = \frac{1}{2}\log_2(XY)$$

Oligo	cDNA
$X = PM_1$, $Y = PM_2$	X= Cy3
$X = PM_1 - MM_1$, $Y = PM_2 - MM_2$	Y= Cy5



$$\log_2 R/G \Rightarrow \log_2 R/G - c(A) = \log_2 R/[k(A)G].$$



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.

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

- 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.

在統計學和機器學習的資料分析中，選擇哪種資料轉換方式主要取決於數據的特性和所使用的模型。以下是一些常見的資料轉換方式：

1. **標準化（Standardization）**：這種轉換方式將數據轉換為均值為0，標準差為1的分佈。這種轉換方式對於許多機器學習算法（如支持向量機和邏輯回歸）都是必要的。
2. **正規化（Normalization）**：這種轉換方式將數據的範圍轉換到[0, 1]或者[-1, 1]。這種轉換方式對於神經網絡和基於距離的算法（如k-近鄰）非常有用。
3. **對數轉換（Log Transformation）**：這種轉換方式對於處理偏態數據非常有用。它可以將長尾分佈轉換為更接近正態分佈的形式。
4. **Box-Cox轉換或Yeo-Johnson轉換**：這兩種轉換方式都是對數轉換的擴展，可以對數據進行更靈活的轉換，以達到更接近正態分佈的目的。
5. **獨熱編碼（One-Hot Encoding）**：這種轉換方式主要用於處理分類變數。它將分類變數轉換為一系列的二進制變數，每個變數代表一個類別。
6. **標籤編碼（Label Encoding）**：這種轉換方式也是用於處理分類變數，但它將每個類別轉換為一個整數。這種轉換方式對於某些基於樹的算法（如決策樹和隨機森林）可能更有效。

選擇哪種轉換方式取決於你的數據和模型。在實際應用中，可能需要嘗試多種轉換方式，以找到最適合你的數據和模型的轉換方式。