第 16 章: 列聯表函式

16: Cross Tabulation Function

在類別資料分析中, 常常會使用到 列聯表 (contingency table), 在 R 中, 一些函式用來製造或操作 列聯表 (contingency table), 例如, `table()`, `xtabs()`, `as.table()`, `is.table()`; `ftable()`, `read.ftable()`, `wirte.ftable()`; `as.data.frame()`; `margin.table()`, `prop.table()`, `addmargins()` 等. 除此之外, R 還有一些關於列聯表的套件, 例如, `xtable`, `vcd`, `reshape2`, `plyr`, `dplyr`, `tidyr`, `tidyverse` 等, 將資料產生或轉換成列聯表. R 本身有一些分析列聯表的函式, 也有許多許多流行病學的套件與函式, 對 列聯表 進行分析. 例如, `Epi`, `epibasix`, `epiDisplay` (之前爲 `epicalc`), `epifit`, `epiR`, `epitools`, `RCOR`, `pROC` 等, 可以進行流行病學分析, 套件主要分析能力大至相同, 但各別套件仍有其特徵.

16.1 類別資料型式

類別資料的輸入常見有 2 種型式, (a) 個別資料 (individual data, micro data, case data); (b) 聚集資料 (aggregated data, macro data, summarized data, ecological data). 個別資料內包含每一位 個體 (subject, individual), 研究者爲目前的研究目的所蒐集的第一手資料, 記錄著每一爲個體的測量值, 個別資料有時稱爲 原始資料 (raw

data, primary data, original data). 資料只有摘要之後的結果, 是由其他來源所得到的資料, 沒有每一位個體的比變數觀測值, 例如沒有每一位個體的性別, 年紀等觀測值, 這種整理分析候的資料有時稱爲 二手資料 或 次級資料 (secondary data).

個別資料可以經過整理成爲聚集資料形態, 但是, 若遺失每一位個體的測量值, 只有聚集資料, 則無法回復原來的個別資料. 因此, 研究盡量使用每一位個體的測量值, 但有些時候, 無法得到每一位個體的測量值, 若資料內全是類別變數, 則 個別資料 與 聚集資料 分析結論部會有差別, 但資料內若有連續變數如年紀, BMI 等, 將連續變數轉換成類別變數, 只呈現或分析聚集資料, 則會造成所謂的 生態謬誤 (ecological fallacy), 是指由 聚集資料 (團體) 所得到的推論, 不能反應 個別資料 (個人的真實) 所得到的推論, 所產生研究推論的誤導.

例 **16.1.** 小細胞肺癌臨床試驗

一個關於小細胞肺癌臨床試驗研究, 探討合併多種化學藥物治療的給藥方式對治療結果的影響. 病患依性別隨機分成 2 組: sequence 組 與 alternating 組, sequence 組 在每 1 治療療程中, 只給固定 1 種合併化學藥物, alternating 組 有 3 種不同合併化學藥物組合, 在每 1 治療療程中, 交替使用不同合併化學藥物組合, 反應變數有 4 個類別: progressive disease, no change, partial remission, complete remission (繼續惡化, 無變化, 部分緩解, 完全緩解), 其中定義 progressive 與 no change 是治療失敗 (fail), partial remission 與 complete remission 是治療成功 (success), 研究結果呈現在表 16.1, 變數定義在表 16.2, 資料在檔案 CateLungcSsmallCellCombAgg.csv, CateLungcSsmallCellCombInd.csv.

表 16.1: 小細胞肺癌臨床試驗

		Response				Outcome	
Treatment	Gender	Progress Disease	No Change	Partial Remission	Complete Remission	no	yes
sequence	male	28	45	29	26	73	55
	female	4	12	5	2	16	8
alternating	male	41	44	20	20	85	40
	female	12	7	3	1	19	44

表 16.2: 小細胞肺癌臨床試驗: 變數名稱與定義

變數	說明
treat	治療組別: 0 = sequence, 1 = alternating
gender	性別: 0 = 男性, 1 = 女性
response	治療反應:
	1 = progressive disease, 2 = no change,
	3 = partial remission, 4 = complete remission
outcome	治療結果: 0 = fail, 1 = success

16.2 列聯表函式: table(), xtabs()

使用函式 `table()`, `xtabs()`, 可以從任何向量, 矩陣, 陣列, 資料框架創造一個列聯表, 回傳一個 列聯表 物件. 函式 `table()` 回傳的物件稱爲 `_contingency table_`. 這是一個 R 物件類別 (class) 爲 `table` 之特殊物件. 使用函式 `as.table()` 可用來強制將 矩陣 或 資料框架 形成列聯表物件. `as.matrix()` 強制將 列聯表物件 形成 矩陣. `as.data.frame()` 強制將 列聯表物件 形成 資料框架. `as.data.frame()` 是 `xtabs()` 的反函式. 使用函式 `is.table()` 可查看物件是否爲列聯表物件.

```
> table(...,
        exclude = if (useNA = =  "no") c(NA, NaN),
        useNA = c("no", "ifany", "always"),
        dnn = list.names(...), deparse.level = 1)

> xtabs(formula = ∼., data = parent.frame(), subset, sparse = FALSE,
        na.action, exclude = c(NA, NaN), drop.unused.levels = FALSE)

> as.data.frame(x, row.names = NULL, ...,
                responseName = "Freq", stringsAsFactors = TRUE,
                sep = "", base = list(LETTERS))

> is.table(x)
```

其中引數

- formula: 使用統計模型公式輸入.
- data: 資料框架名.
- drop.unused.levels = FALSE: 不排除類別水準的計數 (count) 爲 0, 會造成 卡方檢定的錯誤訊息.
- na.action = "na.omit": 缺失值處理方式.
- exclude: 排除類別水準的細項, 自動內設排除缺失值.
- useNA: 處理缺失值選項.
  - "no": 排除缺失值.
  - "ifany": 納入缺失值, 若類別水準的計數 (count) 爲正整數.
  - "always": 永遠納入缺失值成爲 1 類別水準.
    即使類別水準的計數 (count) 爲 0 仍然自成 1 個類別水準.
- dnn: dimnames names, 對回傳 table 物件的個別維度命名.
- deparse.level: 若 dnn = NULL, R 會對回傳 table 物件的個別維度命名.
  - deparse.level = 0: 個別維度沒有命名, dnn = c("", "").

  - deparse.level = 1: 維度命名僅列位爲原有列位變數名, dnn = c("var1.name", "").
  - deparse.level = 2: 個別維度命名爲原有變數名, dnn = c("var1.name", "var2.name").

- row.names: 對 as.data.frame() 回傳的 data frame 的列位命名 (row name).
- responseName = "Freq": 對 as.data.frame() 回傳的計數變數命名爲 "Freq" (count).
- stringsAsFactors: 對 as.data.frame() 回傳的 data frame 變數設成因子變數.

使用函式 xtabs() 可以從資料框架中, 利用統計模型公式 (model formula) 創造一個列聯表. 函式 as.data.frame() 則是函式 xtabs() 的反函式, 從列聯表物件創造一個資料框架.

```
> ## Small Cell Lung Cancer Clinical Trial
> dd = read.table(file = "CateLungcSsmallCellCombInd.csv",
                  header = TRUE, sep = ",",
                  dec = ".",  row.names = NULL)
> head(dd)
  treat gender outcome response count
1     0      0       0        1     1
2     0      0       0        1     1
3     0      0       0        1     1
4     0      0       0        1     1
5     0      0       0        1     1
6     0      0       0        1     1
> #
> dim(dd)
[1] 299   5
> #
> str(dd)
'data.frame':   299 obs. of  5 variables:
 $ treat   : int  0 0 0 0 0 0 0 0 0 0 ...
 $ gender  : int  0 0 0 0 0 0 0 0 0 0 ...
 $ outcome : int  0 0 0 0 0 0 0 0 0 0 ...
 $ response: int  1 1 1 1 1 1 1 1 1 1 ...
 $ count   : int  1 1 1 1 1 1 1 1 1 1 ...
```

```
> #
> ## one-way table
> table(dd$response)

  1   2   3   4
 85 108  57  49
> dd$response = factor(dd$response)
> levels(dd$response) = c("progress", "nochange",
+                         "parital", "complete")
> #
> table(dd$response)

progress nochange  parital complete
      85      108       57       49
>
> ## one-way table
> table(dd$outcome)

  0   1
193 106
> dd$outcome = factor(dd$outcome)
> levels(dd$outcome) = c("fail", "success")
> table(dd$outcome)

   fail success
    193     106
>
> ## one-way table
> table(dd$treat)

  0   1
151 148
> dd$treat = factor(dd$treat)
> levels(dd$treat) = c("seq", "alt")
> table(dd$treat)

seq alt
151 148
>
> ## one-way table()
> table(dd$gender)

  0   1
253  46
> dd$gender = factor(dd$gender)
> levels(dd$gender) = c("male", "female")
> table(dd$gender)

  male female
   253     46
```

```
>
> ## two-way table()
> table(dd$response, dd$outcome)

           fail success
  progress   85       0
  nochange  108       0
  parital     0      57
  complete    0      49
> #
> t2.tab <- table(dd$treat, dd$response)
> t2.tab

     progress nochange parital complete
  seq       32       57      34       28
  alt       53       51      23       21
> #
> class(t2.tab)
[1] "table"
> #
> typeof(t2.tab)
[1] "integer"
> #
> str(t2.tab)
 'table' int [1:2, 1:4] 32 53 57 51 34 23 28 21
 - attr(*, "dimnames") = List of 2
  ..$ : chr [1:2] "seq" "alt"
  ..$ : chr [1:4] "progress" "nochange" "parital" "complete"
> #
> attributes(t2.tab)
$dim
[1] 2 4

$dimnames
$dimnames[[1]]
[1] "seq" "alt"

$dimnames[[2]]
[1] "progress" "nochange" "parital"  "complete"

$class
[1] "table"
>
> ## deparse.level
> table(dd$treat, dd$response, deparse.level = 0)

     progress nochange parital complete
  seq       32       57      34       28
  alt       53       51      23       21
> #
```

```
> table(dd$treat, dd$response, deparse.level = 1)

      progress nochange parital complete
  seq       32       57      34       28
  alt       53       51      23       21
> #
> table(dd$treat, dd$response, deparse.level = 2)
        dd$response
dd$treat progress nochange parital complete
     seq       32       57      34       28
     alt       53       51      23       21
> #
> ## one-way table
> table(dd$response)

progress nochange  parital complete
      85      108       57       49
> dd$response = factor(dd$response)
> levels(dd$response) = c("progress", "nochange",
                          "parital", "complete")
> table(dd$response)

progress nochange  parital complete
      85      108       57       49
>
> ## one-way table
> table(dd$outcome)

   fail success
    193     106
> dd$outcome = factor(dd$outcome)
> levels(dd$outcome) = c("fail", "success")
> table(dd$outcome)

   fail success
    193     106
>
> ## one-way table
> table(dd$treat)

seq alt
151 148
> dd$treat = factor(dd$treat)
> levels(dd$treat) = c("seq", "alt")
> table(dd$treat)

seq alt
151 148
>
> ## one-way table()
```

```
> table(dd$gender)

  male female
   253     46
> dd$gender = factor(dd$gender)
> levels(dd$gender) = c("male", "female")
> table(dd$gender)

  male female
   253     46
>
> ## two-way table()
> table(dd$response, dd$outcome)

           fail success
  progress   85       0
  nochange  108       0
  parital     0      57
  complete    0      49
> t2.tab <- table(dd$treat, dd$response)
> t2.tab

      progress nochange parital complete
  seq       32       57      34       28
  alt       53       51      23       21
> #
> class(t2.tab)
[1] "table"
> #
> typeof(t2.tab)
[1] "integer"
> #
> str(t2.tab)
 'table' int [1:2, 1:4] 32 53 57 51 34 23 28 21
 - attr(*, "dimnames") = List of 2
  ..$ : chr [1:2] "seq" "alt"
  ..$ : chr [1:4] "progress" "nochange" "parital" "complete"
> #
> attributes(t2.tab)
$dim
[1] 2 4

$dimnames
$dimnames[[1]]
[1] "seq" "alt"

$dimnames[[2]]
[1] "progress" "nochange" "parital"  "complete"

$class
```

```
[1] "table"
>
> ## deparse.level
> table(dd$treat, dd$response, deparse.level = 0)

     progress nochange parital complete
  seq       32       57      34       28
  alt       53       51      23       21

> table(dd$treat, dd$response, deparse.level = 1)

     progress nochange parital complete
  seq       32       57      34       28
  alt       53       51      23       21

> table(dd$treat, dd$response, deparse.level = 2)
        dd$response
dd$treat progress nochange parital complete
     seq       32       57      34       28
     alt       53       51      23       21
> #
> ## one-way xtabs()
> table(dd$response)

progress nochange  parital complete
      85      108       57       49

> xtabs(~ response, data = dd)
response
progress nochange  parital complete
      85      108       57       49
>
> ## two-way xtabs()
> table(dd$response, dd$outcome)

           fail success
  progress   85       0
  nochange  108       0
  parital     0      57
  complete    0      49
>
> xtabs(~ response + outcome, data = dd)
          outcome
response   fail success
  progress   85       0
  nochange  108       0
  parital     0      57
  complete    0      49
>
> xtabs(count ~ response + outcome, data = dd)
```

```
          outcome
response   fail success
  progress   85       0
  nochange  108       0
  parital     0      57
  complete    0      49
>
> xtabs(count ~ treat + gender, data = dd)
    gender
treat male female
  seq  128     23
  alt  125     23
> #
> ## three-way table()
> t3.tab = table(dd$treat, dd$response, dd$gender)
> t3.tab
, , = male
      progress nochange parital complete
  seq       28       45      29       26
  alt       41       44      20       20

, , = female
      progress nochange parital complete
  seq        4       12       5        2
  alt       12        7       3        1

> class(t3.tab)
[1] "table"
> typeof(t3.tab)
[1] "integer"
> str(t3.tab)
 'table' int [1:2, 1:4, 1:2] 28 41 45 44 29 20 26 20 4 12 ...
 - attr(*, "dimnames") = List of 3
  ..$ : chr [1:2] "seq" "alt"
  ..$ : chr [1:4] "progress" "nochange" "parital" "complete"
  ..$ : chr [1:2] "male" "female"
> attributes(t3.tab)
$dim
[1] 2 4 2

$dimnames
$dimnames[[1]]
[1] "seq" "alt"

$dimnames[[2]]
[1] "progress" "nochange" "parital"  "complete"

$dimnames[[3]]
[1] "male"   "female"

```

```
$class
[1] "table"
> #
> ## three-way xtabs()
> table(dd$treat, dd$outcome, dd$gender)
, , = male
     fail success
  seq   73      55
  alt   85      40

, , = female
     fail success
  seq   16       7
  alt   19       4

> dd.xtab = xtabs(~ treat + outcome + gender, data = dd)
> dd.xtab
, , gender = male
    outcome
treat fail success
  seq   73      55
  alt   85      40

, , gender = female
    outcome
treat fail success
  seq   16       7
  alt   19       4
> #
> class(dd.xtab)
[1] "xtabs" "table"
> #
> typeof(dd.xtab)
[1] "integer"
> #
> str(dd.xtab)
 int [1:2, 1:2, 1:2] 73 85 55 40 16 19 7 4
 - attr(*, "dimnames") = List of 3
  ..$ treat  : chr [1:2] "seq" "alt"
  ..$ outcome: chr [1:2] "fail" "success"
  ..$ gender : chr [1:2] "male" "female"
 - attr(*, "class") =  chr [1:2] "xtabs" "table"
 - attr(*, "call") =  language xtabs(formula = ~treat + outcome + gender, data = dd)
> #
> attributes(dd.xtab)
$dim
[1] 2 2 2

$dimnames
$dimnames$treat
```

```
[1] "seq" "alt"

$dimnames$outcome
[1] "fail"    "success"

$dimnames$gender
[1] "male"   "female"

$class
[1] "xtabs" "table"

$call
xtabs(formula = ~treat + outcome + gender, data = dd)
> #
> is.table(dd.xtab)
[1] TRUE
> #
> xtabs(count ~ treat + outcome + gender, data = dd)
, , gender = male

      outcome
treat fail success
  seq   73      55
  alt   85      40

, , gender = female

      outcome
treat fail success
  seq   16       7
  alt   19       4

> cd.xtab = xtabs(count ~ treat + response + gender, data = dd)
> cd.xtab
, , gender = male

      response
treat progress nochange parital complete
  seq       28       45      29       26
  alt       41       44      20       20

, , gender = female

      response
treat progress nochange parital complete
  seq        4       12       5        2
  alt       12        7       3        1
>
> class(cd.xtab)
[1] "xtabs" "table"
```

```
>
> typeof(cd.xtab)
[1] "integer"
>
> str(cd.xtab)
 int [1:2, 1:4, 1:2] 28 41 45 44 29 20 26 20 4 12 ...
 - attr(*, "dimnames") = List of 3
  ..$ treat   : chr [1:2] "seq" "alt"
  ..$ response: chr [1:4] "progress" "nochange" "parital" "complete"
  ..$ gender  : chr [1:2] "male" "female"
 - attr(*, "class") =  chr [1:2] "xtabs" "table"
 - attr(*, "call") =  language xtabs(formula = count ~ treat + response + gender, data =
    dd)
>
> attributes(cd.xtab)
$dim
[1] 2 4 2

$dimnames
$dimnames$treat
[1] "seq" "alt"

$dimnames$response
[1] "progress" "nochange" "parital"  "complete"

$dimnames$gender
[1] "male"   "female"

$class
[1] "xtabs" "table"

$call
xtabs(formula = count ~ treat + response + gender, data = dd)
> #
> is.table(cd.xtab)
[1] TRUE
> #
> ## as.data.frame()
> dd.df = as.data.frame(dd.xtab, responseName = "Freq")
> dd.df
  treat outcome gender Freq
1   seq    fail   male   73
2   alt    fail   male   85
3   seq success   male   55
4   alt success   male   40
5   seq    fail female   16
6   alt    fail female   19
7   seq success female    7
8   alt success female    4
> #
```

```
> cd.df = as.data.frame(cd.xtab, responseName = "count")
> cd.df
   treat response gender count
1    seq progress   male    28
2    alt progress   male    41
.............................
15   seq complete female     2
16   alt complete female     1
```

16.3 列聯表函式: ftable(), read.ftable(), wirte.ftable()

函式 `table()` 或 `xtabs()` 對高維度列聯表的呈現類似 list 形式, 較不方便操作, 改使用函式 `ftable()`, 可以從任何向量, 矩陣, 陣列, 資料框架創造一個 扁平列聯表 (flat contingency table), 扁平列聯表 是一個 "`ftalbe`" 類別 (class) 的矩陣物件, 其中變數 (欄位, column) 爲分類因子變數, 另外再加上各組頻率數目, 每一列 (row) 代表每一種分類的類別水準 (level). 使用函式 `ftable()` 得到 `ftable` 物件, 在 R 的列印上, 會比 `table()` 或 `xtabs()` 得到 `_contingency table_` 物件好看. 使用函式 `read.ftable()` 與 `write.ftable()`, 可以用來讀寫 "`ftalbe`" 的扁平列聯表矩陣物件.

```
> ftable(..., exclude = c(NA, NaN), row.vars = NULL,
         col.vars = NULL)

> read.ftable(file, sep = "", quote = "\"",
              row.var.names, col.vars, skip = 0)

> write.ftable(x, file = "", quote = TRUE, append = FALSE,
               digits = getOption("digits"), ...)
```

其中引數

- `exclude`: 排除類別水準的細項, 自動內設排除缺失值.
- `row.vars`: 向量, 選定 flat contingency table 列位名.

- col.vars: 向量, 選定 flat contingency table 變數名 (欄位名).
- dnn: dimnames names, 對回傳 table 物件的個別維度命名.
- row.var.names: 設定讀入 ftable 資料列位名.

```
> ## ftable()
> head(dd)
  treat gender outcome response count
1   seq   male    fail progress     1
2   seq   male    fail progress     1
3   seq   male    fail progress     1
4   seq   male    fail progress     1
5   seq   male    fail progress     1
6   seq   male    fail progress     1
> #
> t1.ftab = ftable(dd$treat, dd$gender, dd$outcome)
> t1.ftab
            fail success
seq male      73      55
    female    16       7
alt male      85      40
    female    19       4
>
> t2.ftab = ftable(data = dd, row.vars = c("treat", "gender"),
                   col.vars = c("outcome"))
> t2.ftab
             outcome fail success
treat gender
seq   male             73      55
      female           16       7
alt   male             85      40
      female           19       4
> #
> t3.ftab = ftable(data = dd, row.vars = c("treat", "gender"),
                   col.vars = c("response"))
> t3.ftab
             response progress nochange parital complete
treat gender
seq   male                 28       45      29       26
      female                4       12       5        2
alt   male                 41       44      20       20
      female               12        7       3        1
> #
> is.table(t3.ftab)
[1] FALSE
> #
> class(t3.ftab)
[1] "ftable"
```

```
> #
> typeof(t3.ftab)
[1] "integer"
> #
> str(t3.ftab)
 'ftable' int [1:4, 1:4] 28 4 41 12 45 12 44 7 29 5 ...
 - attr(*, "row.vars") = List of 2
  ..$ treat : chr [1:2] "seq" "alt"
  ..$ gender: chr [1:2] "male" "female"
 - attr(*, "col.vars") = List of 1
  ..$ response: chr [1:4] "progress" "nochange" "parital" "complete"
> #
> attributes(t3.ftab)
$dim
[1] 4 4

$row.vars
$row.vars$treat
[1] "seq" "alt"

$row.vars$gender
[1] "male"   "female"

$col.vars
$col.vars$response
[1] "progress" "nochange" "parital"  "complete"

$class
[1] "ftable"
>
> ## write.ftable()
> write.ftable(t3.ftab, file = "LungCAftab.txt", quote = FALSE)
> # check file "C:/RData/LungCAftab.txt"
> #
> write.ftable(t3.ftab, quote = FALSE)
             response progress nochange parital complete
treat gender
seq   male                 28       45      29       26
      female                4       12       5        2
alt   male                 41       44      20       20
      female               12        7       3        1
> #
> write.ftable(t3.ftab, quote = FALSE, method = "row.compact")
treat gender response progress nochange parital complete
seq   male                 28       45      29       26
      female                4       12       5        2
alt   male                 41       44      20       20
      female               12        7       3        1
> #
> write.ftable(t3.ftab, quote = FALSE, method = "col.compact")
```

```
      response progress nochange parital complete
treat gender
seq   male           28       45      29       26
      female          4       12       5        2
alt   male           41       44      20       20
      female         12        7       3        1
> #
> write.ftable(t3.ftab, quote = FALSE, method = "compact")
treat gender  response progress nochange parital complete
seq   male                   28       45      29       26
      female                  4       12       5        2
alt   male                   41       44      20       20
      female                 12        7       3        1
> #
> ## read.ftable()
> t4.ftab = read.ftable(file = "LungCAftab.txt")
> t4.ftab
             response progress nochange parital complete
treat gender
seq   male                  28       45      29       26
      female                 4       12       5        2
alt   male                  41       44      20       20
      female                12        7       3        1
> #
> ## read.ftable()
> t4.ftab = read.ftable(file = "LungCAftab.txt")
> t4.ftab
             response progress nochange parital complete
treat gender
seq   male                  28       45      29       26
      female                 4       12       5        2
alt   male                  41       44      20       20
      female                12        7       3        1
> tc.ftab = ftable(data = dd, row.vars = c("treat", "gender", "outcome"),
                   col.vars = c("count"))
> #
> tc.ftab
                     count  1
treat gender outcome
seq   male   fail          73
             success       55
      female fail          16
             success        7
alt   male   fail          85
             success       40
      female fail          19
             success        4
```

16.4 列聯表函式: margin.table(), prop.table(), addmargins()

使用函式 `margin.table()` 可以使用類別 (class) 爲 `table` 的列聯表物件, 或是使用陣列型式 (array) 的列聯表物件計算 邊際總合 (marginal total). 函式 `prop.table()` 從陣列型式之列聯表物件, 計算列聯表物件的 相對頻率 (relative frequency). 函式 `addmargins()` 對列聯表物件進行邊際維度計算總和.

函式 `margin.table()` 與函式 `prop.table()` 無法對類別 (class) 爲 "`ftable`" 的列聯表物件進行操作. 函式 `addmargins()` 可以對類別 (class) 爲 "`table`" 或 "`ftable`" 的列聯表物件進行操作.

```
> margin.table(x, margin = NULL)
> prop.table(x, margin = NULL)
> addmargins(A, margin = seq_along(dim(A)), FUN = sum, quiet = FALSE)
```

其中引數

- `x`: `table` 物件.
- `A`: `table` 或 `ftable` 物件.
- `margin`: 維度下標或向量 (index/vector), 設定計算的邊際維度.
  - `margin = NULL`: 計算列聯表內 總和 或 個別空格分率 (cell count/proportion).
  - `margin = 1`: 計算列聯表的列位 (row) 邊際總和 或 分率 (row marginal total/proportion).
  - `margin = 2`: 計算列聯表的欄位 (column) 邊際總和 或 分率 (column marginal total/proportion)..
  - `margin = k`: 其餘維度則依此列推.

函式 margin.table() 是 函式 apply(x, margin, sum) 的簡化, 函式 prop.table() 是 函式 sweep(x, margin, margin.table(x, margin), "/") 的簡化. 但是, 函式 margin.table() 與函式 prop.table() 無法對類別 (class) 爲 "ftable" 的列聯表物件進行操作.

```
> head(dd)
  treat gender outcome response count
1   seq   male    fail progress     1
2   seq   male    fail progress     1
3   seq   male    fail progress     1
4   seq   male    fail progress     1
5   seq   male    fail progress     1
6   seq   male    fail progress     1
>
> ## 2-way table
> tr.tab = table(dd$treat, dd$response)
> #
> margin.table(tr.tab) # total count
[1] 299
> prop.table(tr.tab)   # cell proportion

         progress   nochange    parital   complete
  seq 0.10702341 0.19063545 0.11371237 0.09364548
  alt 0.17725753 0.17056856 0.07692308 0.07023411
> #
> margin.table(tr.tab, margin = 1) # row (treat) marginal total

seq alt
151 148
> #
> prop.table(tr.tab, margin = 1)   # row (response) marginal proportion

        progress  nochange   parital  complete
  seq 0.2119205 0.3774834 0.2251656 0.1854305
  alt 0.3581081 0.3445946 0.1554054 0.1418919
> #
> margin.table(tr.tab, margin = 2) # column (treat) marginal total

progress nochange  parital complete
      85      108       57       49
> #
> prop.table(tr.tab, margin = 2)   # column (response) marginal proportion

        progress  nochange   parital  complete
  seq 0.3764706 0.5277778 0.5964912 0.5714286
  alt 0.6235294 0.4722222 0.4035088 0.4285714
> #
> #
```

```
> ## 3-way table
> tgr.tab = table(dd$treat, dd$response, dd$gender)
> prop.table(tgr.tab)   # cell proportion

, , = male
        progress    nochange     parital    complete
  seq 0.093645485 0.150501672 0.096989967 0.086956522
  alt 0.137123746 0.147157191 0.066889632 0.066889632

, , = female
       progress    nochange     parital    complete
  seq 0.013377926 0.040133779 0.016722408 0.006688963
  alt 0.040133779 0.023411371 0.010033445 0.003344482

> #
> margin.table(tgr.tab, margin = 1) # row (treat) marginal total

seq alt
151 148
> #
> prop.table(tgr.tab, margin = 1)   # row (response) marginal proportion

, , = male
        progress    nochange     parital    complete
  seq 0.185430464 0.298013245 0.192052980 0.172185430
  alt 0.277027027 0.297297297 0.135135135 0.135135135

, , = female
        progress    nochange     parital    complete
  seq 0.026490066 0.079470199 0.033112583 0.013245033
  alt 0.081081081 0.047297297 0.020270270 0.006756757

> #
> margin.table(tgr.tab, margin = 2) # column (treat) marginal total

progress nochange  parital complete
      85      108       57       49
> #
> prop.table(tgr.tab, margin = 2)   # column (response) marginal proportion

, , = male
       progress   nochange    parital   complete
  seq 0.32941176 0.41666667 0.50877193 0.53061224
  alt 0.48235294 0.40740741 0.35087719 0.40816327

, , = female
       progress   nochange    parital   complete
  seq 0.04705882 0.11111111 0.08771930 0.04081633
  alt 0.14117647 0.06481481 0.05263158 0.02040816

```

```
> #
> margin.table(tgr.tab, margin = 3)

  male female
   253     46
> #
> prop.table(tgr.tab, margin = 3)

, , = male
       progress   nochange    parital   complete
  seq 0.11067194 0.17786561 0.11462451 0.10276680
  alt 0.16205534 0.17391304 0.07905138 0.07905138

, , = female
       progress   nochange    parital   complete
  seq 0.08695652 0.26086957 0.10869565 0.04347826
  alt 0.26086957 0.15217391 0.06521739 0.02173913

> #
> margin.table(tgr.tab, margin = c(1, 3))

     male female
  seq  128     23
  alt  125     23
```