E02

類別資料視覺化

吳漢銘

國立政治大學 統計學系

https://hmwu.idv.tw

大綱

- **Data format**: individual-level data set, aggregated data, cross tabulation.
- **Visualizing Categorical Data**
  - Bar chart, pie chart, Balloon plot
  - Fourfold Display for 2x2 Tables
  - Association Plots
  - Mosaic Display
- Simple Correspondence Analysis
- Multiple Correspondence Analysis

Individual-level data set, Cross tabulation

Raw Data

```
> HairEyeColor.ind <- read.csv("HairEyeColor_ind.csv")
> head(HairEyeColor.ind)
   Hair   Eye  Sex
1 Black Brown Male
2 Black Brown Male
3 Black Brown Male
4 Black Brown Male
5 Black Brown Male
6 Black Brown Male
> tail(HairEyeColor.ind)
     Hair   Eye    Sex
587 Blond Green Female
588 Blond Green Female
589 Blond Green Female
590 Blond Green Female
591 Blond Green Female
592 Blond Green Female
```

Cross-Tabulated Data

```
> HairEyeColor.tbl <- table(HairEyeColor.ind)
> HairEyeColor.tbl
, , Sex = Female

       Eye
Hair    Blue Brown Green Hazel
  Black    9    36     2     5
  Blond   64     4     8     5
  Brown   34    66    14    29
  Red      7    16     7     7

, , Sex = Male

       Eye
Hair    Blue Brown Green Hazel
  Black   11    32     3    10
  Blond   30     3     8     5
  Brown   50    53    15    25
  Red     10    10     7     7
```

Aggregated Data

```
> HairEyeColor.df <-
as.data.frame(HairEyeColor.tbl)
> HairEyeColor.df
    Hair   Eye    Sex Freq
1  Black  Blue Female    9
2  Blond  Blue Female   64
3  Brown  Blue Female   34
...
30 Blond Hazel   Male    5
31 Brown Hazel   Male   25
32   Red Hazel   Male    7
```

- 表格處理函式: rbind {base}, cbind {base}, table {base}, xtabs {stats}, expand.table {epitools}, tabulate {base}, ftable {stats}, xtable {xtable}, stack {utils} .
- 資料調處相關函式: aggregate {stats}, by {base}, cut {base}, with {base}, merge {base}, split {base}.

B01-1

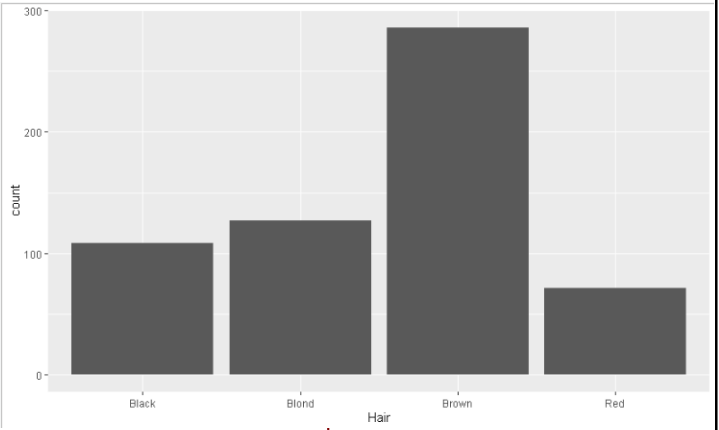
資料處理
(Data Manipulation)

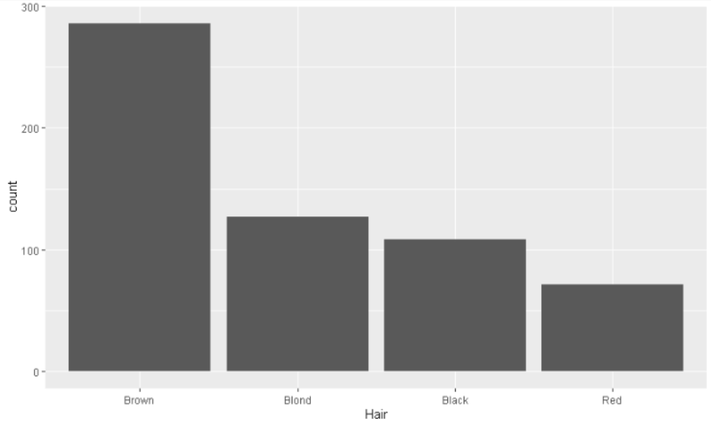
吳漢銘
國立臺北大學 統計學系

http://www.hmwu.idv.tw

Bar plot

```
> library(ggplot2)
> library(dplyr)
> HairEyeColor.ind <- read.csv("HairEyeColor_ind.csv")
> head(HairEyeColor.ind)
   Hair   Eye  Sex
1 Black Brown Male
2 Black Brown Male
3 Black Brown Male
4 Black Brown Male
5 Black Brown Male
6 Black Brown Male
> # geom_bar uses stat = "count" and maps its result to the y aesthetic
> ggplot(HairEyeColor.ind, aes(x = Hair)) + geom_bar()
> library(forcats)
> str(HairEyeColor.ind)
'data.frame':       592 obs. of  3 variables:
 $ Hair: chr  "Black" "Black" "Black" "Black" ...
 $ Eye : chr  "Brown" "Brown" "Brown" "Brown" ...
 $ Sex : chr  "Male" "Male" "Male" "Male" ...
> HairEyeColor.ind.m <- mutate(HairEyeColor.ind, Hair = fct_infreq(Hair))
> str(HairEyeColor.ind.m)
'data.frame':       592 obs. of  3 variables:
 $ Hair: Factor w/ 4 levels "Brown","Blond",..: 3 3 3 3 3 3
 $ Eye : chr  "Brown" "Brown" "Brown" "Brown" ...
 $ Sex : chr  "Male" "Male" "Male" "Male" ...
> ggplot(HairEyeColor.ind.m, aes(x = Hair)) + geom_bar()
```

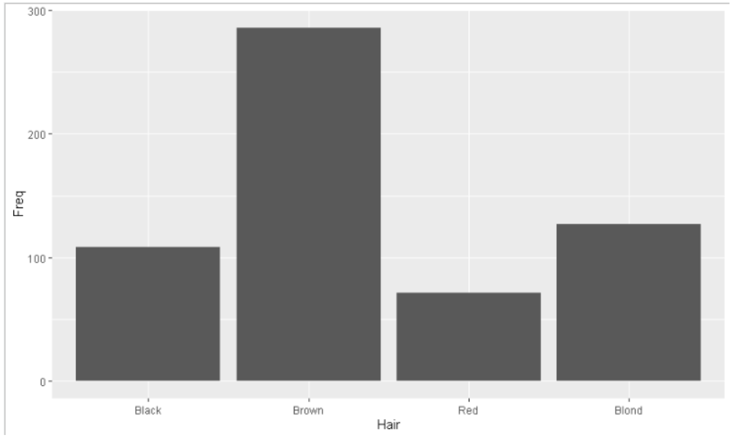




Bar plot

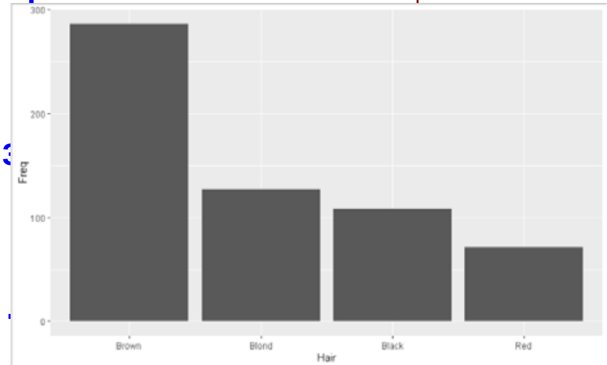
```
> data(HairEyeColor)
> HairEyeColor
, , Sex = Male

       Eye
Hair    Brown Blue Hazel Green
  Black    32   11    10     3
  Brown    53   50    25    15
  Red      10   10     7     7
  Blond     3   30     5     8

, , Sex = Female

       Eye
Hair    Brown Blue Hazel Green
  Black    36    9     5     2
  Brown    66   34    29    14
  Red      16    7     7     7
  Blond     4   64     5     8

> HairEyeColor.df <- as.data.frame(HairEyeColor)
> head(HairEyeColor.df)
   Hair   Eye  Sex Freq
1 Black Brown Male   32
2 Brown Brown Male   53
3   Red Brown Male   10
4 Blond Brown Male    3
5 Black  Blue Male   11
6 Brown  Blue Male   50
```

```
ggplot(HairEyeColor.df, aes(x = Hair, y = Freq)) +
   geom_bar(stat = "identity")
```

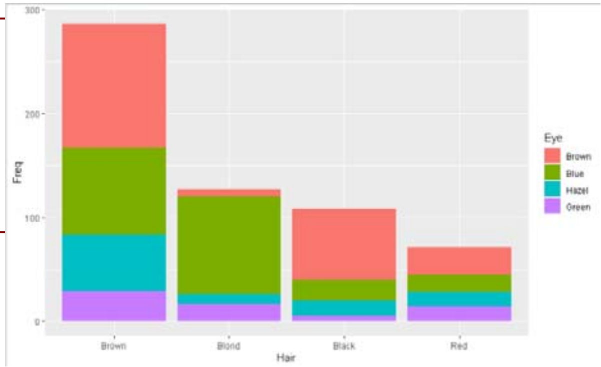


Bar plot

```
> str(HairEyeColor.df)
'data.frame':	32 obs. of  4 variables:
 $ Hair: Factor w/ 4 levels "Black","Brown",..: 1 2 3 4 1 2 3 4 1 2 ...
 $ Eye : Factor w/ 4 levels "Brown","Blue",..: 1 1 1 1 2 2 2 2 3 3 ...
 $ Sex : Factor w/ 2 levels "Male","Female": 1 1 1 1 1 1 1 1 1 1 ...
 $ Freq: num  32 53 10 3 11 50 10 30 10 25 ...
> HairEyeColor.df.m <- mutate(HairEyeColor.df, Hair = reorder(Hair, -Freq, sum),
+                              Eye = reorder(Eye, -Freq, sum))
> str(HairEyeColor.df.m)
'data.frame':	32 obs. of  4 variables:
 $ Hair: Factor w/ 4 levels "Brown","Blond",..: 3 1 4 2 3 1 4 2 3 1
  ..- attr(*, "scores")= num [1:4(1d)] -108 -286 -71 -127
  .. ..- attr(*, "dimnames")=List of 1
  .. .. ..$ : chr [1:4] "Black" "Brown" "Red" "Blond"
 $ Eye : Factor w/ 4 levels "Brown","Blue",..: 1 1 1 1 2 2 2 2 3 3
  ..- attr(*, "scores")= num [1:4(1d)] -220 -215 -93 -64
  .. ..- attr(*, "dimnames")=List of 1
  .. .. ..$ : chr [1:4] "Brown" "Blue" "Hazel" "Green"
 $ Sex : Factor w/ 2 levels "Male","Female": 1 1 1 1 1 1 1 1 1 1 .
 $ Freq: num  32 53 10 3 11 50 10 30 10 25 ...
```

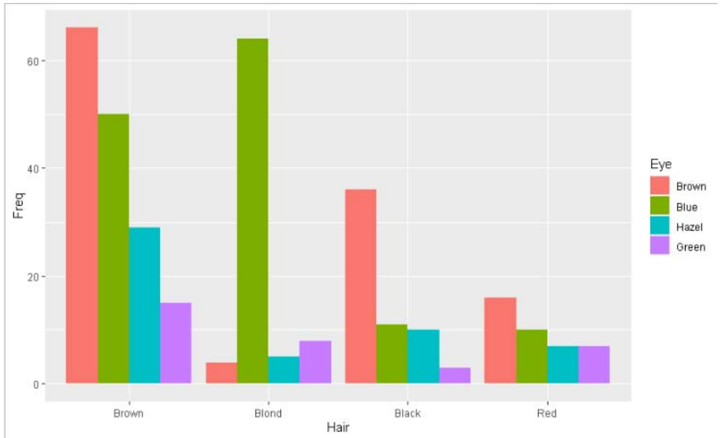


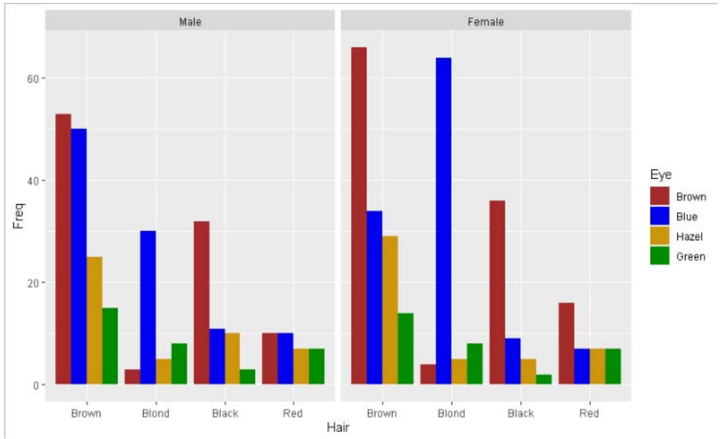
```
ggplot(HairEyeColor.df.m, aes(x = Hair, y = Freq)) +
  geom_bar(stat = "identity")

ggplot(HairEyeColor.df.m, aes(x = Hair, y = Freq, fill = Eye)) +
  geom_bar(stat = "identity")
```

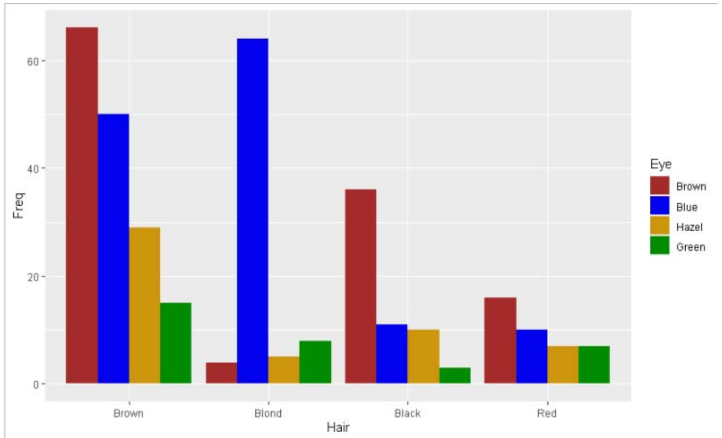


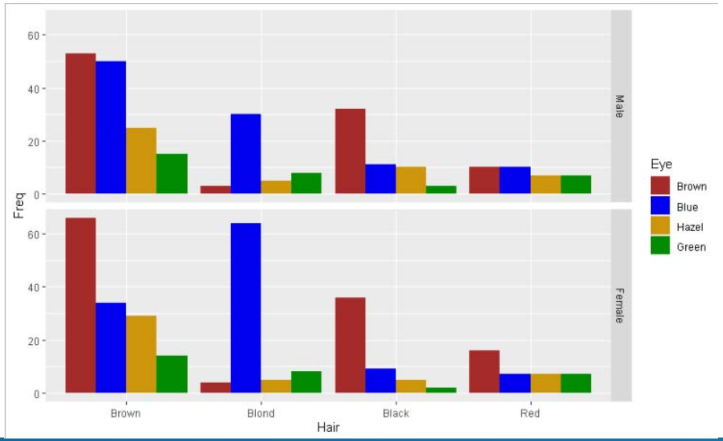
Bar plot

```
p <- ggplot(HairEyeColor.df.m, aes(x = Hair, y = Freq, fill = Eye)) +
  geom_bar(stat = "identity", position = position_dodge(0.9))
p
Eye.col <- c(Brown = "brown", Blue = "blue2", Hazel = "darkgoldenrod3", Green = "green4")
p + scale_fill_manual(values = Eye.col)
p + scale_fill_manual(values = Eye.col) + facet_wrap(~Sex)
p + scale_fill_manual(values = Eye.col) + facet_grid(Sex ~ .)
```





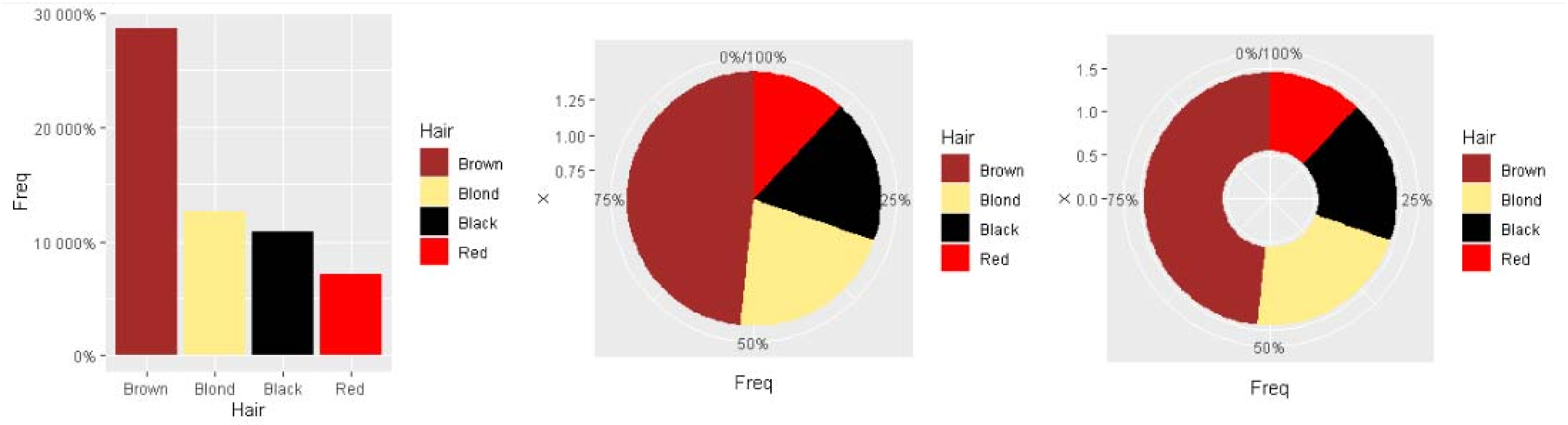




Pie plot

```
Hair.col <- c(Black = "black",  Brown = "brown", Red = "red", Blond = "lightgoldenrod1")

p.bar <- ggplot(HairEyeColor.df.m, aes(x = Hair, y = Freq, fill = Hair)) +
  geom_bar(stat = "identity") +
  scale_fill_manual(values = Hair.col) +
  scale_y_continuous(labels = scales::percent)

library(scales)
p.bar.tmp <- ggplot(HairEyeColor.df.m, aes(x = 1, y = Freq, fill = Hair)) +
  geom_bar(stat = "identity", position = "fill") +
  scale_fill_manual(values = Hair.col)  +
  scale_y_continuous(labels = scales::percent)

p.pie <- p.bar.tmp + coord_polar(theta = "y")

p.doughnut <- p.pie + xlim(0, 1.5)

library(gridExtra)
grid.arrange(p.bar, p.pie, p.doughnut, nrow = 1)
```

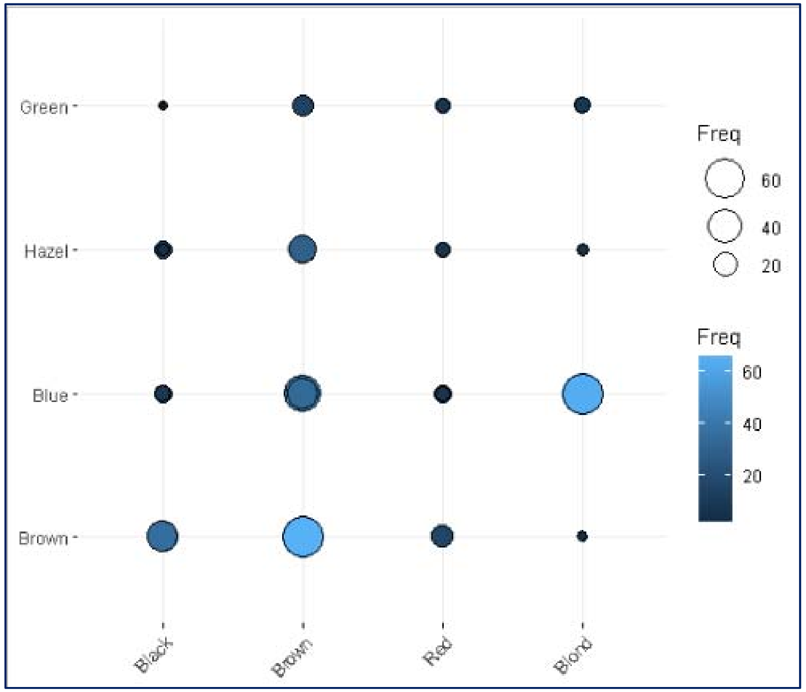
```
> prop.table(xtabs(Freq ~ Hair, data =
HairEyeColor.df.m))
Hair
    Brown     Blond     Black       Red
0.4831081 0.2145270 0.1824324 0.1199324
```

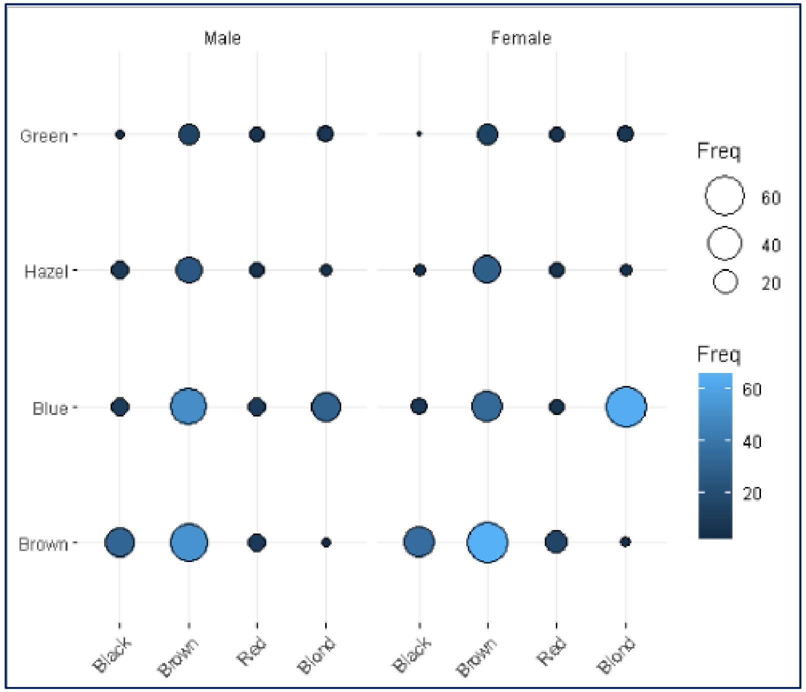


Balloon plot

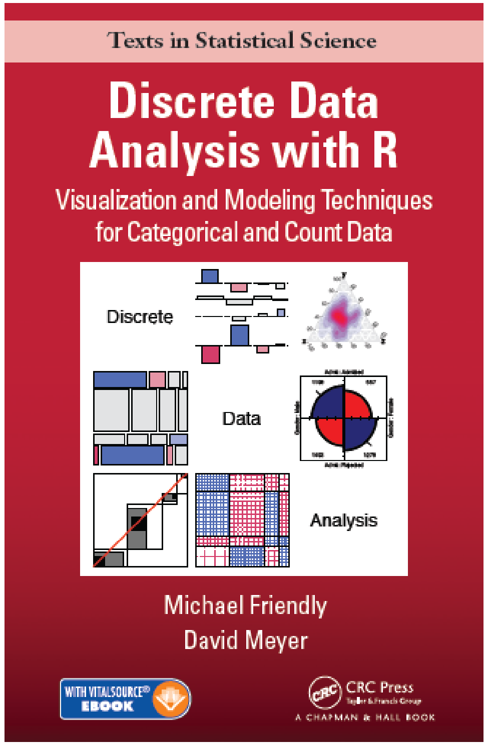
- `ggballoonplot {ggpubr}` draws a graphical matrix of a contingency table, where each cell contains a dot whose size reflects the relative magnitude of the corresponding component.

```
library(ggpubr)
ggballoonplot(HairEyeColor.df, x = "Hair", y = "Eye", size = "Freq",
              fill = "Freq")
ggballoonplot(HairEyeColor.df, x = "Hair", y = "Eye", size = "Freq",
              fill = "Freq", facet.by = "Sex")
```





Some Books

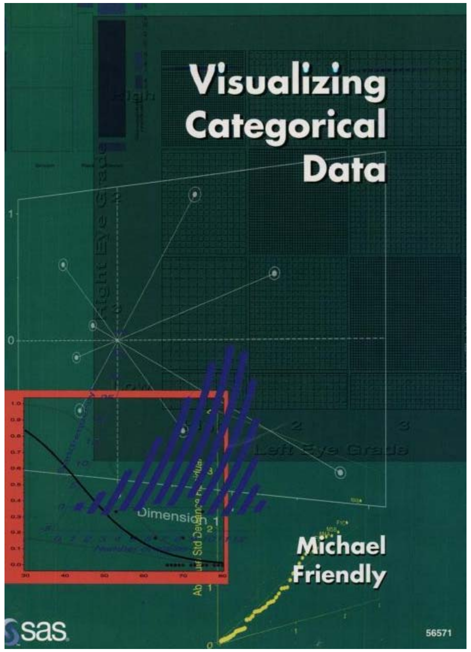

Texts in Statistical Science
Discrete Data Analysis with R
Visualization and Modeling Techniques for Categorical and Count Data
Discrete
Data
Analysis
Michael Friendly
David Meyer
WITH VITALSOURCE® EBOOK
CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

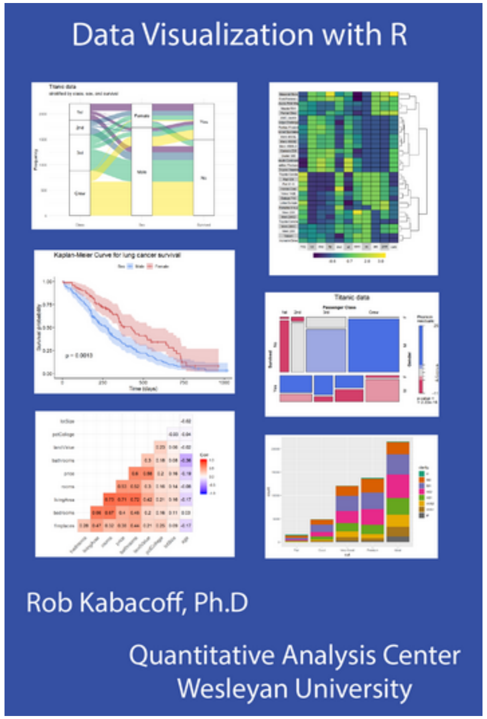
http://ddar.datavis.ca/

```
> library(vcd)
```

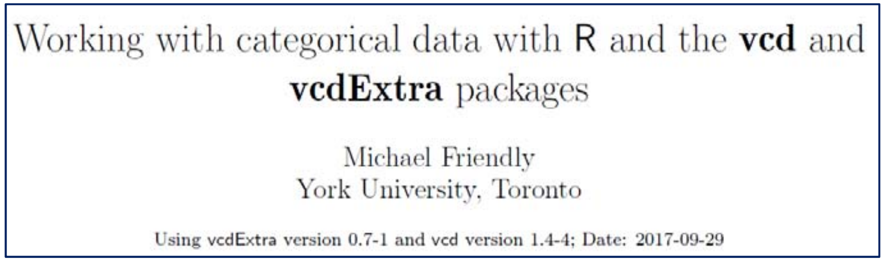
vcd: Visualizing Categorical Data

http://cran.r-project.org/web/packages/vcd/index.html





https://rkabacoff.github.io/datavis/



Working with categorical data with R and the **vcd** and **vcdExtra** packages

Michael Friendly
York University, Toronto

Using vcdExtra version 0.7-1 and vcd version 1.4-4; Date: 2017-09-29

Berkeley admission data as in Friendly (1995).

```
> UCBAdmissions
, , Dept = A
          Gender
Admit      Male Female
  Admitted  512     89
  Rejected  313     19

, , Dept = B
          Gender
Admit      Male Female
  Admitted  353     17
  Rejected  207      8

, , Dept = C
          Gender
Admit      Male Female
  Admitted  120    202
  Rejected  205    391

, , Dept = D
          Gender
Admit      Male Female
  Admitted  138    131
  Rejected  279    244

, , Dept = E
          Gender
Admit      Male Female
  Admitted   53     94
  Rejected  138    299

, , Dept = F
          Gender
Admit      Male Female
  Admitted   22     24
  Rejected  351    317
```

```
> (BerkeleyAd.array <- aperm(UCBAdmissions, c(2, 1, 3)))
, , Dept = A
        Admit
Gender   Admitted Rejected
  Male        512      313
  Female       89       19
, , Dept = B
        Admit
Gender   Admitted Rejected
  Male        353      207
  Female       17        8
, , Dept = C
        Admit
Gender   Admitted Rejected
  Male        120      205
  Female      202      391

, , Dept = D
        Admit
Gender   Admitted Rejected
  Male        138      279
  Female      131      244

, , Dept = E
        Admit
Gender   Admitted Rejected
  Male         53      138
  Female       94      299

, , Dept = F
        Admit
Gender   Admitted Rejected
  Male         22      351
  Female       24      317
```

`aperm {base}`: Array Transposition
Transpose an array by permuting its dimensions and optionally resizing it.

Data: Adminnsion to Berkeley Graduate Programs

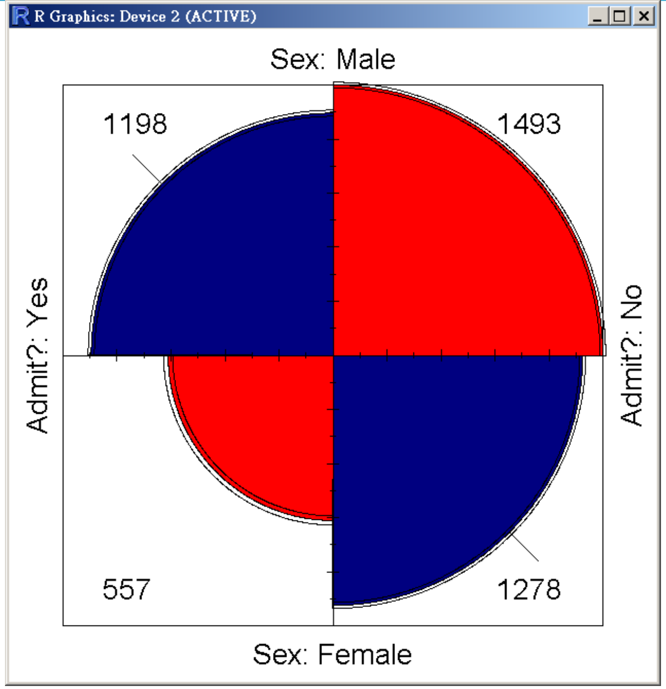
```
> dimnames(BerkeleyAd.array)[[2]] <- c("Yes", "No")
> names(dimnames(BerkeleyAd.array)) <- c("Sex", "Admit?", "Department")
> ##ftable: Flat Contingency Tables
> ftable(BerkeleyAd.array)
              Department   A   B   C   D   E   F
Sex    Admit?
Male   Yes               512 353 120 138  53  22
       No                313 207 205 279 138 351
Female Yes                89  17 202 131  94  24
       No                 19   8 391 244 299 317
```

```
> margin.table(BerkeleyAd.array, 1)
Sex
  Male Female
  2691   1835
> margin.table(BerkeleyAd.array, 2)
Admit?
 Yes   No
1755 2771
> (BerkeleyAd.mdata <- margin.table(BerkeleyAd.array, c(1, 2)))
        Admit?
Sex        Yes   No
  Male   1198 1493
  Female  557 1278
```

Fourfold Display

Table: Adminnsion to Berkeley Graduate Programs.

Gender	Admitted	Rejected	Row Total
	Adminnsion		
Males	1198	1493	2691
Females	557	1278	1835
Column Total	1755	2771	4526



- **Fourfold Display**: display for 2x2 (and 2x2xk) tables which focus on the odds ratio as a measure of association, indicating the direction and significance of associations.
- Each cell is shown by a quarter circle, whose area is proportional to the cell count, in a way that depicts the odds ratio in each of K strata.

- **Confidence rings:** for the odds ratio can be superimposed to provide a visual test of the hypothesis of no association in each stratum.
- The rings for adjacent segments are overlapped when no significant association is shown.

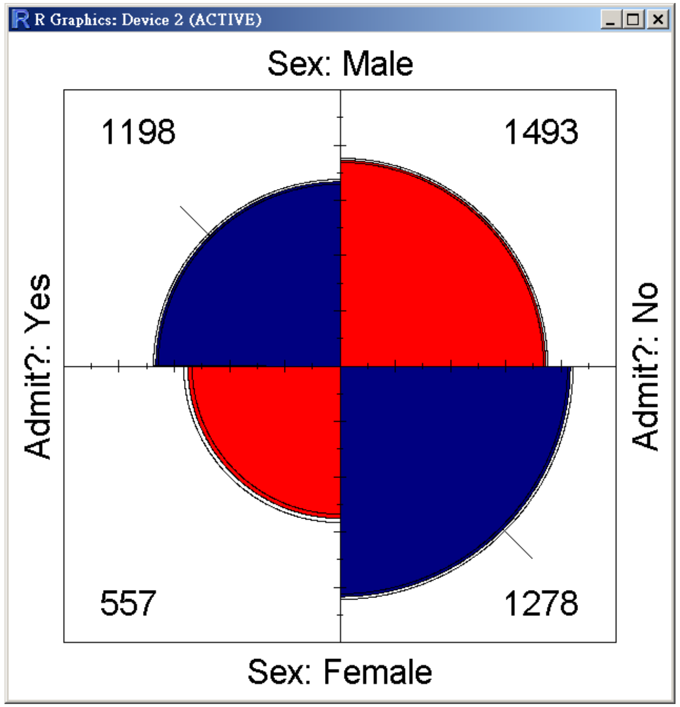
```
> fourfold(BerkeleyAd.mdata, std="all.max")
> fourfold(BerkeleyAd.mdata, margin = 1)
> fourfold(BerkeleyAd.mdata, margin = 2)
```

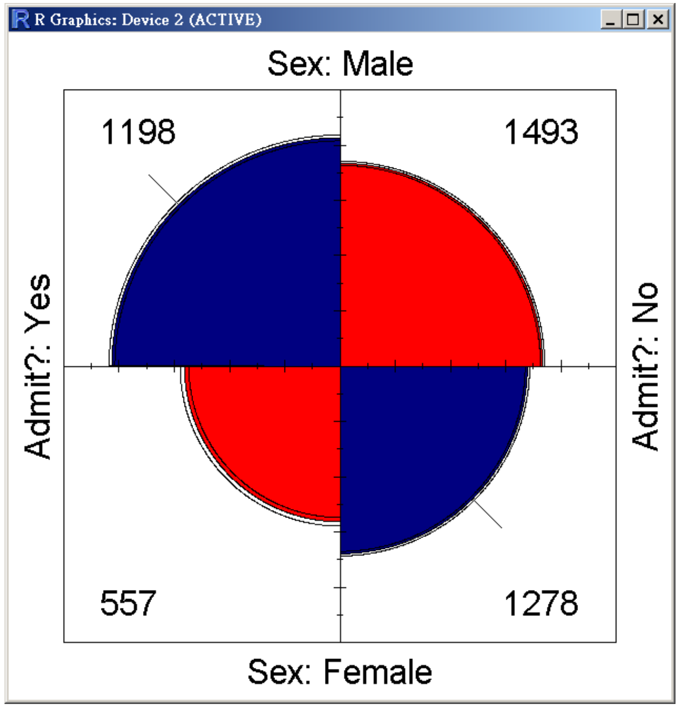
Table: UCBAdmissions: gender equated.

	Row Percents (%)		
Gender	Admitted	Rejected	Row Total
Males	44.52	55.48	100
Females	30.35	69.65	100

Table: UCBAdmissions: admission equated

	Admissions	
Column Percents (%)	Admitted	Rejected
Males	68.26	53.88
Females	31.14	46.12
Column Total	100	100

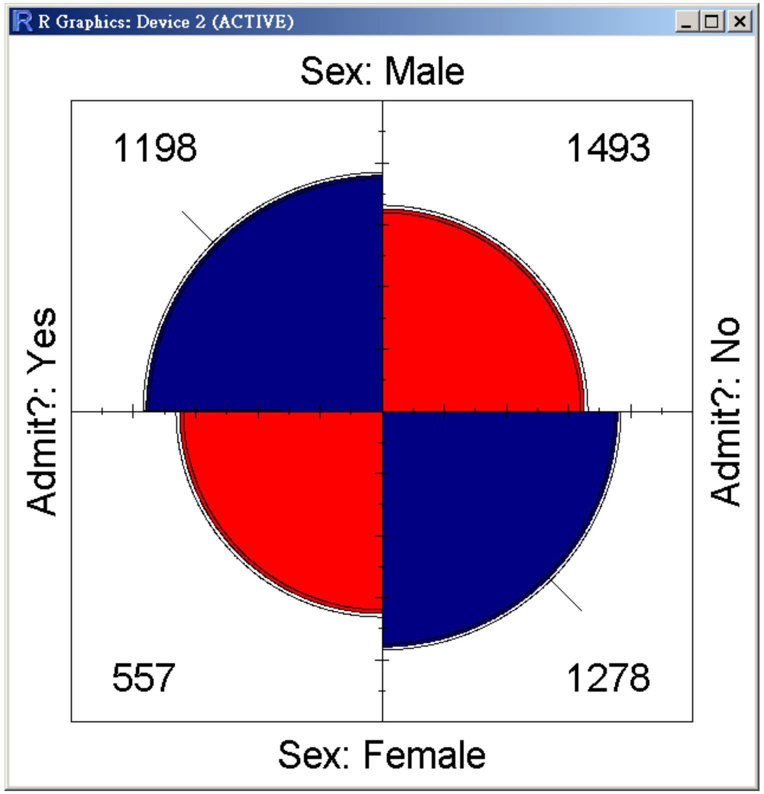




> fourfold(BerkeleyAd.mdata, margin = c(1, 2))

Table: UCBAdmissions: gender and admission equated

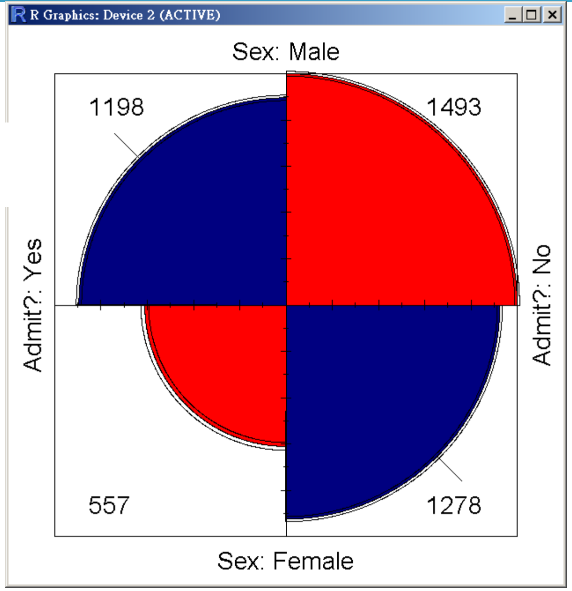
	Admissions		
Column Percents (%)	Admitted	Rejected	
Males	55.89	44.11	
Females	40.31	59.69	
Column Total			100

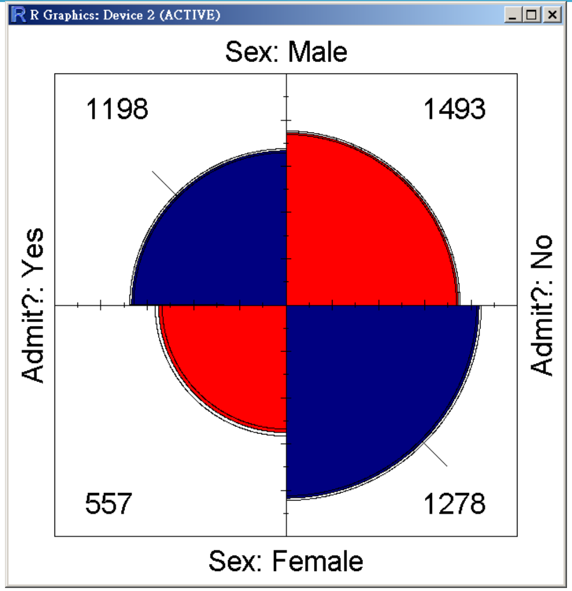


	Admissions		
Column Percents (%)	Admitted	Rejected	Row Total
Males	68.26	53.88	122.14
Females	31.14	46.12	77.26
Column Total	100	100	

Comparison

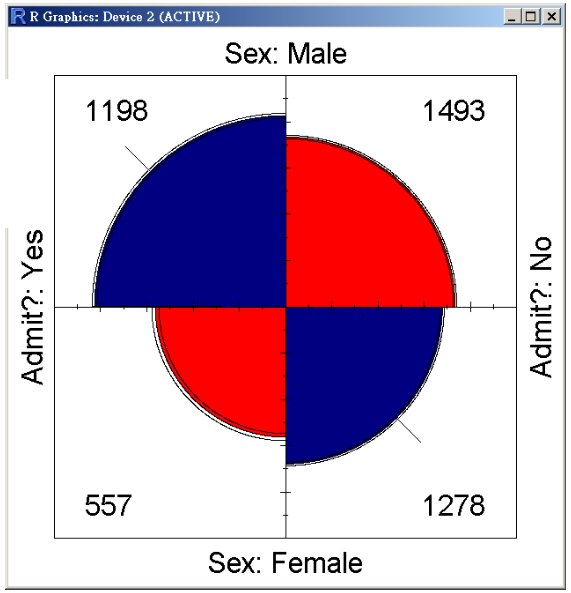
std="all.max"

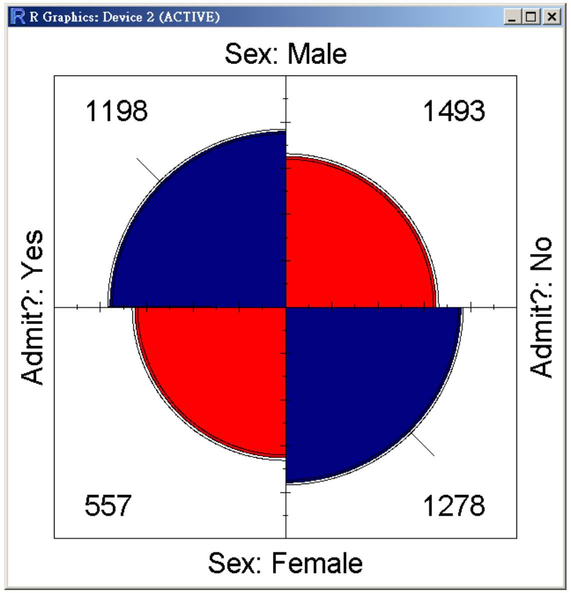




gender equated

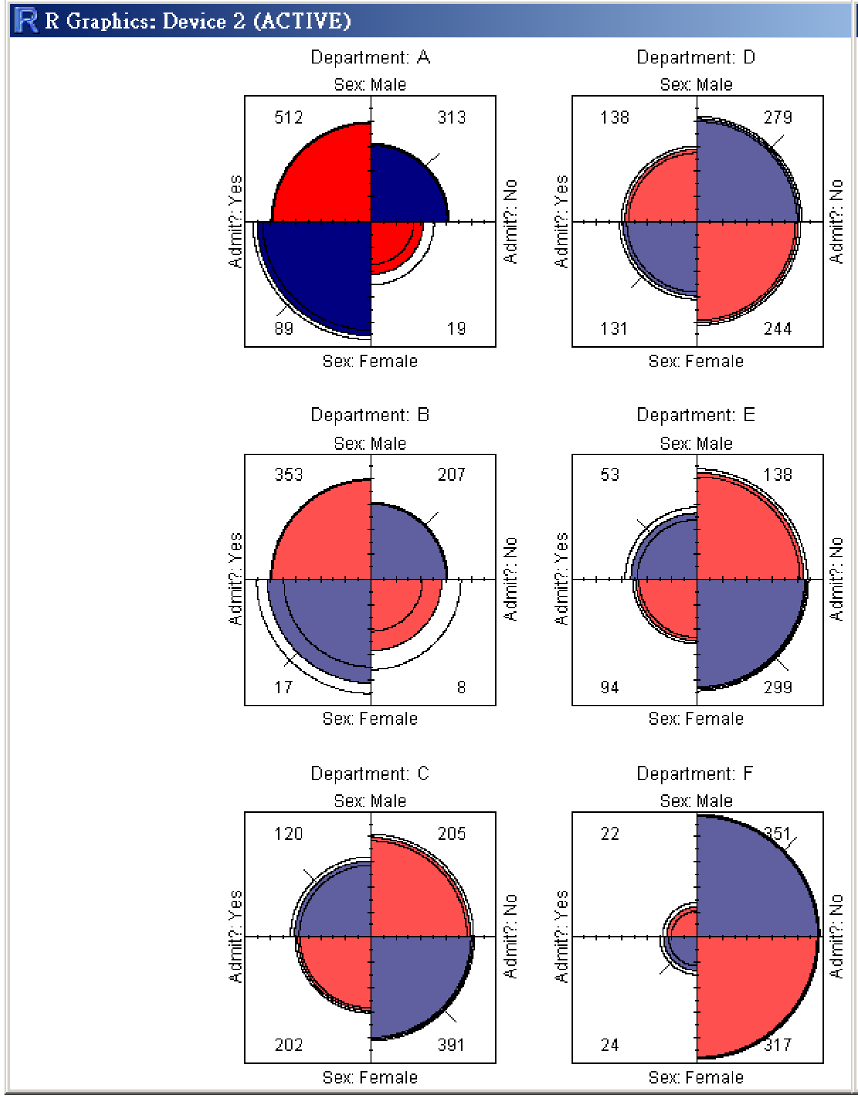
admission equated

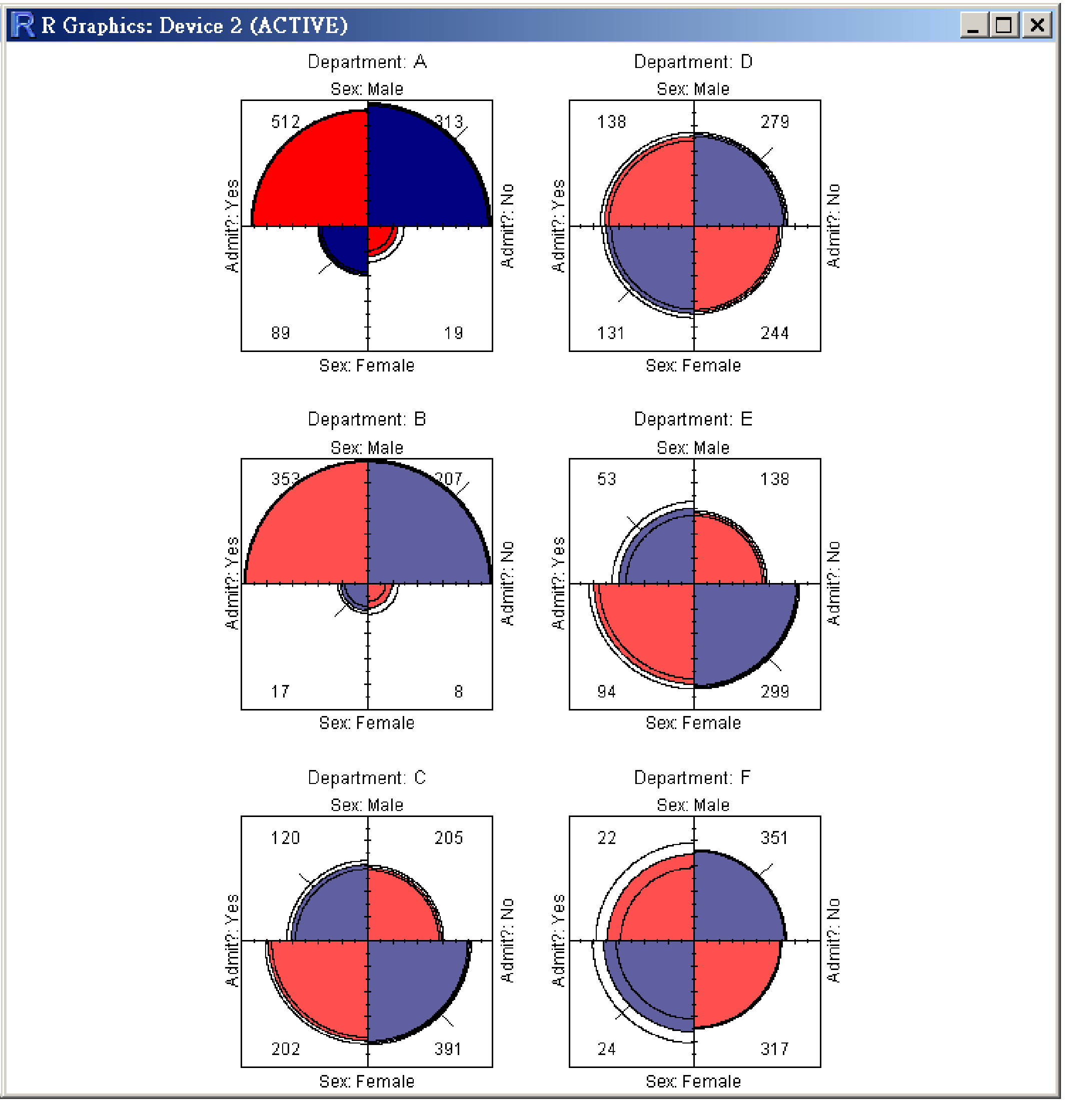




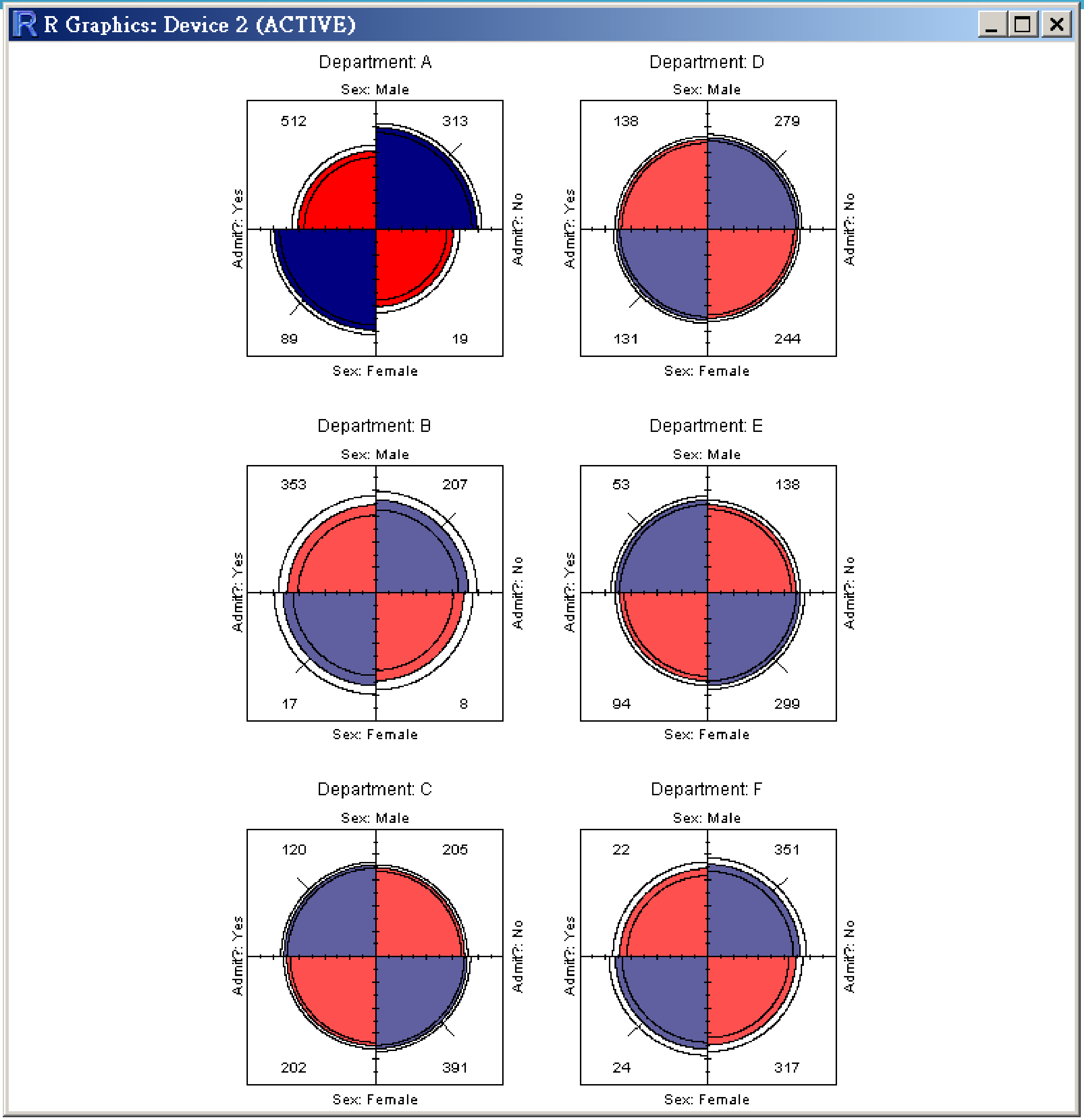
gender and admission equated

```
> fourfold(BerkeleyAd.array, margin = 1)
> fourfold(BerkeleyAd.array, margin = 2)
```

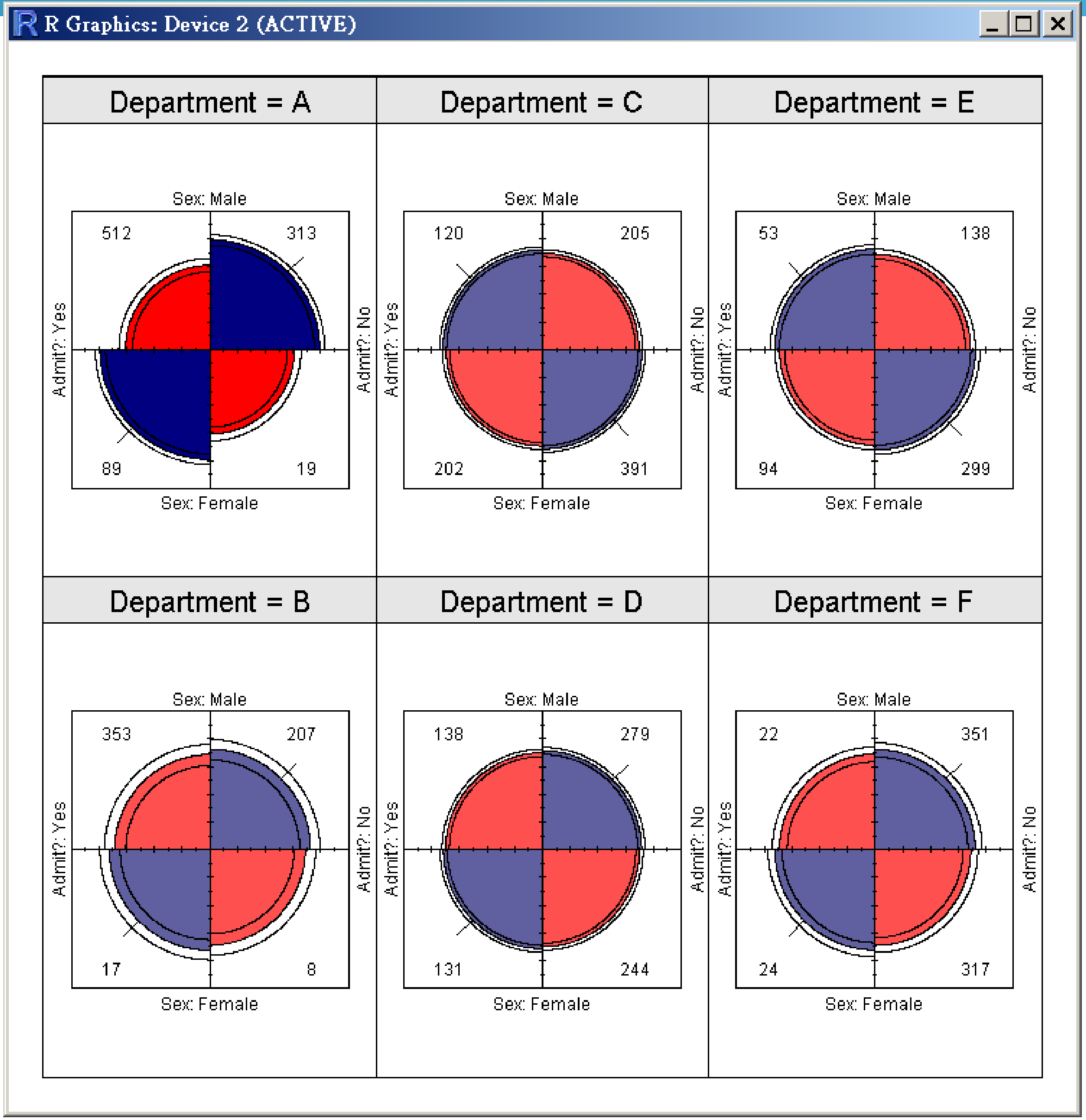




> fourfold(BerkeleyAd.array)



cotabplot(BerkeleyAd.array, panel = cotab_fourfold)



Make a Contingency Table

```
> score <- as.factor(sample(c("High","Low"), 20, replace=TRUE))
> gender <- as.factor(sample(c("F","M"), 20, replace=TRUE))
> my.data <- data.frame(gender=gender, score=score)
> my.data
   gender score
1        M  High
2        F  High
3        F   Low
4        M  High
5        F   Low
...
19       F   Low
20       F   Low
> table(my.data)
      score
gender High Low
     F    1   9
     M    8   2
```

```
> my.table <- table(my.data)
> str(my.table)
 'table' int [1:2, 1:2] 1 8 9 2
 - attr(*, "dimnames")=List of 2
  ..$ gender: chr [1:2] "F" "M"
  ..$ score : chr [1:2] "High" "Low"
> class(my.table)
[1] "table"
```

Data: Hair and Eye Color and Gender in 592 statistics students.

```
> HairEyeColor
, , Sex = Male
       Eye
Hair    Brown Blue Hazel Green
  Black    32   11    10     3
  Brown    53   50    25    15
  Red      10   10     7     7
  Blond     3   30     5     8
, , Sex = Female
       Eye
Hair    Brown Blue Hazel Green
  Black    36    9     5     2
  Brown    66   34    29    14
  Red      16    7     7     7
  Blond     4   64     5     8
```

```
> str(HairEyeColor)
 table [1:4, 1:4, 1:2] 32 53 10 3 11 50 10 30 10 25 ...
 - attr(*, "dimnames")=List of 3
  ..$ Hair: chr [1:4] "Black" "Brown" "Red" "Blond"
  ..$ Eye : chr [1:4] "Brown" "Blue" "Hazel" "Green"
  ..$ Sex : chr [1:2] "Male" "Female"
> class(HairEyeColor)
[1] "table"
```

Make a Contingency Table

```
> (HEC <- structable(Eye ~ Sex + Hair,
                     data = HairEyeColor))
             Eye Brown Blue Hazel Green
Sex    Hair                            
Male   Black        32   11    10     3
       Brown        53   50    25    15
       Red          10   10     7     7
       Blond         3   30     5     8
Female Black        36    9     5     2
       Brown        66   34    29    14
       Red          16    7     7     7
       Blond         4   64     5     8
```

```
> (HEC1 <- structable(Hair ~ Eye + Sex,
                      data = HairEyeColor))
             Hair Black Brown Red Blond
Eye   Sex                              
Brown Male           32    53  10     3
      Female         36    66  16     4
Blue  Male           11    50  10    30
      Female          9    34   7    64
Hazel Male           10    25   7     5
      Female          5    29   7     5
Green Male            3    15   7     8
      Female          2    14   7     8
```

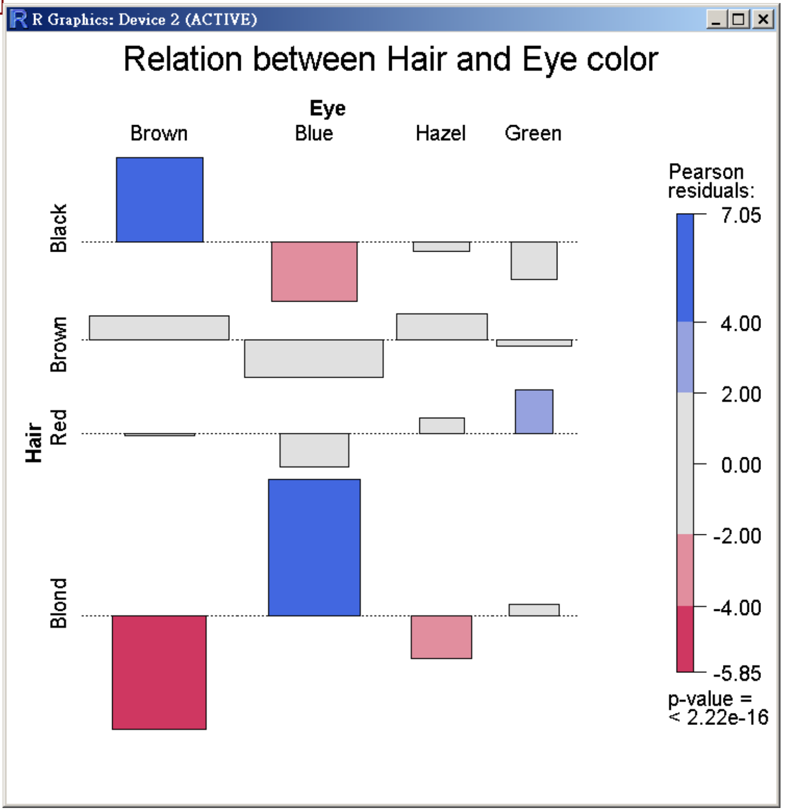
```
> (HEC2 <- structable(~Eye + Sex +
Hair, data = HairEyeColor))
            Sex Male Female
Eye   Hair                 
Brown Black       32     36
      Brown       53     66
      Red         10     16
      Blond        3      4
Blue  Black       11      9
      Brown       50     34
      Red         10      7
      Blond       30     64
Hazel Black       10      5
      Brown       25     29
      Red          7      7
      Blond        5      5
Green Black        3      2
      Brown       15     14
      Red          7      7
      Blond        8      8
```

Association Plots

```
> (x <- margin.table(HairEyeColor, c(1, 2)))
       Eye
Hair    Brown Blue Hazel Green
  Black    68   20    15     5
  Brown   119   84    54    29
  Red      26   17    14    14
  Blond     7   94    10    16
> assoc(x, main = "...", shade = TRUE)
```

assoc {vcd}: Extended Association Plots

Produce an association plot indicating deviations from a specified independence model in a possibly high-dimensional contingency table.



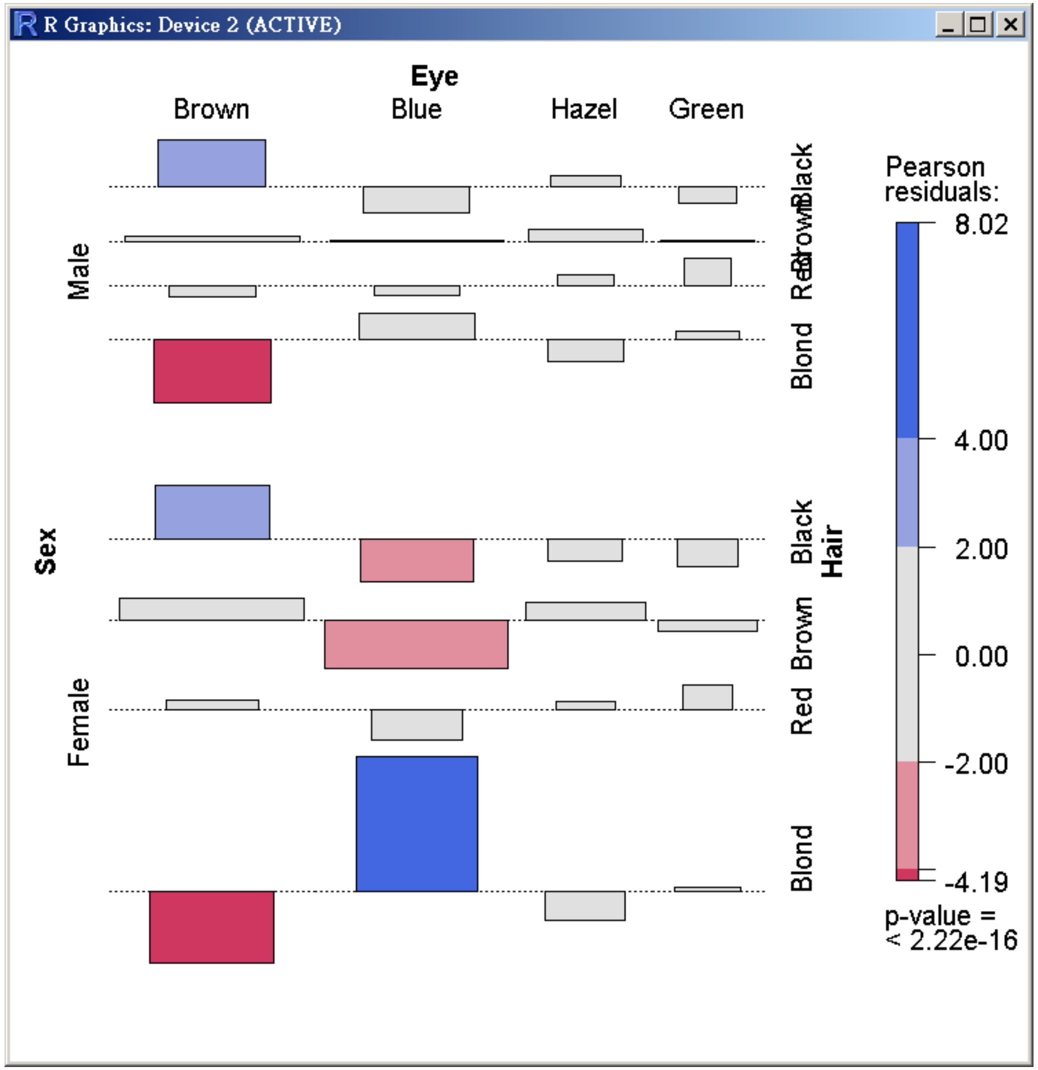
Association plots have been suggested by Cohen (1980) and extended by Friendly (1992) and provide a means for visualizing the residuals of an independence model for a contingency table.

For a contingency table, the signed contribution to Pearson's chi^2 for cell \{ij… k\} is

d_{ij…k} = (f_{ij…k} - e_{ij…k}) / sqrt(e_{ij…k})

Association Plots

```
> assoc(HEC, shade = TRUE)
```

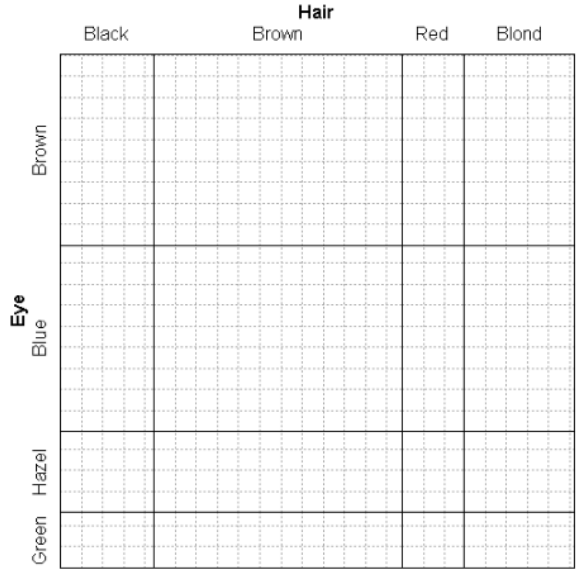


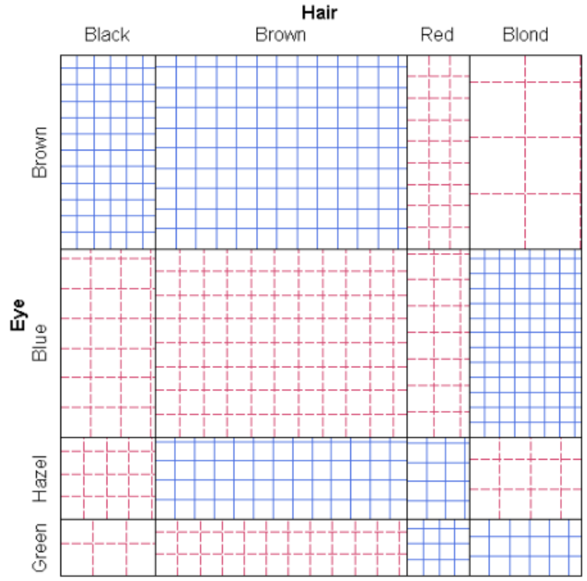
Sieve Plots

sieve {vcd}: Extended Sieve Plots

(Extended) sieve displays for n-way contingency tables: plots rectangles with areas proportional to the expected cell frequencies and filled with a number of squares equal to the observed frequencies. Thus, the densities visualize the deviations of the observed from the expected values.

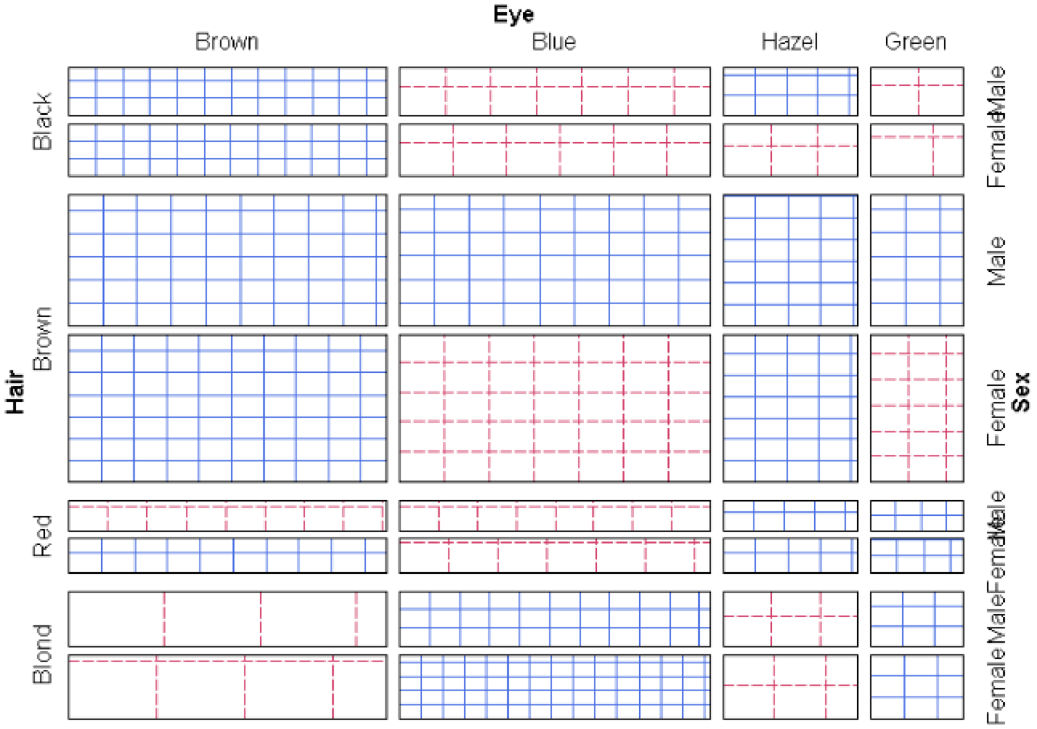
```
> # aggregate over 'sex':
> (haireye <- margin.table(HairEyeColor, c(2,1)))
       Hair
Eye     Black Brown Red Blond
  Brown    68   119  26     7
  Blue     20    84  17    94
  Hazel    15    54  14    10
  Green     5    29  14    16
> sieve(haireye, sievetype = "expected", shade = TRUE) # plot expected values:
> sieve(haireye, shade = TRUE) # plot observed table
```

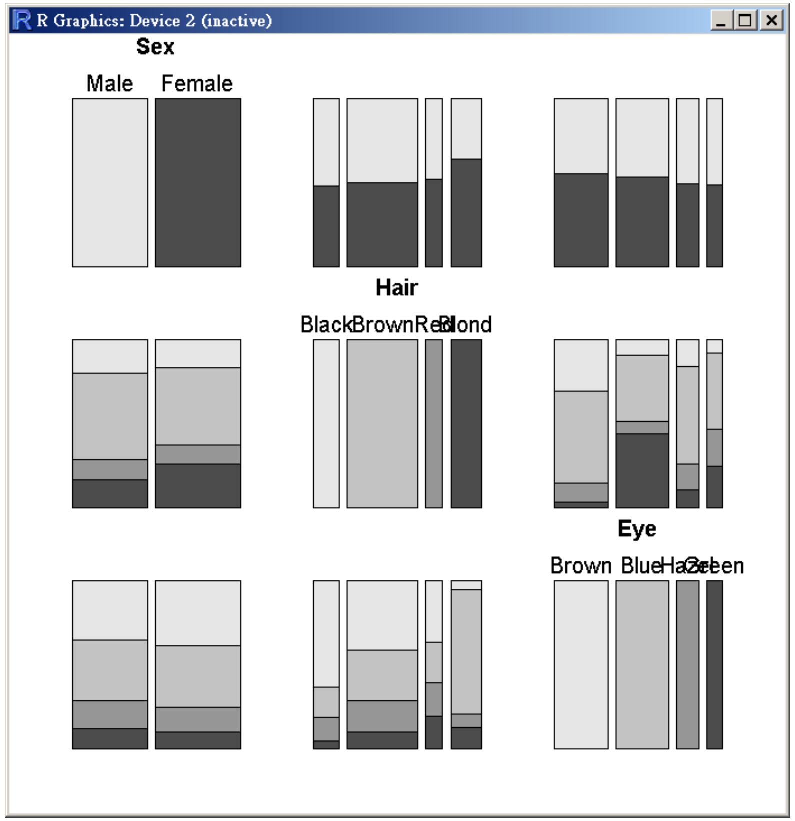




Scatterplot Matrices

```
> # plot complete diagram:
> sieve(HairEyeColor, shade = TRUE)
```





```
> pairs(HEC, highlighting = 1, diag_panel = pairs_diagonal_mosaic,
diag_panel_args = list(fill = grey.colors))
```

Mosiac Displays for Two-way Tables

- Proposed by Hartigan & Kleiner (1981) and extended in Friendly (1994a), represents the counts in a contingency table directly by tiles.
- Tiles size is proportional to the cell frequency.

		Hair Color				
		BLACK	BROWN	RED	BLOND	Total
Eye Color	Brown	68	119	26	7	220
	Blue	20	84	17	94	215
	Hazel	15	54	14	10	93
	Green	5	29	14	16	64
Total	O	108	286	71	127	n 592

$p = \frac{O}{n}$

```
              Marginal proportions
Black       Brown        Red        Blond
0.1824      0.4831      0.1199      0.2145
```

$E = \frac{n}{4}$

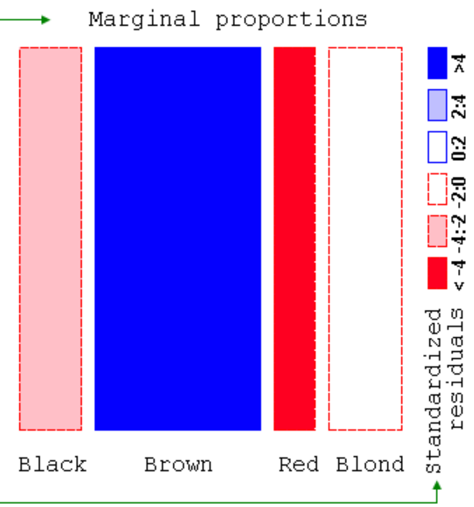
```
              Fitted frequencies
Black       Brown        Red        Blond
148.00      148.00      148.00      148.00
```

$d = \frac{O - E}{\sqrt{E}}$

```
      Standardized Pearson residuals
 Black      Brown        Red      Blond
 -3.29      11.34      -6.33      -1.73
```

Question:
how to understand the nature of the association between hair and eye color.

The Pearson X2 for these data is 138.3 with 9 degrees of freedom, indicating substantial departure from independence.

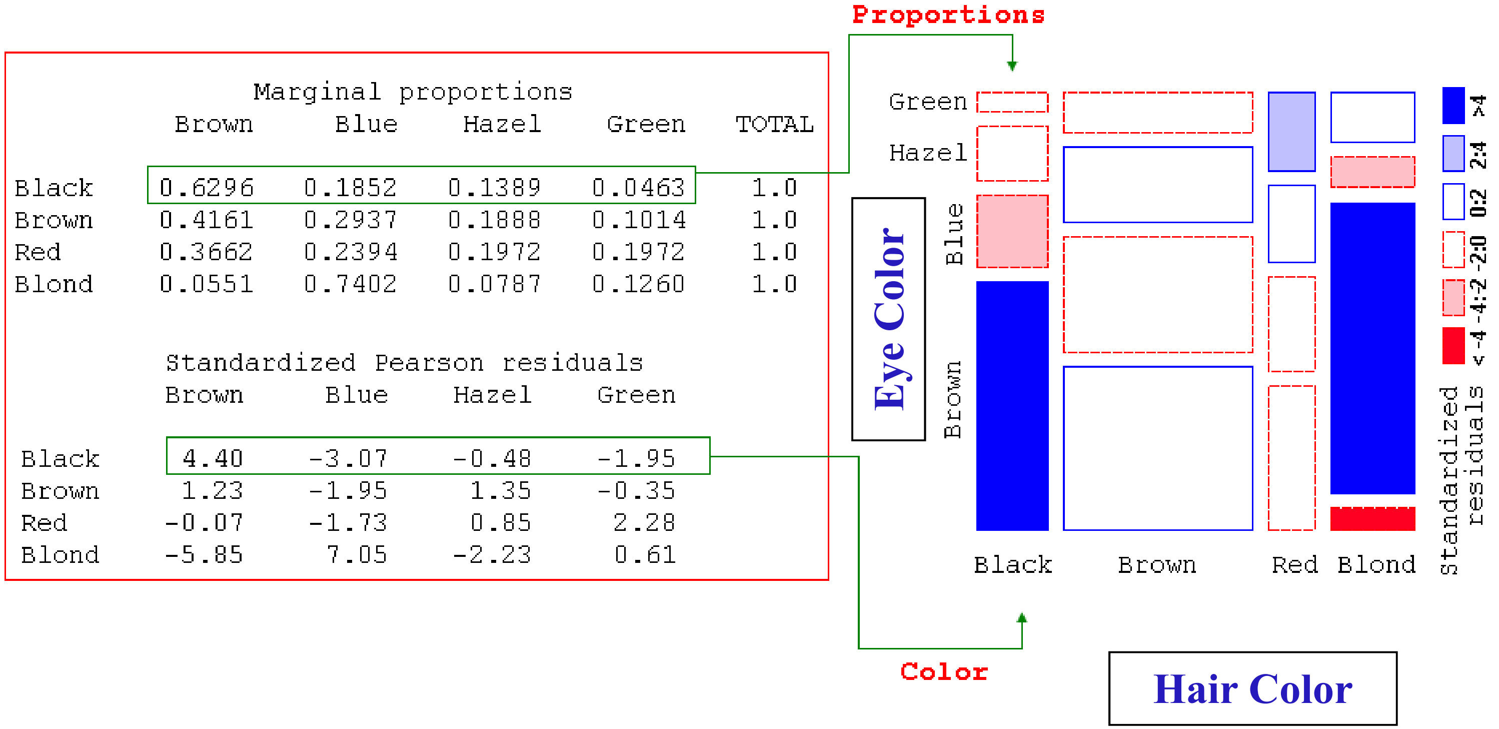


Hair Color

Reference: http://www.math.yorku.ca/SCS/Online/mosaics/about.html

Mosiac Displays: interpretation

- The association between Hair Color and Eye Color:
  - **Positive values** (Blue): cells whose observed frequency is substantially greater than would be found under independence;
  - **Negative values** (Red): indicate cells which occur less often than under independence.



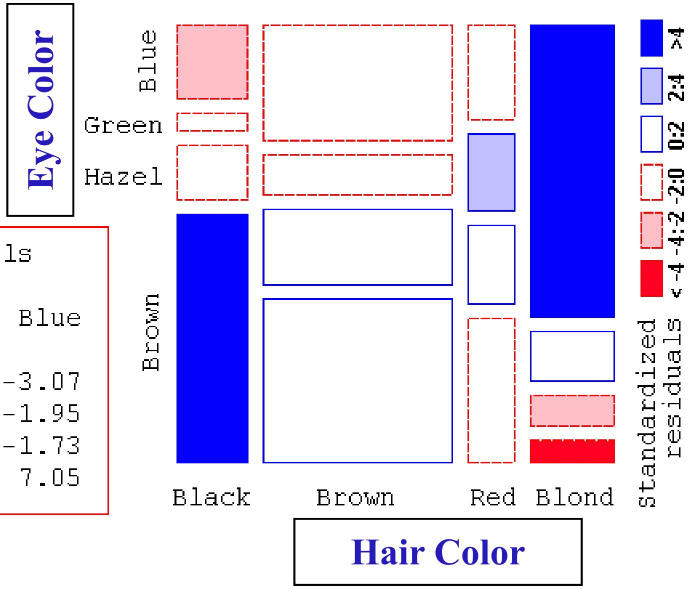
```
                 Marginal proportions
          Brown      Blue      Hazel     Green     TOTAL

Black     0.6296    0.1852    0.1389    0.0463      1.0
Brown     0.4161    0.2937    0.1888    0.1014      1.0
Red       0.3662    0.2394    0.1972    0.1972      1.0
Blond     0.0551    0.7402    0.0787    0.1260      1.0


          Standardized Pearson residuals
          Brown      Blue     Hazel      Green

Black      4.40     -3.07     -0.48     -1.95
Brown      1.23     -1.95      1.35     -0.35
Red       -0.07     -1.73      0.85      2.28
Blond     -5.85      7.05     -2.23      0.61
```

Mosiac Displays: reordering

- Reordering the rows or columns of the two-way table so that the residuals have an opposite corner pattern of signs.
- The association between Hair and Eye color is that
  - people with dark hair tend to have dark eyes,
  - those with light hair tend to have light eyes,
  - people with red hair do not quite fit this pattern

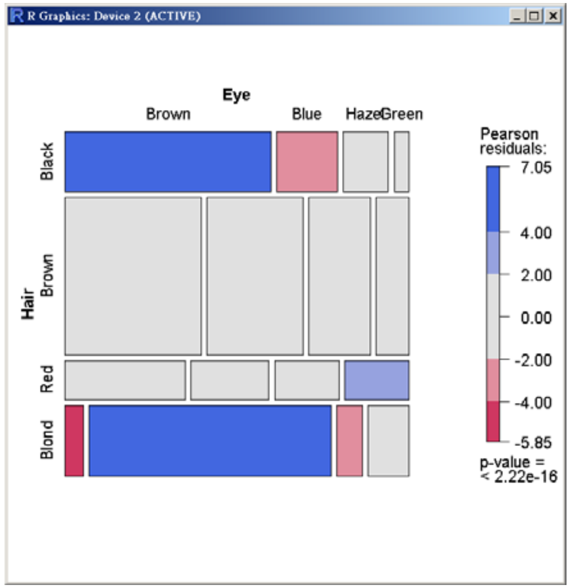
re-ordered residuals

```
              Standardized Pearson residuals

              Brown     Hazel     Green      Blue

Black          4.40     -0.48     -1.95     -3.07
Brown          1.23      1.35     -0.35     -1.95
Red           -0.07      0.85      2.28     -1.73
Blond         -5.85     -2.23      0.61      7.05
```

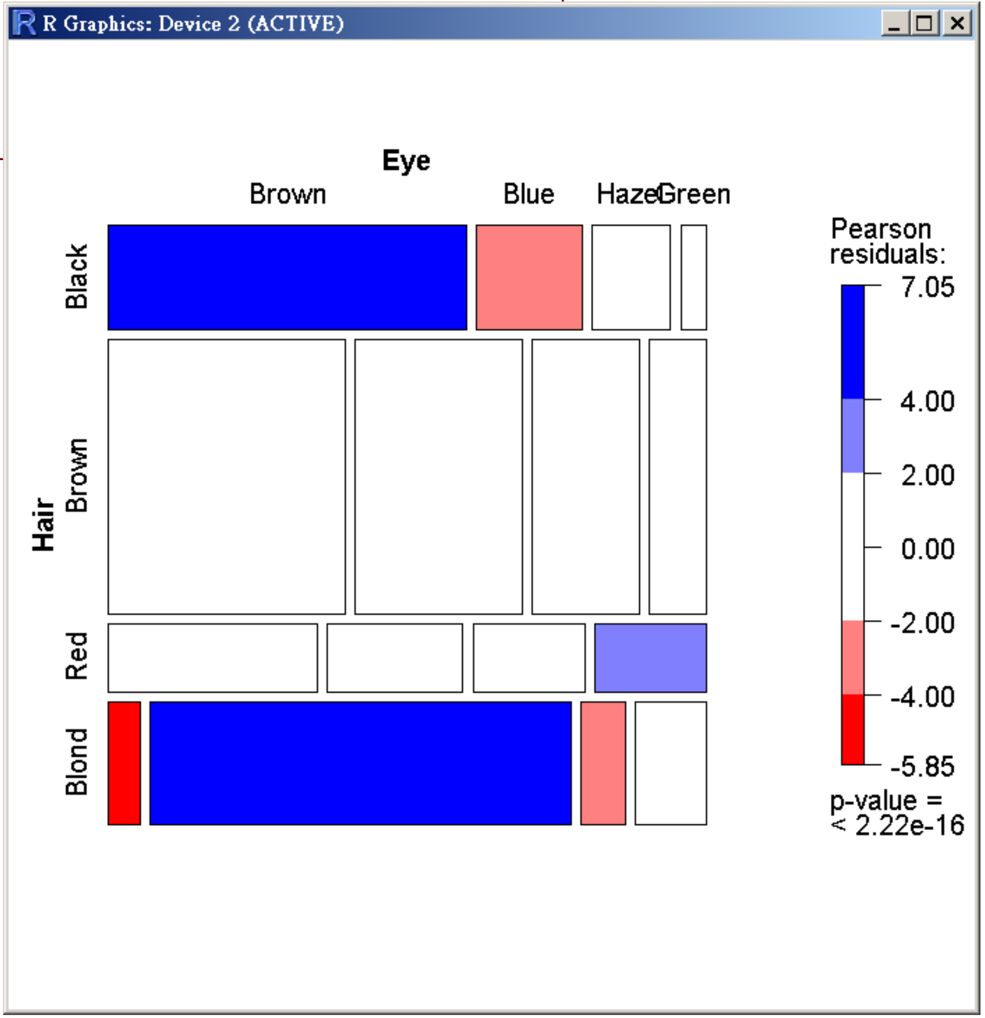


> mosaic(haireye, gp = shading_hsv)

```
> (haireye <- margin.table(HairEyeColor, c(1, 2)))
                  Eye
Hair     Brown Blue Hazel Green
  Black     68   20    15     5
  Brown    119   84    54    29
  Red       26   17    14    14
  Blond      7   94    10    16
```

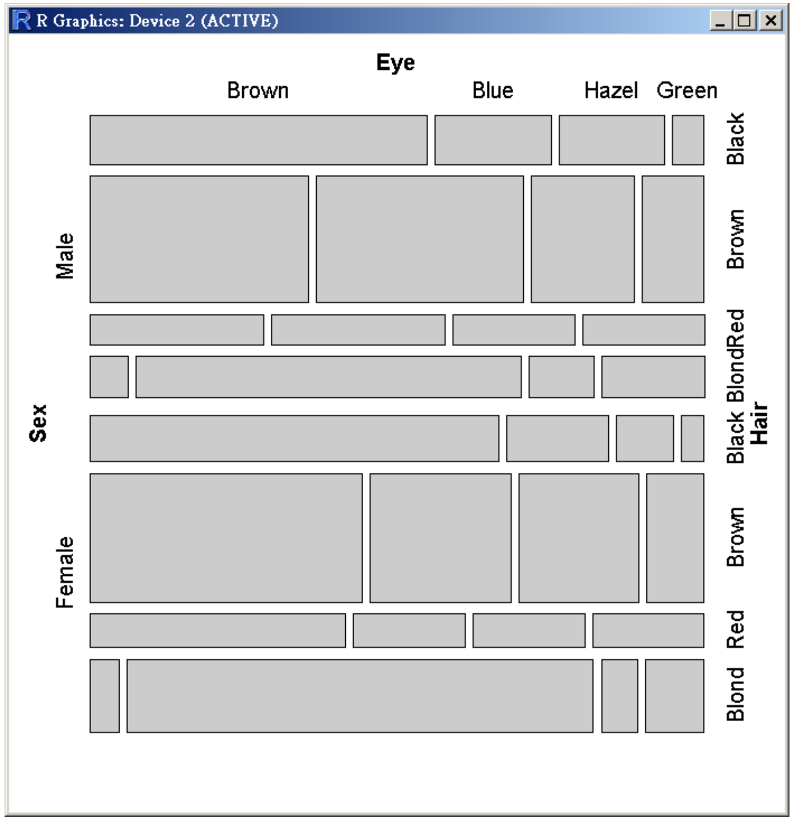


```
> mosaic(haireye, gp = shading_hcl)
```

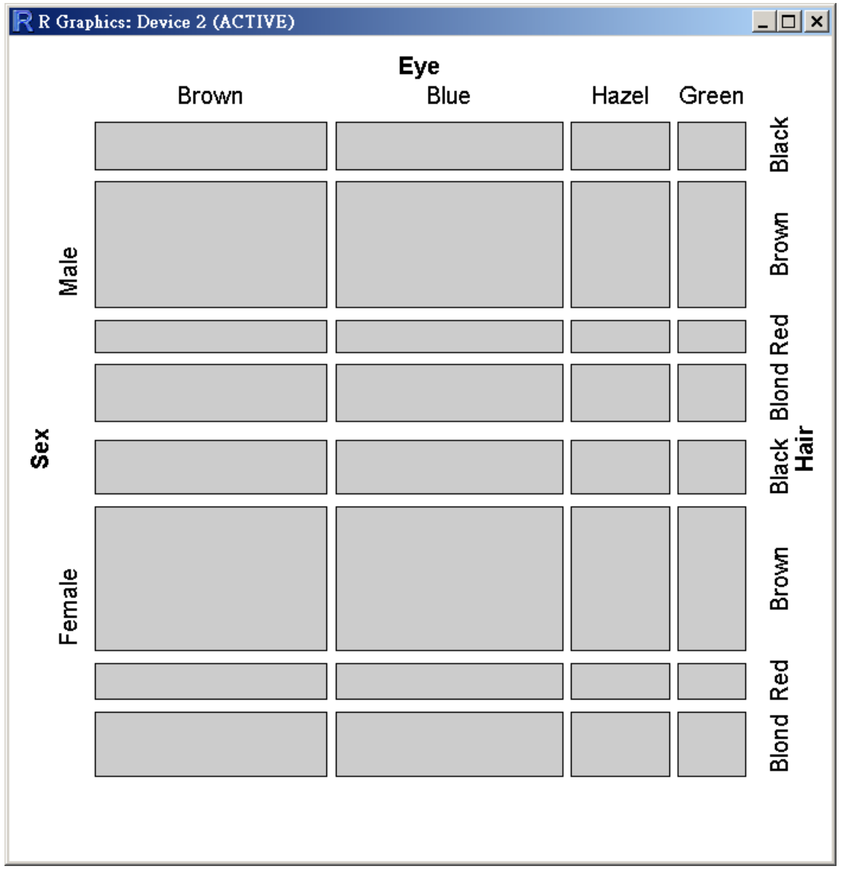


> mosaic(HEC)

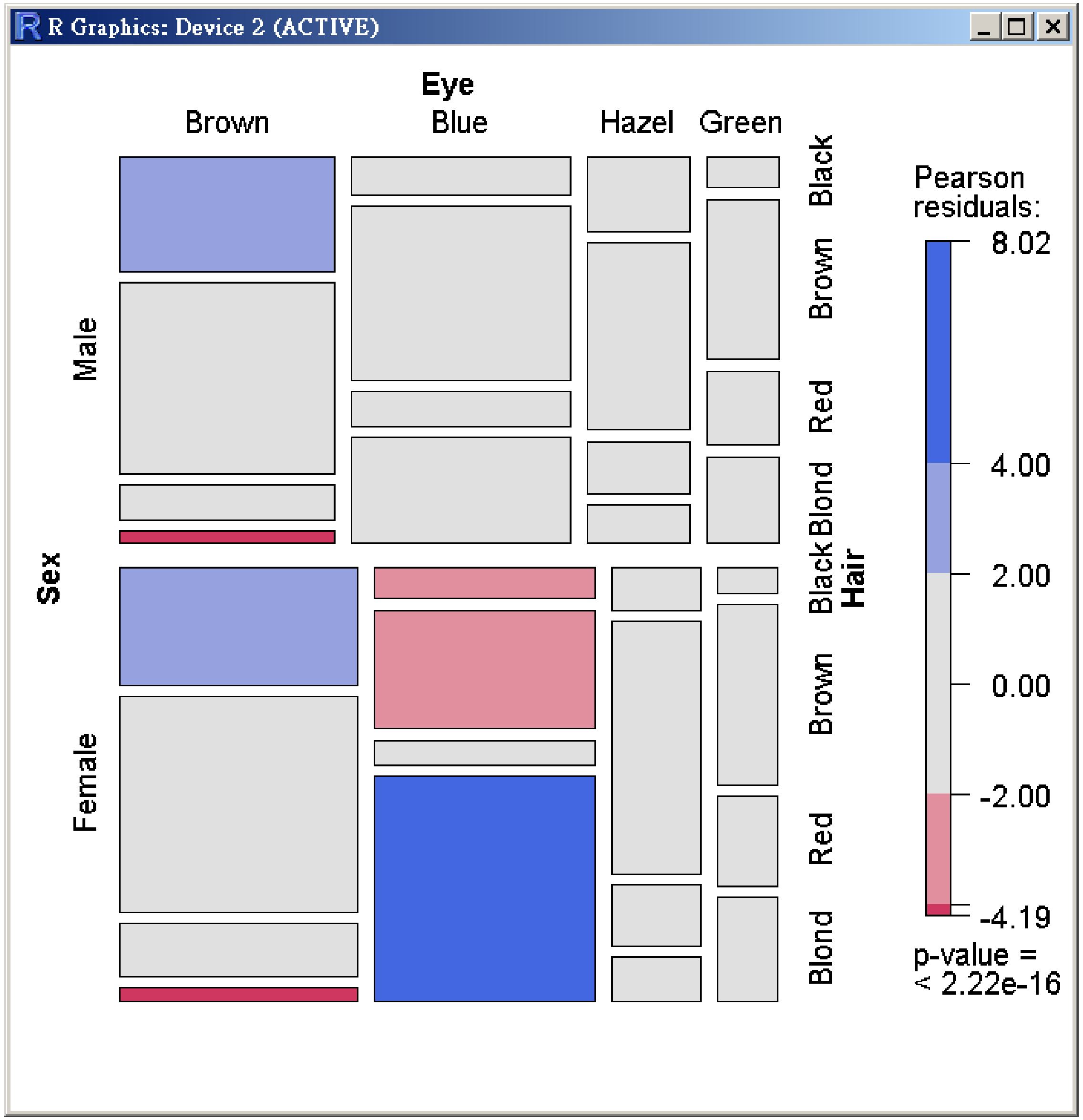
```
> (HEC <- structable(Eye ~ Sex + Hair, data = HairEyeColor))
```



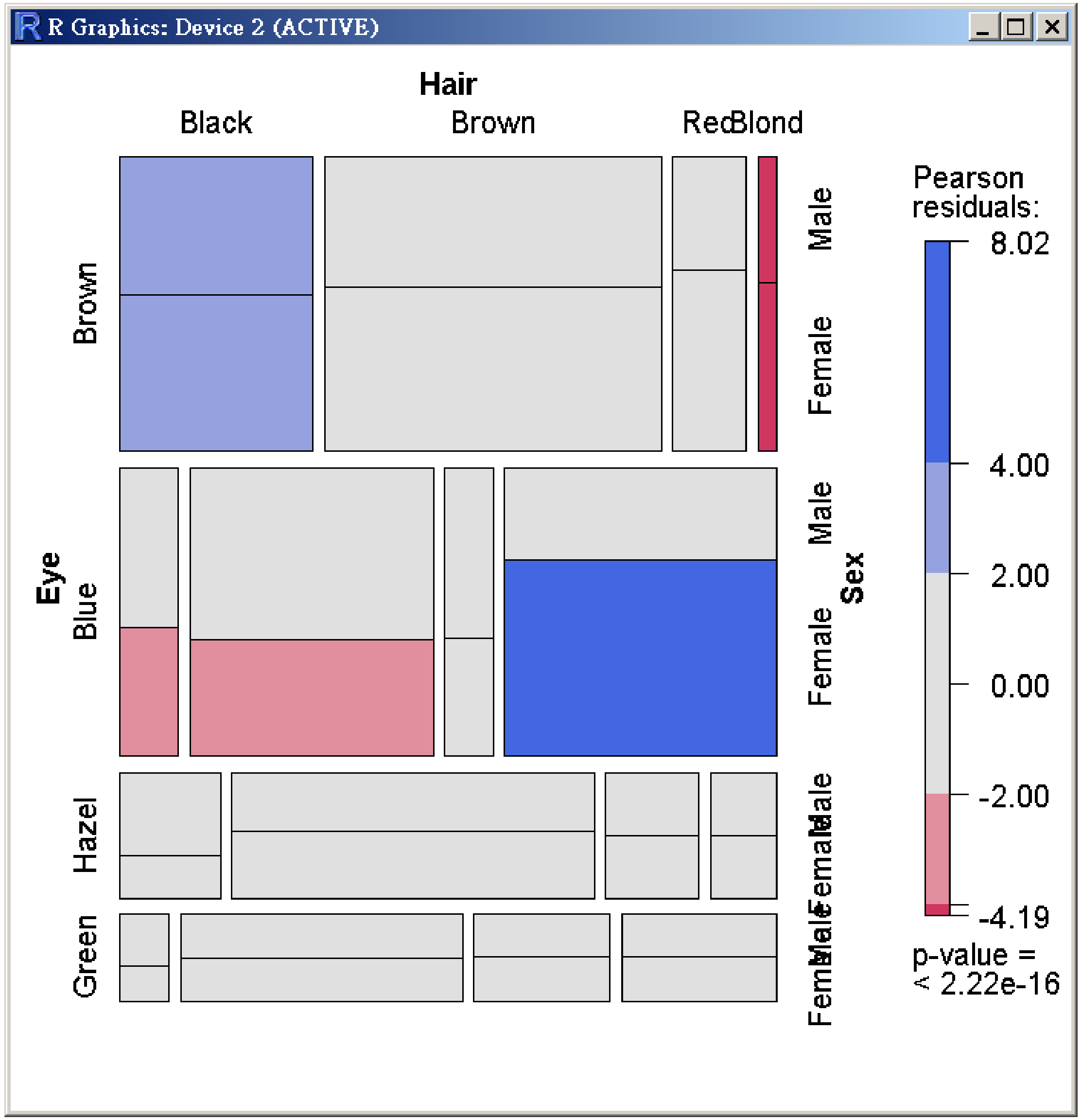
```
> mosaic(HEC, type="expected")
```



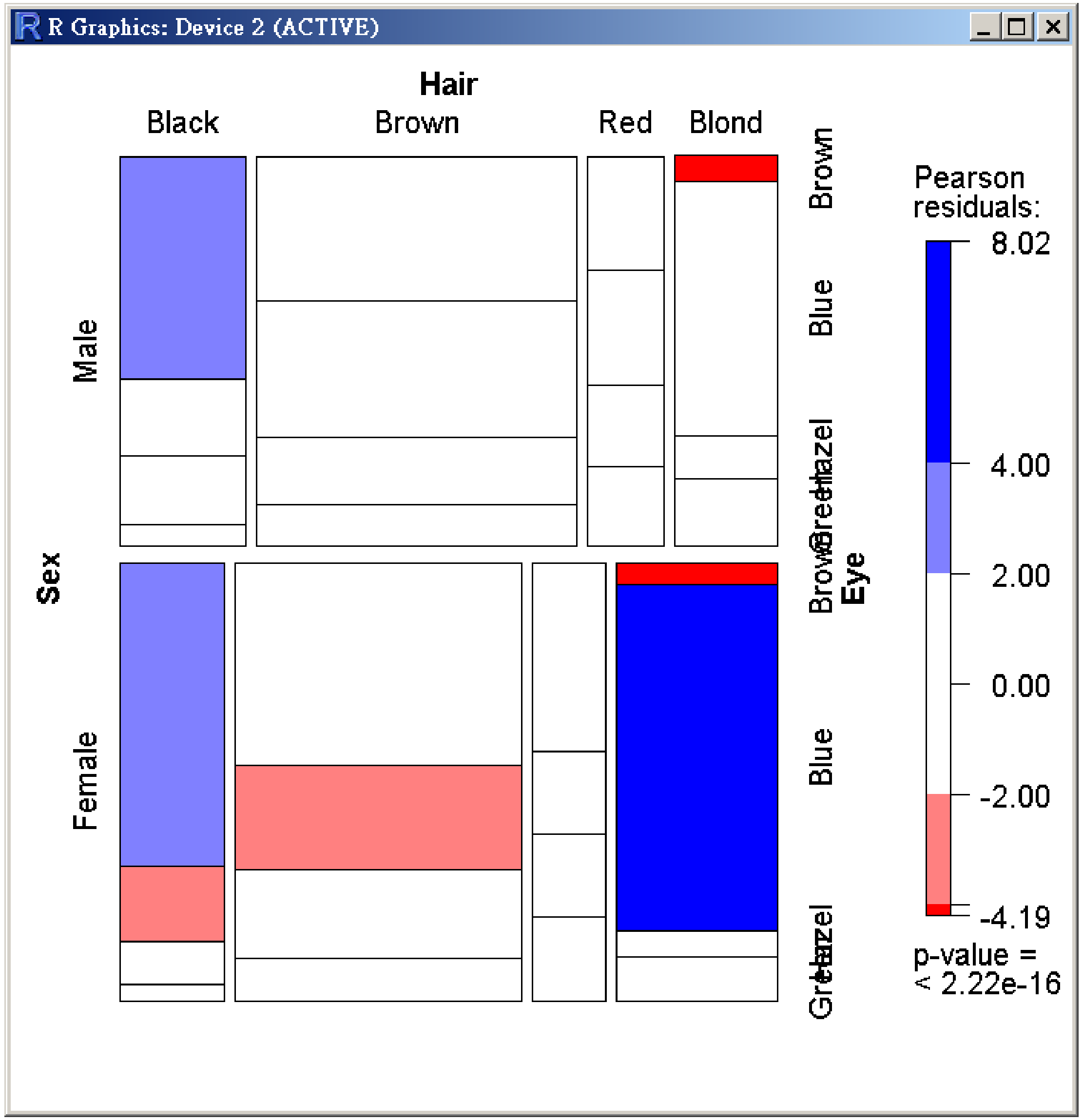
```
> mosaic(~Sex + Eye + Hair, data=HairEyeColor,
shade=TRUE)
```



```
> mosaic(Sex ~ Eye + Hair, data=HairEyeColor,
gp=shading_hcl)
```

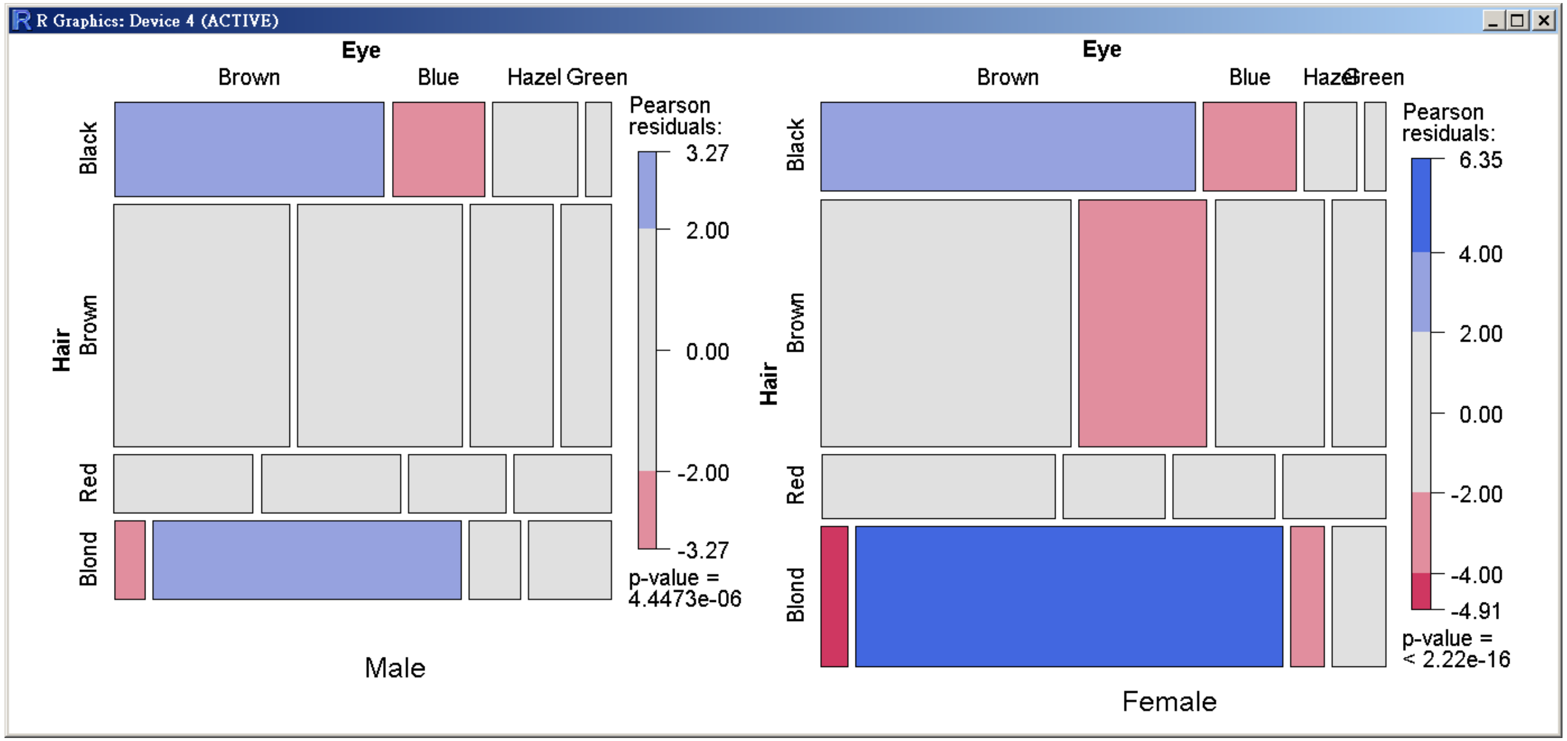


```
> mosaic(Eye ~ Sex + Hair, data=HairEyeColor,
gp=shading_hsv)
```



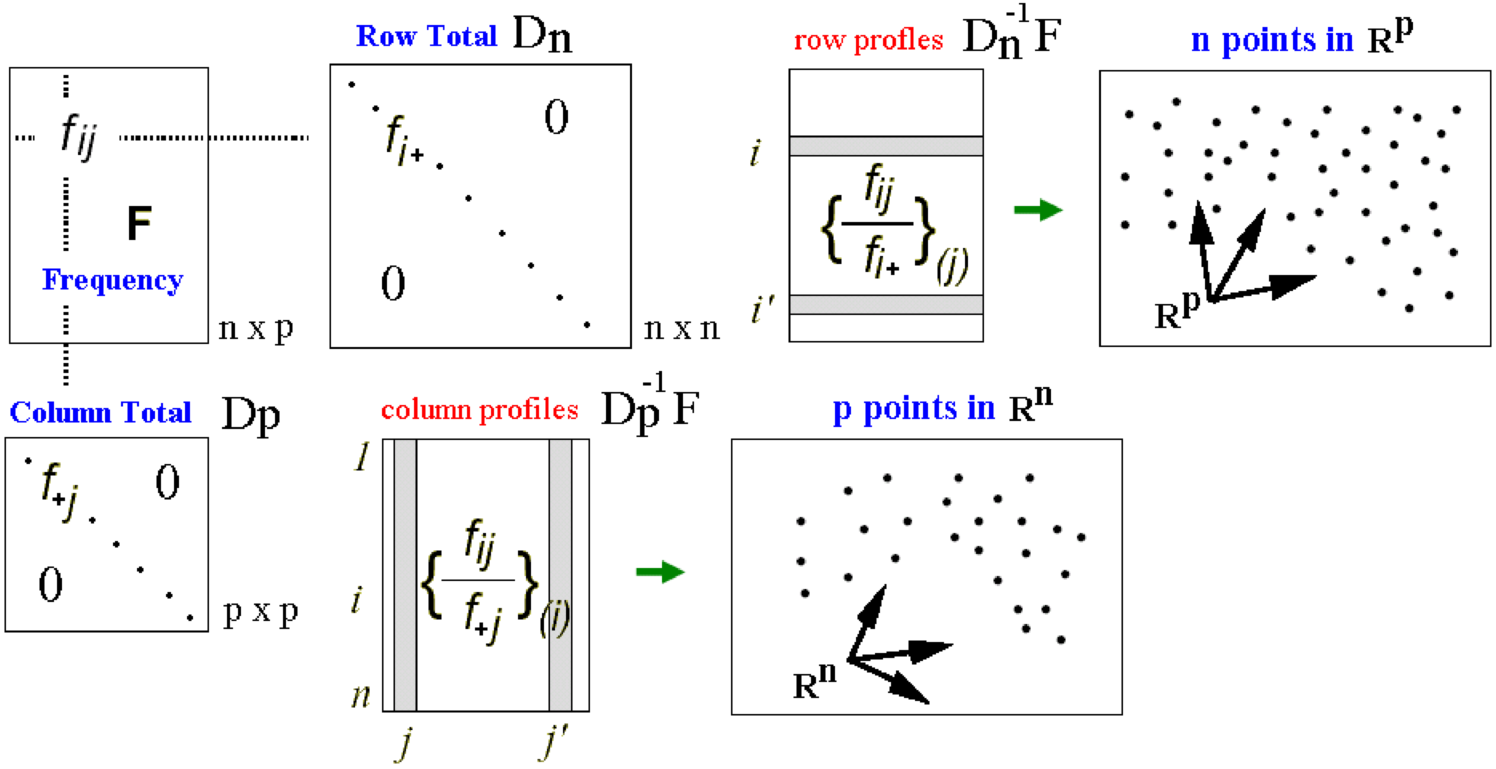
Viewport

```
> pushViewport(viewport(layout = grid.layout(ncol = 2)))
> pushViewport(viewport(layout.pos.col = 1))
> mosaic(HEC[["Male"]], margins = c(left = 2.5, top = 2.5, 0), sub="Male",
newpage = FALSE, gp = shading_hcl)
> popViewport()
> pushViewport(viewport(layout.pos.col = 2))
> mosaic(HEC[["Female"]], margins = c(top = 2.5, 0), sub="Female", newpage =
FALSE, gp = shading_hcl)
> popViewport(2)
```



Simple Correspondance Analysis (CA)

- Correspondence Analysis = PCA for categorical variables.
- Correspondence analysis is designed to analyze simple two-way and multi-way tables containing some measure of correspondence between the rows and columns.
- CA finds scores for the row and column categories on a small number of dimensions which account for the greatest proportion of the chi² for association between the row and column categories, just as principal components account for maximum variance.



Correspondance Analysis (conti.)

	n row points in $\mathbf{R}^{p}$ space	p column points in $\mathbf{R}^{n}$ space
Analysis of Table X	$\mathbf{X} = \mathbf{D}_n^{-1}\mathbf{F}$ p coordinates $\frac{f_{ij}}{f_{i.}}$, for $j=1, 2, \ldots, p.$	$\mathbf{X} = \mathbf{D}_p^{-1}\mathbf{F}'$ n coordinates $\frac{f_{ij}}{f_{.j}}$, for $i=1, 2, \ldots, n.$
with Metric M	$\mathbf{M} = \mathbf{D}_p^{-1}$ Chi-square distances $d^2(i,i') = \sum_{j=1}^{p} \frac{1}{f_{.j}} \left(\frac{f_{ij}}{f_{i.}} - \frac{f_{i'j}}{f_{i'.}} \right)^2$	$\mathbf{M} = \mathbf{D}_n^{-1}$ $d^2(j,j') = \sum_{i=1}^{n} \frac{1}{f_{i.}} \left(\frac{f_{ij}}{f_{.j}} - \frac{f_{ij'}}{f_{.j'}} \right)^2$
Criterion N	$\mathbf{N} = \mathbf{D}_n$ mass of point i : $f_{i.}$	$\mathbf{N} = \mathbf{D}_p$ mass of point j : $f_{.j}$

The reason for choosing the chi-square distance is: it verifies the property of distributional equivalency:

1. If two columns having identical profiles are aggregated, then the distances between rows remain unchanged.

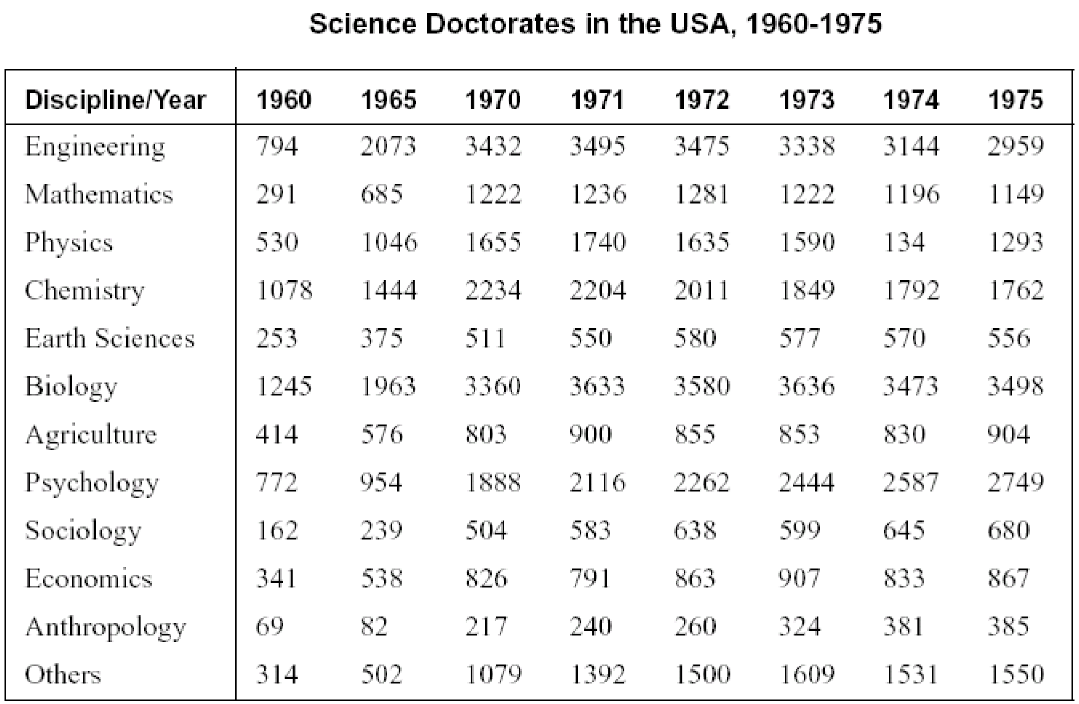
2. If two rows having identical distribution profiles are aggregated, then the distances between columns remain unchanged.

The property is important, because it guarantees a satisfactory invariance of the results irrespective of how the variables were originally coded.

	In R^{p}	In R^{n}
Matrix to diagonalize	$\mathbf{S} = \mathbf{F}'\mathbf{D}_n^{-1}\mathbf{F}\mathbf{D}_p^{-1}$	$\mathbf{T} = \mathbf{F}\mathbf{D}_p^{-1}\mathbf{F}'\mathbf{D}_n^{-1}$
Principal axes	$\mathbf{S}\mathbf{u}_\alpha = \lambda_\alpha \mathbf{u}_\alpha$	$\mathbf{T}\mathbf{v}_\alpha = \lambda_\alpha \mathbf{v}_\alpha$
Coordinates of points on the axes	$\psi_\alpha = \mathbf{D}_n^{-1}\mathbf{F}\mathbf{D}_p^{-1}\mathbf{u}_\alpha$ $\psi_{\alpha i} = \sum_{j=1}^{p} \frac{f_{ij}}{f_{i.}f_{.j}} u_{\alpha j}$	$\varphi_\alpha = \mathbf{D}_p^{-1}\mathbf{F}'\mathbf{D}_n^{-1}\mathbf{v}$ $\varphi_{\alpha j} = \sum_{i=1}^{n} \frac{f_{ij}}{f_{i.}f_{.j}} v_{\alpha i}$

Correspondance Analysis (conti.)

- Row points for the disciplines, Column points for the years.
- The anthropology degree and the engineering degree are far from each other because their profiles are different, mathematics degree is near the engineering degree because their profiles are similar.
- Each year point represents the profile of that year across the various disciplines.



Science Doctorates in the USA, 1960-1975

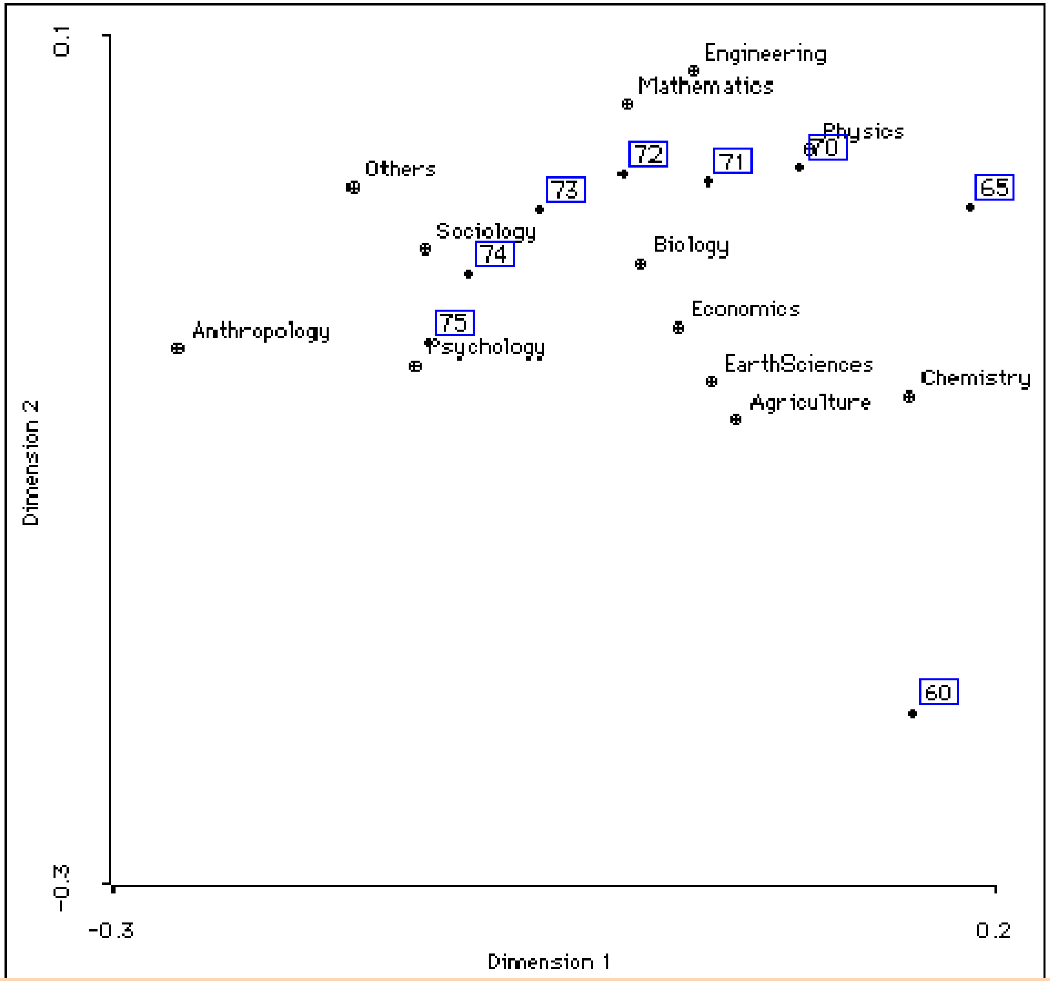
Discipline/Year	1960	1965	1970	1971	1972	1973	1974	1975
Engineering	794	2073	3432	3495	3475	3338	3144	2959
Mathematics	291	685	1222	1236	1281	1222	1196	1149
Physics	530	1046	1655	1740	1635	1590	134	1293
Chemistry	1078	1444	2234	2204	2011	1849	1792	1762
Earth Sciences	253	375	511	550	580	577	570	556
Biology	1245	1963	3360	3633	3580	3636	3473	3498
Agriculture	414	576	803	900	855	853	830	904
Psychology	772	954	1888	2116	2262	2444	2587	2749
Sociology	162	239	504	583	638	599	645	680
Economics	341	538	826	791	863	907	833	867
Anthropology	69	82	217	240	260	324	381	385
Others	314	502	1079	1392	1500	1609	1531	1550

The multidimensional time series on the number of science doctorates conferred in the USA from 1960 to 1975 (Greenacre, 1984).

Correspondance Analysis (conti.)

Interpretation

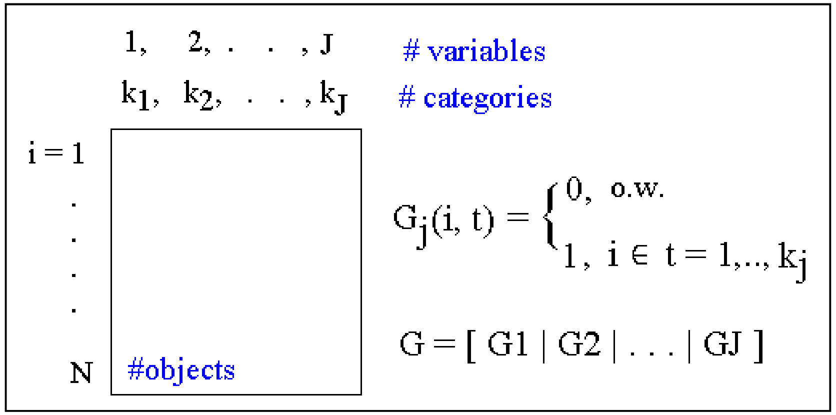
- Each discipline point will lie in the neighborhood of the year in which the discipline's profile is prominent.
- There are relatively more agriculture, earth science and chemistry degrees in 1960, while the trend from 1965 to 1975 appears to be away from the physical sciences towards the social sciences.
- The points such as earth sciences and economics lie within the parabolic configuration of the years points; this implies that the profiles of these disciplines are higher than average in the early and later years.



Note that the positions of two sets of points with respect to each other are not directly comparable and should be interpreted with caution.

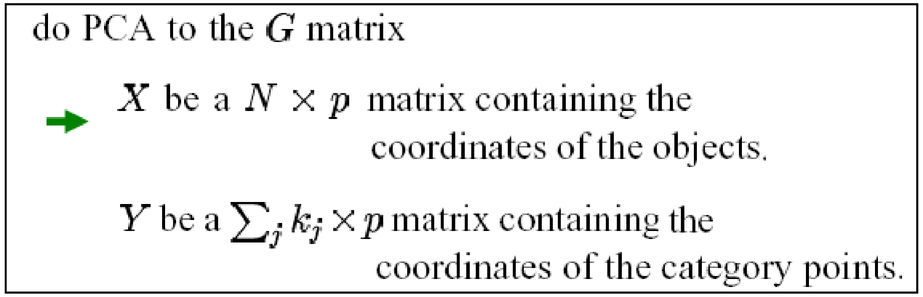
Multiple Correspondance Analysis (Homogeneity Analysis)

- Multiple Correspondence Analysis (MCA) is known as homogeneity analysis, or dual scaling, or reciprocal averaging.
- The general idea of homogeneity analysis is to make a joint plot in p-space of all objects (or individuals) and the categories of all variables.
- Objects close to the categories they fall in and categories close to objects belonging in them



$$G_j(i, t) = \begin{cases} 0, & \text{o.w.} \\ 1, & i \in t = 1,.., k_j \end{cases}$$

$$G = [\ G1 \mid G2 \mid . . . \mid GJ\ ]$$



do PCA to the G matrix

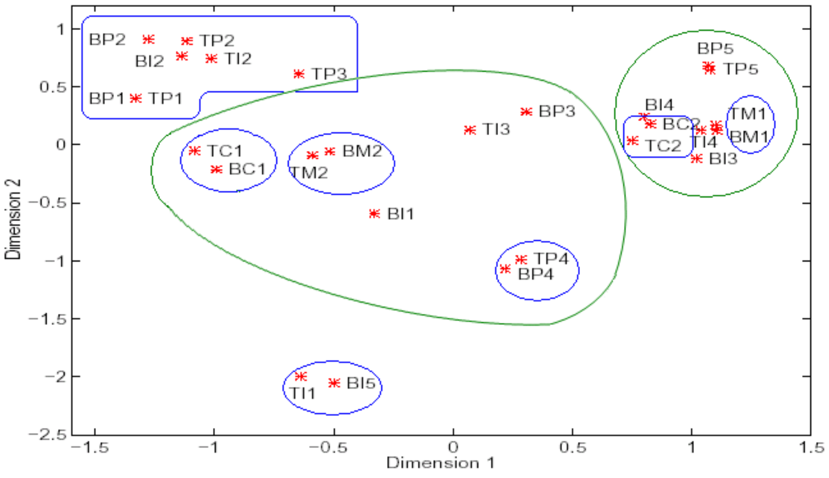
→ X be a $N \times p$ matrix containing the coordinates of the objects.

Y be a $\sum_j k_j \times p$ matrix containing the coordinates of the category points.

Homogeneity Analysis (conti.)

Mammals Dentition Example

The data for this example are taken from Hartigan (1975) (also discussed in Michailidis and De Leeuw,1999). Dental characteristics are used in the classification of 66 different kinds of mammals. Mammals' teeth are divided into four groups: incisors, canines, premolars, and molars.



Category quantifications of the variables in the mammals dentition example

Description for Variables

TI: Top incisors;
1: 0 incisors, 2: 1 incisors,
3: 2 incisors, 4: 3 or more incisors

BI: Bottom incisors;
1: 0 incisors, 2: 1 incisors,
3: 2 incisors, 4: 3 incisors
5: 4 incisors

TC: Top canine;
1: 0 canines, 2: 1 canines,

BC: Bottom canine;
1: 0 canines, 2: 1 canines,

TP: Top premolar;
1: 0 premolars, 2: 1 premolars,
3: 2 premolars, 4: 3 premolars
5: 4 premolars

BP: Bottom premolar;
1: 0 premolars, 2: 1 premolars,
3: 2 premolars, 4: 3 premolars
5: 4 premolars

TM: Top molar;
1: 0-2 molars, 2: 3 or more molars,

BM: Bottom molar;
1: 0-2 molars, 2: 3 or more molars

T	B	T	B	T	B	T	B	
I	I	C	C	P	P	M	M	
4	5	2	2	4	4	2	2	Opposum
4	4	2	2	5	5	2	2	Hairy-Tail-Mole
4	3	2	1	4	4	2	2	Common-Mole
4	4	2	2	5	5	2	2	Star-Nose-Mole
3	4	2	2	4	4	2	2	Brown-Bat
3	4	2	2	3	4	2	2	Silver-Hair-Bat
3	4	2	2	3	3	2	2	Pigmy-Bat
3	4	2	2	2	3	2	2	House-Bat
2	4	2	2	3	3	2	2	Red-Bat
2	4	2	2	3	3	2	2	Hoary-Bat
3	4	2	2	3	4	2	2	Lump-Nose-Bat
1	1	1	1	1	1	2	2	Armadillo
3	2	1	1	3	3	2	2	Pika
3	2	1	1	4	3	2	2	Snowshoe-Rabit
2	2	1	1	3	2	2	2	Beaver
2	2	1	1	3	2	2	2	Marmot
2	2	1	1	3	2	2	2	Groundhog
2	2	1	1	3	2	2	2	Prairie-Dog
2	2	1	1	3	2	2	2	Ground-Squirrel
2	2	1	1	3	2	2	2	Chipmunk
2	2	1	1	2	2	2	2	Gray-Squirrel
2	2	1	1	2	2	2	2	Fox-Squirrel
2	2	1	1	2	2	2	2	Pocket-Gopher
2	2	1	1	2	2	2	2	Kangaroo-Rat
2	2	1	1	1	1	2	2	Pack-Rat
2	2	1	1	1	1	2	2	Field-Mouse
2	2	1	1	1	1	2	2	Muskrat
2	2	1	1	1	1	2	2	Black-Rat
2	2	1	1	1	1	2	2	House-Mouse
2	2	1	1	2	2	2	2	Porcupine
2	2	1	1	2	2	2	2	Guinea-Pig
2	4	2	2	5	5	2	2	Coyote
4	4	2	2	5	5	1	2	Wolf
4	4	2	2	5	5	1	2	Fox
4	4	2	2	5	5	1	2	Bear
4	4	2	2	5	5	1	1	Civet-Cat
4	4	2	2	5	5	2	1	Raccoon
4	4	2	2	5	5	1	1	Marten
4	4	2	2	5	5	1		Fisher
4	4	2	2	4	4	1	1	Weasel
4	4	2	2	4	4	1	1	Mink
4	4	2	2	4	4	1	1	Ferrer
4	4	2	2	5	5	1	1	Wolverine
4	4	2	2	4	4	1	1	Badger
4	4	2	2	4	4	1	1	Skunk
4	4	2	2	5	4	1	1	River-Otter
4	3	2	2	4	4	1	1	Sea-Otter
4	4	2	2	4	3	1	1	Jaguar
4	4	2	2	4	3	1	1	Ocelot
4	4	2	2	4	3	1	1	Cougar
4	4	2	2	4	3	1	1	Lynx
4	3	2	2	5	5	1	1	Fur-Seal
4	3	2	2	5	5	1	1	Sea-Lion
2	1	2	2	4	4	1	1	Walrus
4	3	2	2	4	4	1	1	Grey-Seal
3	2	2	2	5	5	1		Elephant-Seal
3	4	2	2	4	4	2	2	Peccary
1	5	2	1	4	4	2	2	Elk
1	5	1	1	4	4	2	2	Deer
1	5	1	1	4	4	2	2	Moose
1	5	2	1	4	4	2	2	Reindeer
1	5	1	1	4	4	2	2	Antelope
1	5	1	1	4	4	2	2	Bison
1	5	1	1	4	4	2	2	Mountain-Goat
1	5	1	1	4	4	2	2	Muskox
1	5	1	1	4	4	2	2	Mountain-Sheep