

Simpson's Paradox

coop711

2015년 5월 12일

UCB Admissions Case

1973년 UC Berkeley 대학원은 자신이 여성이라는 이유로 입학허가에서 차별을 받았다는 여성의 소송에 휘말렸다.

```
str(UCBAdmissions)
```

```
## table [1:2, 1:2, 1:6] 512 313 89 19 353 207 17 8 120 205 ...
## - attr(*, "dimnames")=List of 3
## ..$ Admit : chr [1:2] "Admitted" "Rejected"
## ..$ Gender: chr [1:2] "Male" "Female"
## ..$ Dept : chr [1:6] "A" "B" "C" "D" ...
```

```
class(UCBAdmissions)
```

```
## [1] "table"
```

```
attributes(UCBAdmissions)
```

```
## $dim
## [1] 2 2 6
##
## $dimnames
## $dimnames$Admit
## [1] "Admitted" "Rejected"
##
## $dimnames$Gender
## [1] "Male" "Female"
##
## $dimnames$Dept
## [1] "A" "B" "C" "D" "E" "F"
##
##
## $class
## [1] "table"
```

```
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
```

Table and Data Frame에서 살펴본 바와 같이 3차원 array 구조를 가진 `UCBAdmissions` 를 한 눈에 파악하기 위하여 `ftable()` 을 이용하였다. 성별 입학허가를 비교하기 위하여 `Admit` 과 `Gender` 의 위치를 바꾸는 다양한 방법이 있음을 알 수 있다.

```
ftable(UCBAdmissions)
```

```
##           Dept    A    B    C    D    E    F
## Admit      Gender
## Admitted Male      512 353 120 138  53  22
##           Female      89  17 202 131  94  24
## Rejected Male      313 207 205 279 138 351
##           Female      19   8 391 244 299 317
```

```
ftable(UCBAdmissions, row.vars=2:1)
```

```
##           Dept    A    B    C    D    E    F
## Gender Admit
## Male   Admitted      512 353 120 138  53  22
##       Rejected      313 207 205 279 138 351
## Female Admitted      89  17 202 131  94  24
##       Rejected      19   8 391 244 299 317
```

```
ftable(UCBAdmissions, row.vars=c("Gender", "Admit"))
```

```
##           Dept    A    B    C    D    E    F
## Gender Admit
## Male   Admitted      512 353 120 138  53  22
##       Rejected      313 207 205 279 138 351
## Female Admitted      89  17 202 131  94  24
##       Rejected      19   8 391 244 299 317
```

또한 `ftable()` 로 만든 `table`이 4×6 매트릭스이고 `$dim` 과 `$class` 외에도 `$row.vars` , `$col.vars` 요소를 포함하고 있다는 것을 알 수 있다.

```
attributes(ftable(UCBAdmissions))
```

```
## $dim
## [1] 4 6
##
## $class
## [1] "ftable"
##
## $row.vars
## $row.vars$Admit
## [1] "Admitted" "Rejected"
##
## $row.vars$Gender
## [1] "Male"    "Female"
##
##
## $col.vars
## $col.vars$Dept
## [1] "A" "B" "C" "D" "E" "F"
```

3차원 array에서 각 요소를 추출하는 방법은 다음과 같다. 우선 입학허가자 중 남자들만 뽑아보면,

```
UCBAdmissions[1, 1, ]
```

```
##      A      B      C      D      E      F
## 512 353 120 138   53   22
```

와 같이 전공별로 주어짐을 알 수 있고, 입학허가자 전체를 살펴보면,

```
UCBAdmissions[1, , ]
```

```
##           Dept
## Gender      A      B      C      D      E      F
##   Male    512 353 120 138  53  22
##   Female   89  17 202 131  94  24
```

와 같이 성별, 전공별로 나뉘어짐을 알 수 있다. 지원한 남성들을 입학허가 여부와 전공별로 나눠 보면,

```
UCBAdmissions[, 1, ]
```

```
##           Dept
## Admit      A      B      C      D      E      F
##   Admitted 512 353 120 138   53   22
##   Rejected 313 207 205 279 138 351
```

A전공과 B전공에 많은 인원이 지원하고 입학허가도 많이 나왔음을 알 수 있다.

여자들을 입학허가 여부와 전공별로 나눠 보면,

```
UCBAdmissions[, 2, ]
```

```
##           Dept
## Admit      A      B      C      D      E      F
##   Admitted 89 17 202 131   94   24
##   Rejected 19   8 391 244 299 317
```

남학생들이 많이 지원한 A전공과 B전공에는 적은 수효가 지원하였음을 알 수 있다. `apply()` 를 이용하여 집계를 내는 방법에 대하여 알아보자. 우선, 입학허가 여부에 대하여 집계를 내어보면,

```
apply(UCBAdmissions, 1, sum)
```

```
## Admitted Rejected
##      1755      2771
```

입학이 허가된 인원보다 그렇지 않은 인원이 훨씬 많았고, 남녀 지원자수의 합계를 비교해 보면

```
apply(UCBAdmissions, 2, sum)
```

```
##      Male Female
##    2691    1835
```

남학생이 더 많이 지원했음을 알 수 있다. 전공별 지원자수를 살펴보면,

```
apply(UCBAdmissions, 3, sum)
```

```
##      A      B      C      D      E      F
##  933  585  918  792  584  714
```

A전공과 B전공에 많은 지원이 있었음을 알 수 있다. 남녀별, 입학허가 여부별로 집계해 보면,

```
apply(UCBAdmissions, c(1, 2), sum)
```

```
##              Gender
## Admit          Male Female
##   Admitted 1198      557
##   Rejected 1493     1278
```

얼핏 윤곽은 파악되나, 자세한 내역은 백분율을 내어 봐야 함을 알 수 있다. 성별, 전공별 집계를 내어보면,

```
apply(UCBAdmissions, c(2, 3), sum)
```

```
##              Dept
## Gender          A      B      C      D      E      F
##   Male    825  560  325  417  191  373
##   Female  108   25  593  375  393  341
```

A전공, B전공은 남학생이 월등히 많이 지원하였고, C전공, E전공에는 여학생이 많이 지원했음을 알 수 있다. 입학허가 여부별, 전공별 집계를 내보면,

```
apply(UCBAdmissions, c(1, 3), sum)
```

```
##              Dept
## Admit          A      B      C      D      E      F
##   Admitted 601  370  322  269  147   46
##   Rejected 332  215  596  523  437  668
```

A전공, B전공은 쉽게 허가가 나오는 반면에 F전공은 매우 입학허가 나오기 어려운 전공임을 알 수 있다. 각 전공별로 남녀의 입학허가 및 탈락 비율을 비교하려면,

```
options(digits=2)
prop.table(UCBAdmissions, margin=2:3)
```

```
## , , Dept = A
##
##           Gender
## Admit      Male Female
##   Admitted 0.621  0.824
##   Rejected 0.379  0.176
##
## , , Dept = B
##
##           Gender
## Admit      Male Female
##   Admitted 0.630  0.680
##   Rejected 0.370  0.320
##
## , , Dept = C
##
##           Gender
## Admit      Male Female
##   Admitted 0.369  0.341
##   Rejected 0.631  0.659
##
## , , Dept = D
##
##           Gender
## Admit      Male Female
##   Admitted 0.331  0.349
##   Rejected 0.669  0.651
##
## , , Dept = E
##
##           Gender
## Admit      Male Female
##   Admitted 0.277  0.239
##   Rejected 0.723  0.761
##
## , , Dept = F
##
##           Gender
## Admit      Male Female
##   Admitted 0.059  0.070
##   Rejected 0.941  0.930
```

와 같은 방법을 쓸 수 있다. `margin=2:3` 이 전공별, 성별을 의미한다는 것을 알 수 있다. 한 눈에 파악할 수 있도록 `ftable()` 을 사용하되 `margin=2:3` 으로 하여 성별, 전공별로 입학허가여부의 비율을 합하면 1이 되도록 하였다.

```
ftable(prop.table(UCBAdmissions, margin=2:3))
```

```
##           Dept      A      B      C      D      E      F
## Admit      Gender
## Admitted Male      0.621 0.630 0.369 0.331 0.277 0.059
##           Female      0.824 0.680 0.341 0.349 0.239 0.070
## Rejected Male      0.379 0.370 0.631 0.669 0.723 0.941
##           Female      0.176 0.320 0.659 0.651 0.761 0.930
```

성별, 입학허가율을 비교하기 위하여 `row.vars=2:1` 로 하여 순서를 바꾼다.

```
ftable(prop.table(UCBAdmissions, margin=2:3), row.vars=2:1)
```

```
##           Dept      A      B      C      D      E      F
## Gender Admit
## Male   Admitted      0.621 0.630 0.369 0.331 0.277 0.059
##       Rejected      0.379 0.370 0.631 0.669 0.723 0.941
## Female Admitted      0.824 0.680 0.341 0.349 0.239 0.070
##       Rejected      0.176 0.320 0.659 0.651 0.761 0.930
```

이 중 입학허가율만 비교한다면,

```
prop.table(UCBAdmissions, margin=2:3)[1, , ]
```

```
##           Dept
## Gender      A      B      C      D      E      F
## Male   0.62 0.63 0.37 0.33 0.28 0.059
## Female 0.82 0.68 0.34 0.35 0.24 0.070
```

임을 알 수 있어서 오히려 여자들의 입학허가율이 전공별로는 더 높거나 거의 같은 수준임을 알 수 있다. 이는 그 위의 결과를 이용하여

```
ftable(prop.table(UCBAdmissions, margin=2:3))[1:2, ]
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,] 0.62 0.63 0.37 0.33 0.28 0.059
## [2,] 0.82 0.68 0.34 0.35 0.24 0.070
```

라고 하여도 되지만 이름을 잃어버린다. 한편 전체적으로는

```
options(digits=3)
prop.table(apply(UCBAdmissions, c(1, 2), sum), margin=2)
```

```
##           Gender
## Admit      Male Female
## Admitted 0.445  0.304
## Rejected 0.555  0.696
```

남자들의 입학허가율이 여자들의 입학허가율보다 높다. 이는 다음 식으로부터 보다 확실히 알 수 있다.

```
prop.table(apply(UCBAdmissions, c(1,2), sum), margin=2)[1, ]
```

```
##      Male Female  
## 0.445 0.304
```

전공별 지원자를 집계해 보면,

```
apply(UCBAdmissions, 3, sum)
```

```
##      A      B      C      D      E      F  
## 933 585 918 792 584 714
```

성별, 전공별 입학허가율만 따로 떼어 내면,

```
prop.table(UCBAdmissions, margin=2:3)[1, , ]
```

```
##              Dept  
## Gender          A      B      C      D      E      F  
## Male 0.621 0.63 0.369 0.331 0.277 0.0590  
## Female 0.824 0.68 0.341 0.349 0.239 0.0704
```

전공별 지원자수를 새로운 R object로 저장하고,

```
total.applicants.major<-apply(UCBAdmissions, 3, sum)
```

총 지원자수를 또 다른 R object로 저장한다.

```
total.applicants<-sum(total.applicants.major)  
total.applicants
```

```
## [1] 4526
```

남녀별 입학허가율을 새로운 R 객체로 저장하면,

```
admitted.rates.m.major<-prop.table(UCBAdmissions,margin=2:3)[1, 1, ]  
admitted.rates.f.major<-prop.table(UCBAdmissions,margin=2:3)[1, 2, ]
```

남자의 전공별 입학허가율은,

```
admitted.rates.m.major
```

```
##      A      B      C      D      E      F  
## 0.621 0.630 0.369 0.331 0.277 0.059
```

이고, 여자의 전공별 입학허가율은,


```
admitted.rates.f.major
```

```
##           A           B           C           D           E           F
## 0.8241 0.6800 0.3406 0.3493 0.2392 0.0704
```

으로 계산되어 여자들의 입학허가율이 높거나 대체로 비슷함을 알 수 있다. 이 여섯 개의 입학허가율을 전공별 지원자 수를 가중치로 고려한 하나의 입학허가율로 계산하면,

```
admitted.rates.m<-sum(admitted.rates.m.major*total.applicants.major)/total.applicants
admitted.rates.f<-sum(admitted.rates.f.major*total.applicants.major)/total.applicants
c(male=admitted.rates.m, female=admitted.rates.f)
```

```
##      male female
## 0.387 0.430
```

와 같이 계산되어 전공을 고려하지 않고 계산한 남녀별 입학허가율과는 반대의 결과가 나온다.

GLM Approach with Cases Data Frame

입학허가 여부를 이항 변수로 보고, 성별과 학과를 독립변수로 보는 glm 모델을 생각해보자. 이 모델을 적합시키려면 case 별로 기록된 data frame이 필요하다. 입학허가에 성별 차이가 있는지 파악하기 위하여 logit을 link로 하는 binomial family에 적합시켜 보자.

```
load("UCB_glm.rda")
UCB.glm.1<-glm(Admit~Gender, family=binomial(link="logit"), data=UCBAdmissions.cases)
UCB.glm.1
```

```
##
## Call:  glm(formula = Admit ~ Gender, family = binomial(link = "logit"),
##         data = UCBAdmissions.cases)
##
## Coefficients:
## (Intercept)  GenderFemale
##          0.22          0.61
##
## Degrees of Freedom: 4525 Total (i.e. Null); 4524 Residual
## Null Deviance:      6040
## Residual Deviance: 5950 AIC: 5950
```

```
summary(UCB.glm.1)
```

```
##
## Call:
## glm(formula = Admit ~ Gender, family = binomial(link = "logit"),
##      data = UCBA admissions.cases)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -1.544   -1.272    0.851    1.085    1.085
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.2201     0.0388   5.68 1.4e-08 ***
## GenderFemale    0.6104     0.0639   9.55 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 6044.3  on 4525  degrees of freedom
## Residual deviance: 5950.9  on 4524  degrees of freedom
## AIC: 5955
##
## Number of Fisher Scoring iterations: 4
```

성별의 계수가 통계적으로 매우 유의하게 나오고 있어서 남녀 간 입학허가의 가능성에 차이가 있음을 알 수 있다. 이를 회귀계수를 중심으로 보다 자세히 살펴보면

```
coef(UCB.glm.1)
```

```
## (Intercept) GenderFemale
##          0.22          0.61
```

이 값은 바로 $\log \frac{(1-p_{female})/p_{female}}{(1-p_{male})/p_{male}}$ 에 해당한다. 이는 R의 `glm()` 에서 success를 취급하는 방식에 기인하는데 이를 `binomial()` 의 help 파일에서 인용하면 다음과 같다.

Details:

'family' is a generic function with methods for classes "glm" and "lm" (the latter returning 'gaussian()').

For the 'binomial' and 'quasibinomial' families the response can be specified in one of three ways:

1. As a factor: 'success' is interpreted as the factor not having the first level (and hence usually of having the second level).
2. As a numerical vector with values between '0' and '1', interpreted as the proportion of successful cases (with the total number of cases given by the 'weights').
3. As a two-column integer matrix: the first column gives the number of successes and the second the number of failures.

The 'quasibinomial' and 'quasipoisson' families differ from the 'binomial' and 'poisson' families only in that the dispersion parameter is not fixed at one, so they can model over-dispersion. For the binomial case see McCullagh and Nelder (1989, pp. 124-8). Although they show that there is (under some restrictions) a model with variance proportional to mean as in the quasi-binomial model, note that 'glm' does not compute maximum-likelihood estimates in that model. The behaviour of S is closer to the quasi- variants.

위에서 계산한 바 있는 남성과 여성의 전체 입학허가율을 이 식에 대입해 보면 같은 값을 얻게 된다.

```
p.gender<-prop.table(xtabs(Freq ~ Admit+Gender, data = UCBAAdmissions.df), margin=2)
p.gender
```

```
##           Gender
## Admit      Male Female
##   Admitted 0.445  0.304
##   Rejected 0.555  0.696
```

```
log(p.gender[2,2]/p.gender[1,2]/(p.gender[2,1]/p.gender[1,1]))
```

```
## [1] 0.61
```

log(odds ratio) 로 의미 파악이 잘 안되면 `exp()` 을 취하여 살펴볼 수도 있다.

```
exp(coef(UCB.glm.1))
```

```
## (Intercept) GenderFemale
##           1.25          1.84
```

즉, 학과를 고려하지 않았을 때 여성 불합격률의 odds ratio가 남성보다 1.8배 높다는 의미이다. 이는 남성 입학허가율의 odds ratio가 여성보다 1.8배 높다는 의미이기도 하다.

성별 차이에도 학과별 차이를 고려한 모델은

```
UCB.glm.2<-glm(Admit~Gender+Dept, family=binomial(link="logit"), data=UCBAdmissions.cases)
UCB.glm.2
```

```
##
## Call:  glm(formula = Admit ~ Gender + Dept, family = binomial(link = "logit"),
##      data = UCBAdmissions.cases)
##
## Coefficients:
## (Intercept)  GenderFemale      DeptB      DeptC      DeptD
##      -0.5821      -0.0999      0.0434      1.2626      1.2946
##      DeptE      DeptF
##      1.7393      3.3065
##
## Degrees of Freedom: 4525 Total (i.e. Null);  4519 Residual
## Null Deviance:      6040
## Residual Deviance: 5190  AIC: 5200
```

```
summary(UCB.glm.2)
```

```
##
## Call:
## glm(formula = Admit ~ Gender + Dept, family = binomial(link = "logit"),
##      data = UCBAdmissions.cases)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.361  -0.959   0.374   0.931   1.477
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.5821     0.0690  -8.44  <2e-16 ***
## GenderFemale -0.0999     0.0808  -1.24    0.22
## DeptB         0.0434     0.1098   0.40    0.69
## DeptC         1.2626     0.1066  11.84  <2e-16 ***
## DeptD         1.2946     0.1058  12.23  <2e-16 ***
## DeptE         1.7393     0.1261  13.79  <2e-16 ***
## DeptF         3.3065     0.1700  19.45  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 6044.3  on 4525  degrees of freedom
## Residual deviance: 5187.5  on 4519  degrees of freedom
## AIC: 5201
##
## Number of Fisher Scoring iterations: 5
```

이제, 성별 차이는 더 이상 유의하지 않고, 학과별 차이(C, D, E, F)가 지배적이다. 성별 분석과 마찬가지로 계수를 살펴 보면,

```
coef(UCB.glm.2)
```

```
## (Intercept) GenderFemale      DeptB      DeptC      DeptD
##      -0.5821      -0.0999      0.0434      1.2626      1.2946
##      DeptE      DeptF
##      1.7393      3.3065
```

이제 성별차이는 거의 없다는 것, 오히려 여성들이 약간 더 유리하다는 것을 다음 odds ratio로 확인할 수 있다. 여성들의 불합격률의 odds ratio는 남성 불합격률 odds ratio의 90% 수준이라는 의미이다.

```
exp(coef(UCB.glm.2)[ "GenderFemale" ])
```

```
## GenderFemale
##           0.905
```

학과별 입학허가 가능성(사실은 R의 binomial family 구성에 따라 불합격 가능성)은 기준인 A학과와의 비교로 이루어진다. B학과의 경우

```
exp(coef(UCB.glm.2)[ "DeptB" ])
```

```
## DeptB
##      1.04
```

로 A학과보다 약간 높다는 것을 알 수 있는 데, 이는 성별을 고려한 입학허가율(혹은 불합격률)에 있어서 남자들의 경우

```
matrix(data=male.admissions, nrow=2, ncol=6, dimnames=dim.names.UCB)
```

```
##           Dept
## Admit      A      B      C      D      E      F
## Admitted 0.621 0.63 0.369 0.331 0.277 0.059
## Rejected 0.379 0.37 0.631 0.669 0.723 0.941
```

여자들의 경우

```
matrix(data=female.admissions, nrow=2, ncol=6, dimnames=dim.names.UCB)
```

```
##           Dept
## Admit      A      B      C      D      E      F
## Admitted 0.824 0.68 0.341 0.349 0.239 0.0704
## Rejected 0.176 0.32 0.659 0.651 0.761 0.9296
```

이었던 점을 감안하여 각각의 입학허가율(혹은 불합격률)을 성별 지원자를 가중치로 넣은 가중평균으로 계산한 결과로부터 비교할 수 있는 일이다. 즉,

```
c(male=admitted.rates.m, female=admitted.rates.f)
```

```
##    male female  
## 0.387 0.430
```

이므로, 여성불합격률과 남성불합격률의 odds ratio의 비는

```
((1-admitted.rates.f)/admitted.rates.f)/((1-admitted.rates.m)/admitted.rates.m)
```

```
## [1] 0.838
```

로 나와 회귀계수로부터 예상한 결과와 대체적으로 부합한다.

성별, 학과별 지원자수는

```
applicants.gender.dept<-table(UCBAdmissions.cases[c("Gender","Dept")])  
applicants.gender.dept
```

```
##           Dept  
## Gender      A   B   C   D   E   F  
##   Male    825 560 325 417 191 373  
##   Female  108  25 593 375 393 341
```

로부터 확인가능하다. 예를 들어서 A학과의 입학허가율은

```
A.admissions<-(male.admissions[1,1]*applicants.gender.dept[1,1]+female.admissions[1,1]*applicants.gender.dept[2,1])/colSums(applicants.gender.dept)[1]  
A.admissions
```

```
##      A  
## 0.644
```

B학과의 입학허가율은

```
B.admissions<-(male.admissions[1,2]*applicants.gender.dept[1,2]+female.admissions[1,2]*applicants.gender.dept[2,2])/colSums(applicants.gender.dept)[2]  
B.admissions
```

```
##      B  
## 0.632
```

로 계산되어 불합격률의 odds ratio, $\frac{1-B.admissions}{B.admissions} = 0.581$ 와 비교했을 때 예상했던 대로 A학과 불합격률의 odds ratio, 0.552보다 약간 높다. 그밖에 다른 학과들의 입학허가율을 지원자수를 고려한 가중평균으로 계산하면,

```
(male.admissions[1,3]*applicants.gender.dept[1,3]+female.admissions[1,3]*applicants.gender.dept[2,3])/colSums(applicants.gender.dept)[3]
```

```
##      C
## 0.351
```

```
(male.admissions[1,4]*applicants.gender.dept[1,4]+female.admissions[1,4]*applicants.gender.dept[2,4])/colSums(applicants.gender.dept)[4]
```

```
##      D
## 0.34
```

```
(male.admissions[1,5]*applicants.gender.dept[1,5]+female.admissions[1,5]*applicants.gender.dept[2,5])/colSums(applicants.gender.dept)[5]
```

```
##      E
## 0.252
```

```
(male.admissions[1,6]*applicants.gender.dept[1,6]+female.admissions[1,6]*applicants.gender.dept[2,6])/colSums(applicants.gender.dept)[6]
```

```
##      F
## 0.0644
```

로 계산되어 학과별 odds ratio비교가 가능해진다. F학과의 경우 C학과와 비교했을 때,

```
F.C<-exp(coef(UCB.glm.2)[ "DeptF" ])/exp(coef(UCB.glm.2)[ "DeptC" ])
names(F.C)<-"F to C odds ratio"
F.C
```

```
## F to C odds ratio
##              7.72
```

만큼 불합격률의 odds ratio가 차이날 것으로 판단되는 데 이는 $\frac{(1-0.0644)/0.0644}{(1-0.351)/0.351} = 7.857$ 로 쉽게 확인된다.

anova 로 두 모델 간의 차이를 분석하면

```
anova(UCB.glm.1, UCB.glm.2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Admit ~ Gender
## Model 2: Admit ~ Gender + Dept
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      4524      5951
## 2      4519      5187  5      763    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

즉, 학과별 영향을 고려하면서 성별 영향은 사라진 모델이 적합함을 알 수 있다. 이와 같은 방식으로 제3의 변수가 주는

영향을 파악할 수 있다.

이와 같은 경우, case별로 재구성한 data frame이 필요하다.

물론 counts로 구성된 data frame 으로도 glm 분석이 가능하다.

GLM (data frame with Counts)

binomial family 를 갖는 glm 에 맞추려면 UCBAmissions.df 에서 \$Admit 의 "Admitted" 와 "Rejected" 를 별도의 변수로 떼 내어야 한다.

```
UCBAmissions.df
```

```
##      Admit Gender Dept Freq
## 1  Admitted   Male    A  512
## 2  Rejected   Male    A  313
## 3  Admitted Female    A   89
## 4  Rejected Female    A   19
## 5  Admitted   Male    B  353
## 6  Rejected   Male    B  207
## 7  Admitted Female    B   17
## 8  Rejected Female    B    8
## 9  Admitted   Male    C  120
## 10 Rejected   Male    C  205
## 11 Admitted Female    C  202
## 12 Rejected Female    C  391
## 13 Admitted   Male    D  138
## 14 Rejected   Male    D  279
## 15 Admitted Female    D  131
## 16 Rejected Female    D  244
## 17 Admitted   Male    E   53
## 18 Rejected   Male    E  138
## 19 Admitted Female    E   94
## 20 Rejected Female    E  299
## 21 Admitted   Male    F   22
## 22 Rejected   Male    F  351
## 23 Admitted Female    F   24
## 24 Rejected Female    F  317
```

Admitted 를 Admit=="Admitted" 를 테스트하는 논리변수라하면,

```
Admitted<-UCBAmissions.df$Admit=="Admitted"
```

Admitted와 Rejected를 새로운 Counts 변수로 갖는 data frame은,

```
UCBAmissions.df.2<-data.frame(UCBAmissions.df[Admitted, 2:3], Admitted=UCBAmissions.df[Admitted, 4], Rejected=UCBAmissions.df[!Admitted, 4])
UCBAmissions.df.2
```



```
##      Gender Dept Admitted Rejected
## 1      Male    A      512      313
## 3    Female    A       89       19
## 5      Male    B      353      207
## 7    Female    B       17       8
## 9      Male    C      120      205
## 11   Female    C      202      391
## 13     Male    D      138      279
## 15   Female    D      131      244
## 17     Male    E       53      138
## 19   Female    E       94      299
## 21     Male    F       22      351
## 23   Female    F       24      317
```

이렇게 새로이 만든 data frame에 다음과 같은 모델을 적용한다.

```
UCB.glm.3<-glm(cbind(Admitted, Rejected)~Gender, family=binomial(logit), data=UCBAdmissions.df.2)
```

data frame with cases로 분석하였을 때와의 차이점은 "Admitted" 를 success 로 보고 있다는 점이다. 따라서 회귀계수의 부호가 반대로 나올 것으로 예측할 수 있다.

```
summary(UCB.glm.3)
```

```
##
## Call:
## glm(formula = cbind(Admitted, Rejected) ~ Gender, family = binomial(logit),
##      data = UCBAdmissions.df.2)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q        Max
## -16.791    -4.761    -0.436     5.103    11.202
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.2201     0.0388   -5.68  1.4e-08 ***
## GenderFemale  -0.6104     0.0639   -9.55 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 877.06  on 11  degrees of freedom
## Residual deviance: 783.61  on 10  degrees of freedom
## AIC: 856.5
##
## Number of Fisher Scoring iterations: 4
```

부호가 바뀐 점을 제외하면 회귀계수가 동일함을 확인할 수 있다. 전공을 고려하면,

```
UCB.glm.4<-glm(cbind(Admitted, Rejected)~Gender+Dept, family=binomial(logit), data=UCBAdmissions.df.2)
```

이 또한 부호만 바뀌고 회귀계수는 동일하다.

```
summary(UCB.glm.4)
```

```
##
## Call:
## glm(formula = cbind(Admitted, Rejected) ~ Gender + Dept, family = binomial(logit),
##      data = UCBAdmissions.df.2)
##
## Deviance Residuals:
##      1      3      5      7      9     11     13     15     17
## -1.249   3.719  -0.056   0.271   1.253  -0.924   0.083  -0.086   1.221
##     19     21     23
## -0.851  -0.208   0.205
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.5821    0.0690   8.44  <2e-16 ***
## GenderFemale    0.0999    0.0808   1.24    0.22
## DeptB          -0.0434    0.1098  -0.40    0.69
## DeptC          -1.2626    0.1066 -11.84  <2e-16 ***
## DeptD          -1.2946    0.1058 -12.23  <2e-16 ***
## DeptE          -1.7393    0.1261 -13.79  <2e-16 ***
## DeptF          -3.3065    0.1700 -19.45  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 877.056  on 11  degrees of freedom
## Residual deviance:  20.204  on  5  degrees of freedom
## AIC: 103.1
##
## Number of Fisher Scoring iterations: 4
```

anova 로 두 모델 간의 차이를 분석하면,

```
anova(UCB.glm.3, UCB.glm.4, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: cbind(Admitted, Rejected) ~ Gender
## Model 2: cbind(Admitted, Rejected) ~ Gender + Dept
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1          10          784
## 2           5           20  5       763  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

이 역시 같은 분석 결과를 얻는다.

GLM (data frame with Proportion)

이번에는 남자의 합격비율을 계산하여 모델에 넣는다.

```
UCB.total<-UCBAdmissions.df.2$Admitted+UCBAdmissions.df.2$Rejected
UCBAdmissions.df.2$P.Admitted<-UCBAdmissions.df.2$Admitted/UCB.total
UCBAdmissions.df.2
```

```
##      Gender Dept Admitted Rejected P.Admitted
## 1    Male    A     512      313    0.6206
## 3  Female    A      89       19    0.8241
## 5    Male    B     353      207    0.6304
## 7  Female    B      17        8    0.6800
## 9    Male    C     120      205    0.3692
## 11 Female    C     202      391    0.3406
## 13   Male    D     138      279    0.3309
## 15 Female    D     131      244    0.3493
## 17   Male    E      53      138    0.2775
## 19 Female    E      94      299    0.2392
## 21   Male    F      22      351    0.0590
## 23 Female    F      24      317    0.0704
```

glm 에 넣으려면 각 전공별 지원자수(UCB.total)를 weights= 에 설정해 주어야 한다.

```
UCB.glm.5<-glm(P.Admitted~Gender, family=binomial(logit), data=UCBAdmissions.d
f.2, weights=UCB.total)
```

data frame with counts를 활용한 분석과 같은 결과를 얻는다.

```
summary(UCB.glm.5)
```

```
##
## Call:
## glm(formula = P.Admitted ~ Gender, family = binomial(logit),
##      data = UCBA admissions.df.2, weights = UCB.total)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -16.791   -4.761   -0.436    5.103   11.202
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.2201     0.0388   -5.68  1.4e-08 ***
## GenderFemale -0.6104     0.0639   -9.55 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 877.06  on 11  degrees of freedom
## Residual deviance: 783.61  on 10  degrees of freedom
## AIC: 856.5
##
## Number of Fisher Scoring iterations: 4
```

전공별 상황까지 고려하면,

```
UCB.glm.6<-glm(P.Admitted~Gender+Dept, family=binomial(logit), data=UCBA admissions.df.2, weights=UCB.total)
```

이 역시 data frame with counts를 활용한 분석과 같은 결과가 나왔다.

```
summary(UCB.glm.6)
```

```
##
## Call:
## glm(formula = P.Admitted ~ Gender + Dept, family = binomial(logit),
##      data = UCBA admissions.df.2, weights = UCB.total)
##
## Deviance Residuals:
##      1      3      5      7      9     11     13     15     17
## -1.249   3.719  -0.056   0.271   1.253  -0.924   0.083  -0.086   1.221
##     19     21     23
## -0.851  -0.208   0.205
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.5821    0.0690   8.44  <2e-16 ***
## GenderFemale    0.0999    0.0808   1.24    0.22
## DeptB          -0.0434    0.1098  -0.40    0.69
## DeptC          -1.2626    0.1066 -11.84  <2e-16 ***
## DeptD          -1.2946    0.1058 -12.23  <2e-16 ***
## DeptE          -1.7393    0.1261 -13.79  <2e-16 ***
## DeptF          -3.3065    0.1700 -19.45  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 877.056  on 11  degrees of freedom
## Residual deviance:  20.204  on  5  degrees of freedom
## AIC: 103.1
##
## Number of Fisher Scoring iterations: 4
```

이 역시 anova 로 분석하면,

```
anova(UCB.glm.5, UCB.glm.6, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: P.Admitted ~ Gender
## Model 2: P.Admitted ~ Gender + Dept
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1         10         784
## 2          5          20    5      763  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

같은 결과를 얻게 된다.

Smoking and Survival

1972년부터 1974년 사이에 영국의 위컴(Whickham)에서 선거등록부에 올라와 있는 주민 여섯 명 1명 꼴로 실시한 조사결과와 그로부터 20년 후에 실시한 추수조사(follow-up study)결과를 비교한다. 자료를 읽어들이어서 data frame으로 저장한다.

```
s<-read.table("Smoking_simpson.txt", stringsAsFactors=TRUE)
str(s)
```

```
## 'data.frame':    12 obs. of  4 variables:
## $ V1: Factor w/ 3 levels "18-44","45-64",...: 1 1 1 1 2 2 2 2 3 3 ...
## $ V2: Factor w/ 2 levels "o","x": 1 1 2 2 1 1 2 2 1 1 ...
## $ V3: Factor w/ 2 levels "no","yes": 1 2 1 2 1 2 1 2 1 2 ...
## $ V4: int   19 269 13 327 78 167 52 147 42 7 ...
```

s

```
##      V1 V2  V3  V4
## 1 18-44 o  no  19
## 2 18-44 o yes 269
## 3 18-44 x  no  13
## 4 18-44 x yes 327
## 5 45-64 o  no  78
## 6 45-64 o yes 167
## 7 45-64 x  no  52
## 8 45-64 x yes 147
## 9 65-   o  no  42
## 10 65-   o yes   7
## 11 65-   x  no 165
## 12 65-   x yes  28
```

변수들에 이름을 주고,

```
names(s)<-c("Age", "Smoking", "Survived", "Counts")
s
```

```
##      Age Smoking Survived Counts
## 1 18-44      o      no      19
## 2 18-44      o     yes     269
## 3 18-44      x      no      13
## 4 18-44      x     yes     327
## 5 45-64      o      no      78
## 6 45-64      o     yes     167
## 7 45-64      x      no      52
## 8 45-64      x     yes     147
## 9 65-       o      no      42
## 10 65-       o     yes       7
## 11 65-       x      no     165
## 12 65-       x     yes      28
```

사실상 ordered factor인 "Age" 를 제대로 규정해 주고,

```
s$Age<-factor(s$Age, ordered=TRUE)
str(s)
```

```
## 'data.frame':    12 obs. of  4 variables:
##  $ Age      : Ord.factor w/ 3 levels "18-44"<"45-64"<...: 1 1 1 1 2 2 2 2 3 3
## ...
##  $ Smoking  : Factor w/ 2 levels "o","x": 1 1 2 2 1 1 2 2 1 1 ...
##  $ Survived: Factor w/ 2 levels "no","yes": 1 2 1 2 1 2 1 2 1 2 ...
##  $ Counts   : int   19 269 13 327 78 167 52 147 42 7 ...
```

연령대별로 흡연여부와 생존여부를 살펴본다.

```
xtabs(Counts~Survived+Smoking+Age,data=s)
```

```
## , , Age = 18-44
##
##      Smoking
## Survived  o   x
##      no   19  13
##      yes 269 327
##
## , , Age = 45-64
##
##      Smoking
## Survived  o   x
##      no   78  52
##      yes 167 147
##
## , , Age = 65-
##
##      Smoking
## Survived  o   x
##      no   42 165
##      yes   7  28
```

연령대를 고려하지 않고 집계하면,

```
xtabs(Counts~Survived+Smoking,data=s)
```

```
##      Smoking
## Survived  o   x
##      no  139 230
##      yes 443 502
```

흡연여부와 생존률의 관계를 살피기 어려우므로,

```
options("digits"=2)
prop.table(xtabs(Counts~Survived+Smoking, data=s), margin=2)
```

```
##           Smoking
## Survived      o      x
##           no  0.24 0.31
##           yes 0.76 0.69
```

놀랍게도 흡연자들의 생존률이 비흡연자들의 생존률보다 높게 나타나고 있다. 그러나 연령대별로 나눠보면,

```
prop.table(xtabs(Counts~Survived+Smoking+Age, data=s), margin=c(2, 3))
```

```
## , , Age = 18-44
##
##           Smoking
## Survived      o      x
##           no  0.066 0.038
##           yes 0.934 0.962
##
## , , Age = 45-64
##
##           Smoking
## Survived      o      x
##           no  0.318 0.261
##           yes 0.682 0.739
##
## , , Age = 65-
##
##           Smoking
## Survived      o      x
##           no  0.857 0.855
##           yes 0.143 0.145
```

어느 연령대에서나 비흡연자의 생존률이 높게 나와서 이 또한 전형적인 Simpson's Paradox에 해당함을 알 수 있다. 생존률만 일목요연하게 비교할 수 있으려면, `prop.table` 을 이용할 수 있는데,

```
prop.table(xtabs(Counts~Survived+Smoking+Age, data=s), margin=c(2, 3))[2, , ]
```

```
##           Age
## Smoking 18-44 45-64 65-
##           o  0.93  0.68  0.14
##           x  0.96  0.74  0.15
```

`fTable` 을 사용할 경우

```
fTable(prop.table(xtabs(Counts~Survived+Smoking+Age, data=s), margin=c(2, 3)))
```



```
##           Age 18-44 45-64 65-
## Survived Smoking
## no      o          0.066 0.318 0.857
##         x          0.038 0.261 0.855
## yes     o          0.934 0.682 0.143
##         x          0.962 0.739 0.145
```

전체를 살펴보는 데는 문제가 없지만, 일부분을 추출하면

```
fable(prop.table(xtabs(Counts~Survived+Smoking+Age, data=s), margin=c(2, 3)))
[1:2, ]
```

```
##      [,1] [,2] [,3]
## [1,] 0.066 0.32 0.86
## [2,] 0.038 0.26 0.85
```

이름을 잃어버린다. 이 자료도 glm 을 사용하여 분석할 수 있다.

```
Not.Survived<-s$Survived=="no"
s.2<-data.frame(s[Not.Survived,1:2], Not.Survived=s[Not.Survived, 4], Survive
d=s[!Not.Survived, 4])
s.2
```

```
##      Age Smoking Not.Survived Survived
## 1  18-44      o             19      269
## 3  18-44      x             13      327
## 5  45-64      o             78      167
## 7  45-64      x             52      147
## 9  65-      o             42       7
## 11 65-      x            165      28
```

binomial family 에 logit 을 link로 하는 glm 에 적합시키자. 먼저, 흡연여부에 대해서만 파악해 보면

```
s.glm.3<-glm(cbind(Survived, Not.Survived)~Smoking, family=binomial(logit), dat
a=s.2)
summary(s.glm.3)
```

```
##
## Call:
## glm(formula = cbind(Survived, Not.Survived) ~ Smoking, family = binomial(log
it),
##     data = s.2)
##
## Deviance Residuals:
##      1      3      5      7      9     11
##  7.82  12.90  -2.83   1.64  -9.16 -15.60
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   1.1591     0.0972   11.92  <2e-16 ***
## Smokingx     -0.3786     0.1257   -3.01   0.0026 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 574.73  on 5  degrees of freedom
## Residual deviance: 565.53  on 4  degrees of freedom
## AIC: 598.6
##
## Number of Fisher Scoring iterations: 4
```

회귀계수를 다시 살펴보면,

```
coef(s.glm.3)
```

```
## (Intercept)      Smokingx
##           1.16        -0.38
```

비흡연자(Smokingx)의 생존률 log(odds ratio)가 더 낮게 나오는 것을 확인할 수 있다. 보다 구체적으로 `exp()` 를 취해 보면,

```
exp(coef(s.glm.3))
```

```
## (Intercept)      Smokingx
##           3.19         0.68
```

이는 앞에서 계산한,

```
s.rates.1<-prop.table(xtabs(Counts~Survived+Smoking, data=s), margin=2)
s.rates.1
```

```
##           Smoking
## Survived    o    x
##      no  0.24 0.31
##      yes 0.76 0.69
```

로부터

```
(s.rates.1[2,2]/s.rates.1[1,2])/(s.rates.1[2,1]/s.rates.1[1,1])
```

```
## [1] 0.68
```

와 일치한다. 연령대를 고려하면,

```
s.glm.4<-glm(cbind(Survived, Not.Survived)~Smoking+Age, family=binomial(logit),
data=s.2)
summary(s.glm.4)
```

```
##
## Call:
## glm(formula = cbind(Survived, Not.Survived) ~ Smoking + Age,
##      family = binomial(logit), data = s.2)
##
## Deviance Residuals:
##      1      3      5      7      9     11
## -0.4881  0.5368  0.0934 -0.1104  0.5643 -0.2635
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    0.496     0.135    3.67  0.00024 ***
## Smokingx       0.309     0.172    1.80  0.07210 .
## Age.L        -3.392     0.187   -18.10 < 2e-16 ***
## Age.Q        -0.309     0.139    -2.22  0.02656 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 574.73158  on 5  degrees of freedom
## Residual deviance:  0.93516  on 2  degrees of freedom
## AIC: 38.01
##
## Number of Fisher Scoring iterations: 4
```

흡연여부는 더 이상 유의하지 않고, 연령대가 유력한 요인으로 등장한다. 회귀계수의 분석은 생략하고, anova 로 분석하면,

```
anova(s.glm.3, s.glm.4, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: cbind(Survived, Not.Survived) ~ Smoking
## Model 2: cbind(Survived, Not.Survived) ~ Smoking + Age
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1          4          566
## 2          2          1  2          565    <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

연령대가 유의한 요인임을 확인한다.

Titanic Survival Rates in 3rd Class and Crew

```
str(Titanic)
```

```
## table [1:4, 1:2, 1:2, 1:2] 0 0 35 0 0 0 17 0 118 154 ...
## - attr(*, "dimnames")=List of 4
##   ..$ Class      : chr [1:4] "1st" "2nd" "3rd" "Crew"
##   ..$ Sex        : chr [1:2] "Male" "Female"
##   ..$ Age        : chr [1:2] "Child" "Adult"
##   ..$ Survived: chr [1:2] "No" "Yes"
```

```
apply(Titanic, c(1,2), sum)
```

```
##           Sex
## Class  Male Female
## 1st    180    145
## 2nd    179    106
## 3rd    510    196
## Crew   862     23
```

```
apply(Titanic, c(1,2), sum)[3:4,]
```

```
##           Sex
## Class  Male Female
## 3rd    510    196
## Crew   862     23
```

```
apply(Titanic, c(1,4), sum)[3:4,]
```

```
##           Survived
## Class  No Yes
## 3rd    528 178
## Crew   673 212
```

```
options(digits=2)
apply(Titanic, c(1,2,4), sum)
```

```
## , , Survived = No
##
##      Sex
## Class Male Female
## 1st    118      4
## 2nd    154     13
## 3rd    422    106
## Crew   670      3
##
## , , Survived = Yes
##
##      Sex
## Class Male Female
## 1st     62    141
## 2nd     25     93
## 3rd     88     90
## Crew   192     20
```

```
apply(Titanic, c(1,2,4), sum)[3:4,,]
```

```
## , , Survived = No
##
##      Sex
## Class Male Female
## 3rd    422    106
## Crew   670      3
##
## , , Survived = Yes
##
##      Sex
## Class Male Female
## 3rd     88     90
## Crew   192     20
```

```
ftable(apply(Titanic, c(1,2,4), sum)[3:4,,])
```

```
##           Survived  No Yes
## Class Sex
## 3rd  Male           422  88
##      Female         106  90
## Crew Male           670 192
##      Female           3  20
```

```
ftable(apply(Titanic, c(1,4,2), sum)[3:4,,])
```

```
##                Sex Male Female
## Class Survived
## 3rd    No           422      106
##        Yes           88       90
## Crew   No           670        3
##        Yes          192       20
```

```
fable(apply(Titanic, c(1,4,2), sum)[3:4,,])[1:2,]
```

```
##      [,1] [,2]
## [1,]  422  106
## [2,]   88   90
```

```
fable(apply(Titanic, c(1,4,2), sum)[3:4,,])[3:4,]
```

```
##      [,1] [,2]
## [1,]  670    3
## [2,]  192   20
```

```
prop.table(fable(apply(Titanic, c(1,4,2), sum)[3:4,,])[1:2,], margin=2)
```

```
##      [,1] [,2]
## [1,] 0.83 0.54
## [2,] 0.17 0.46
```

```
prop.table(fable(apply(Titanic, c(1,4,2), sum)[3:4,,])[3:4,], margin=2)
```

```
##      [,1] [,2]
## [1,] 0.78 0.13
## [2,] 0.22 0.87
```

```
matrix(prop.table(fable(apply(Titanic, c(1,4,2), sum)[3:4,,])[1:2,], margin=2), nrow=2, dimnames=list(dimnames(Titanic)$Survived, dimnames(Titanic)$Sex))
```

```
##      Male Female
## No   0.83    0.54
## Yes  0.17    0.46
```

```
matrix(prop.table(fable(apply(Titanic, c(1,4,2), sum)[3:4,,])[3:4,], margin=2), nrow=2, dimnames=list(dimnames(Titanic)$Survived, dimnames(Titanic)$Sex))
```

##		Male	Female
##	No	0.78	0.13
##	Yes	0.22	0.87