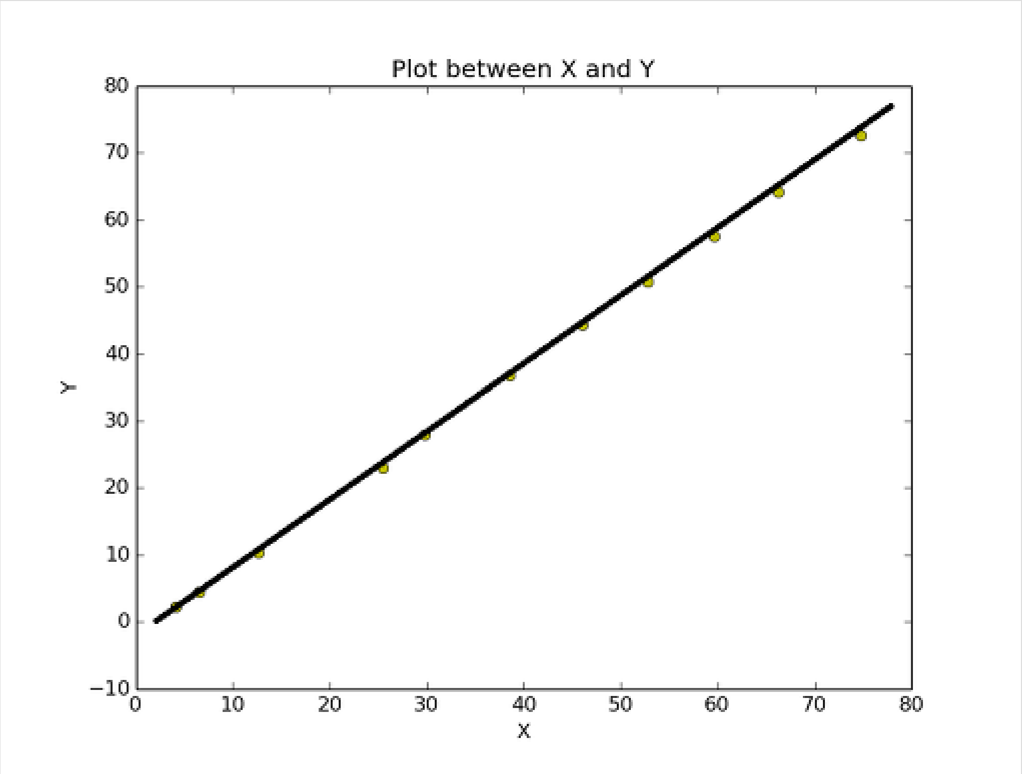
Introductory Statistics with R
(Chap. 13~14)

Seungyeon Seo

Chap.13 Logistic Regression

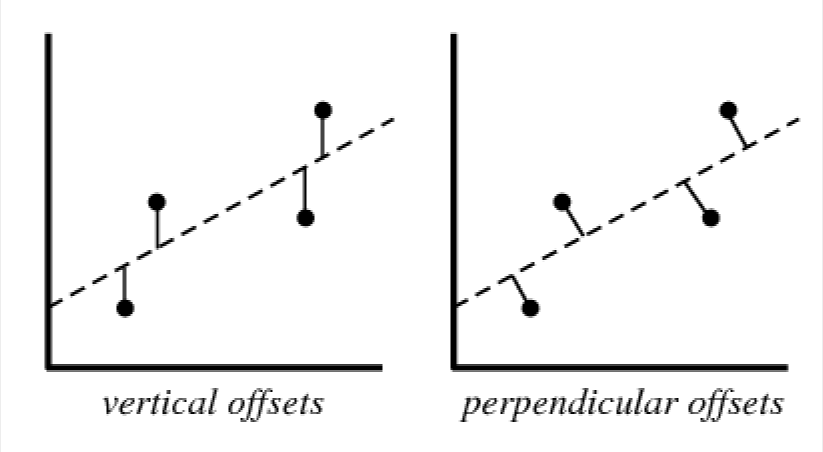
Linear Regression Analysis



모든 data들에 대해서 y = a + bx라는 식

Residual (error)
 = real y data - expected y data
 = real y data - (a + bx)
(because, expected y data = a + bx)

R^2 = (real y data - expected y data)^2



http://operatingsystems.tistory.com/entry/Data-Mining-Linear-Regression
http://mathworld.wolfram.com/LeastSquaresFitting.html

Introduction

❖ Data

Binary outcomes (only two possible values);
Diseased / Nondiseased or 0 / 1

❖ Purpose

Dose-response relationships
The effect of multiple variables simultaneously

❖ Limitations

A limited range
Regression models – predicting off-scale values below zero or above

❖ Solutions

The probabilities on a transformed scale

▼

Logistic regression analysis

❖ Alternatives

Due to mathematically convenient properties

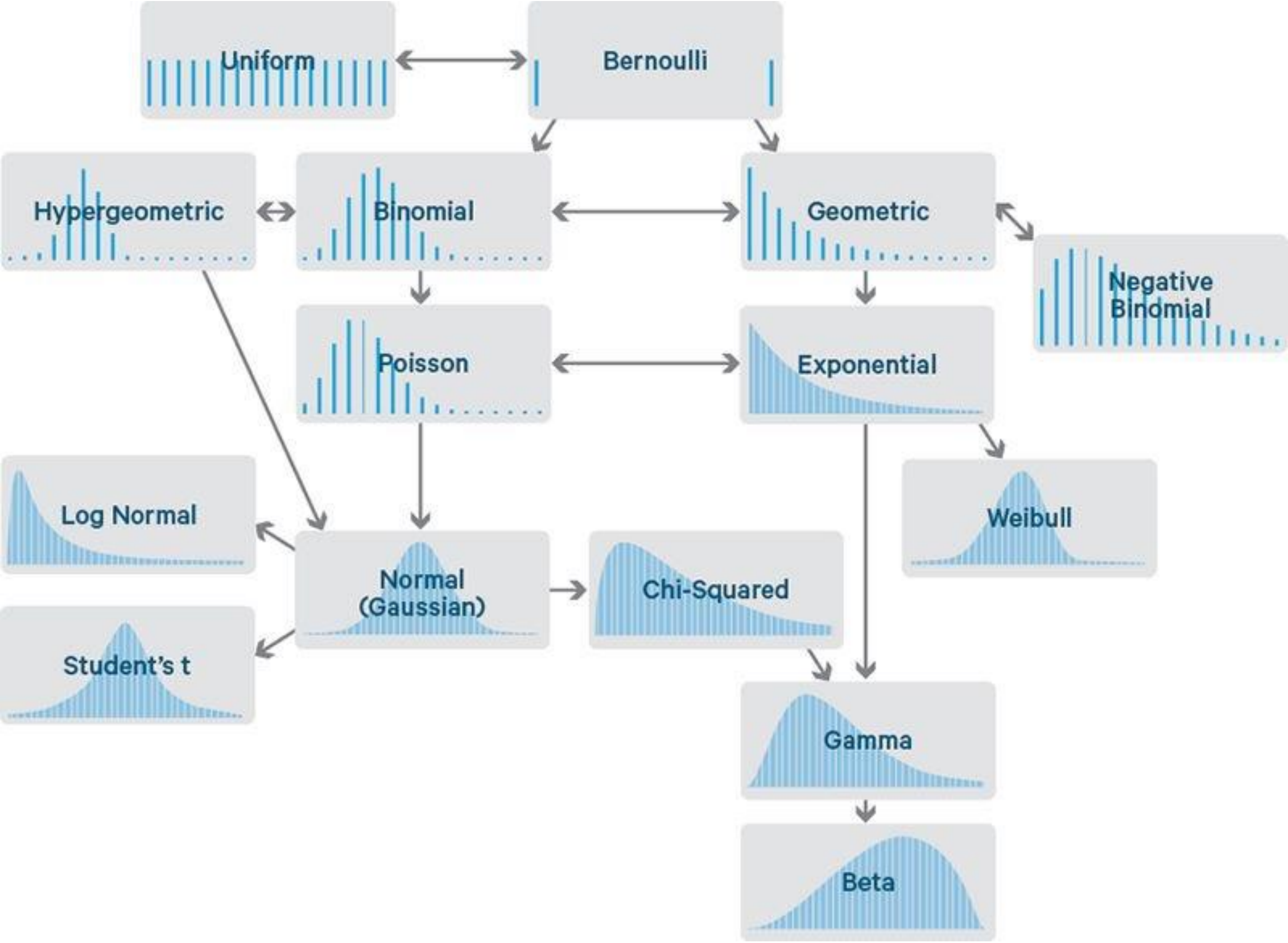
Probit function
Log (-log p) – survival analysis models

❖ Logistic regression analysis

1. Response distribution (the binomial distribution)
2. *Link function* (logit p = log [p/ (1 − p)])

❖ Multiplicative Poission model

1. Link function (logit λ, λ is the mean of the Poisson-distributed observation)


Uniform
Bernoulli
Hypergeometric
Binomial
Geometric
Negative Binomial
Poisson
Exponential
Log Normal
Weibull
Normal (Gaussian)
Chi-Squared
Student's t
Gamma
Beta

Logistic Regression on Tabular Data

1. Response distribution (the binomial distribution)
2. *Link function* (logit p = log [p/ (1 − p)]])

logit $p = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + . . . \beta_k x_k$

logit p = log [p/ (1 − p)] -> *log odds*

1. No error term as in linear models
2. No variance parameter as in the normal distribution

1.
The method of maximum likelihood

▼

The least-squares method

Conclusion;
The likelihood function $L (\beta)$

2.
The difference between the maximized value of −2 log L and the similar quantity under a "maximal model"

glm function

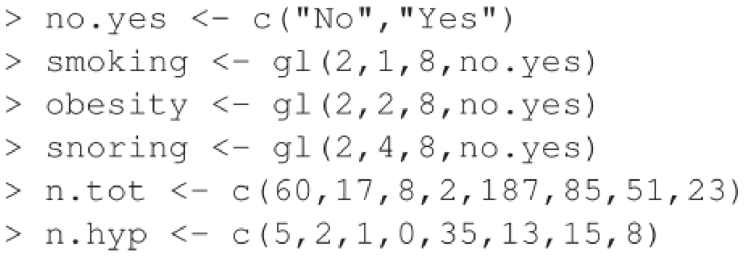
lm function (**L**inear normal **M**odels)

1. The same model formulas
2. Extractor function (summary)

+

3. Family argument (family=binomial("logit"))

Logistic Regression on Tabular Data

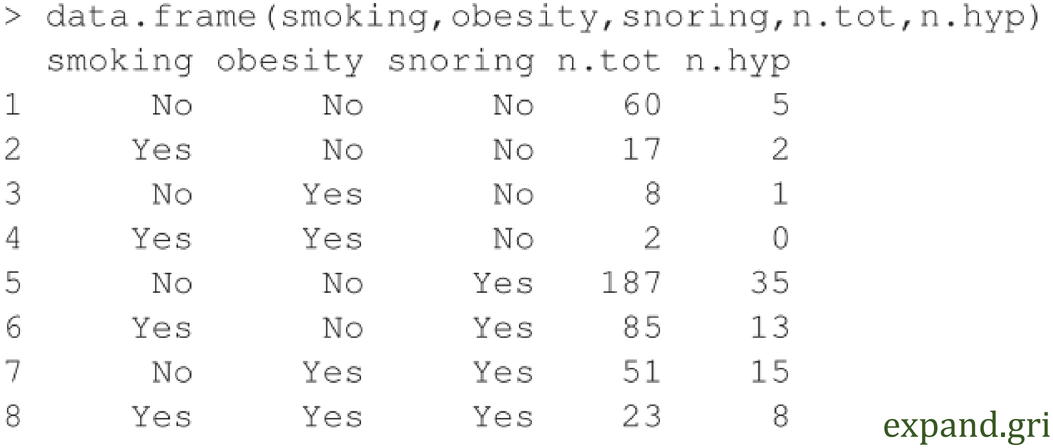


```
> no.yes <- c("No","Yes")
> smoking <- gl(2,1,8,no.yes)
> obesity <- gl(2,2,8,no.yes)
> snoring <- gl(2,4,8,no.yes)
> n.tot <- c(60,17,8,2,187,85,51,23)
> n.hyp <- c(5,2,1,0,35,13,15,8)
```

<- 으로 variables
no.yes 로 2개만 있는 것 확인

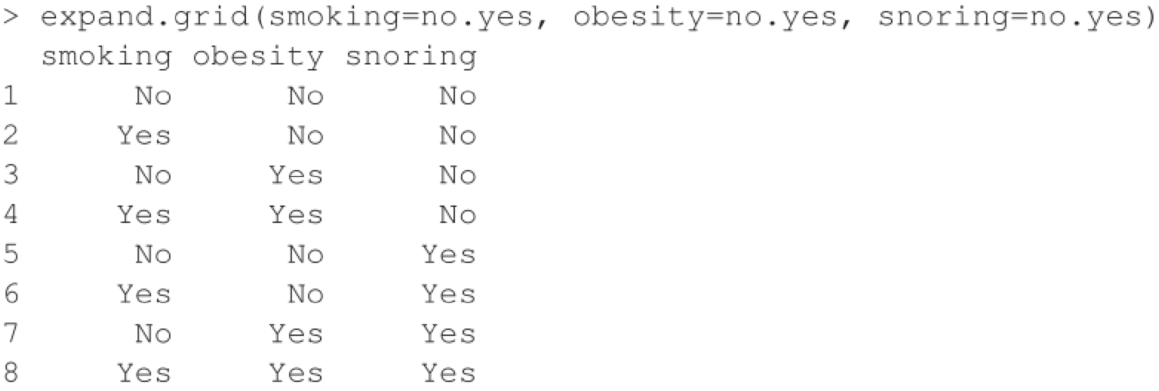
gl function (**G**enerate **L**evels)
Number of levels
Repeat count of each level
Total length of the vector

data.frame function



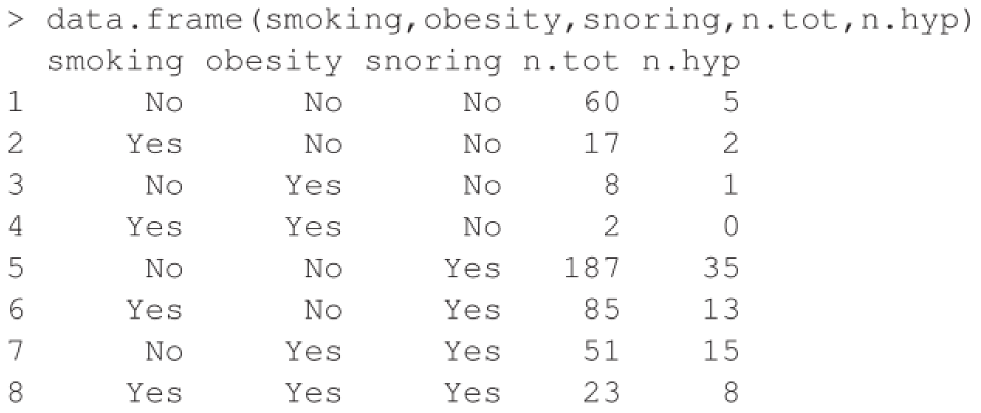
```
> data.frame(smoking,obesity,snoring,n.tot,n.hyp)
  smoking obesity snoring n.tot n.hyp
1      No      No      No    60     5
2     Yes      No      No    17     2
3      No     Yes      No     8     1
4     Yes     Yes      No     2     0
5      No      No     Yes   187    35
6     Yes      No     Yes    85    13
7      No     Yes     Yes    51    15
8     Yes     Yes     Yes    23     8
```

expand.grid function

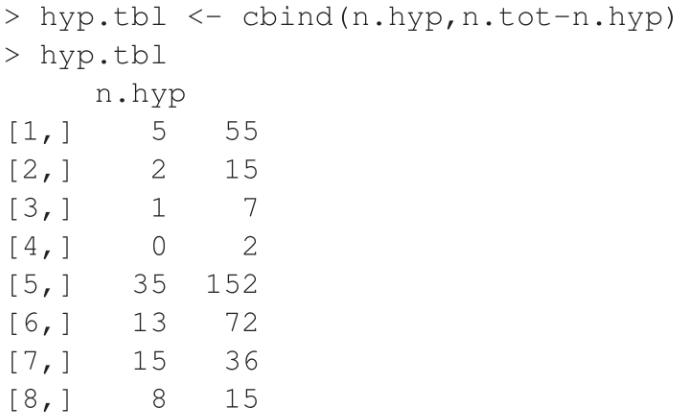


```
> expand.grid(smoking=no.yes, obesity=no.yes, snoring=no.yes)
  smoking obesity snoring
1      No      No      No
2     Yes      No      No
3      No     Yes      No
4     Yes     Yes      No
5      No      No     Yes
6     Yes      No     Yes
7      No     Yes     Yes
8     Yes     Yes     Yes
```

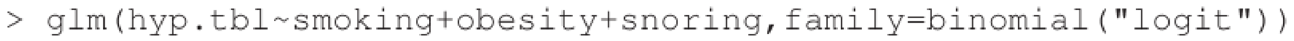
Logistic Regression_Practice



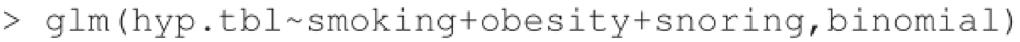
```
> data.frame(smoking,obesity,snoring,n.tot,n.hyp)
  smoking obesity snoring n.tot n.hyp
1      No      No      No    60     5
2     Yes      No      No    17     2
3      No     Yes      No     8     1
4     Yes     Yes      No     2     0
5      No      No     Yes   187    35
6     Yes      No     Yes    85    13
7      No     Yes     Yes    51    15
8     Yes     Yes     Yes    23     8
```



```
> hyp.tbl <- cbind(n.hyp,n.tot-n.hyp)
> hyp.tbl
     n.hyp
[1,]     5  55
[2,]     2  15
[3,]     1   7
[4,]     0   2
[5,]    35 152
[6,]    13  72
[7,]    15  36
[8,]     8  15
```

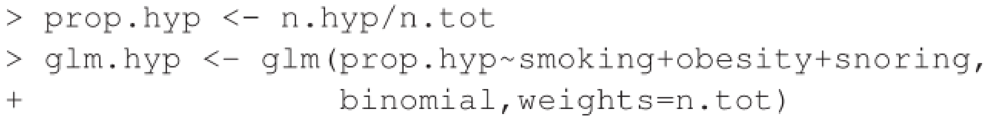


```
> glm(hyp.tbl~smoking+obesity+snoring,family=binomial("logit"))
```

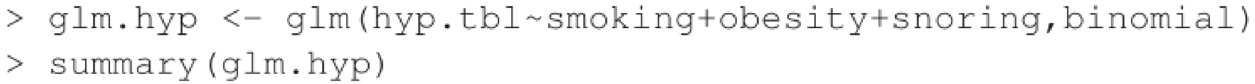


```
> glm(hyp.tbl~smoking+obesity+snoring,binomial)
```

glm function
summary ()

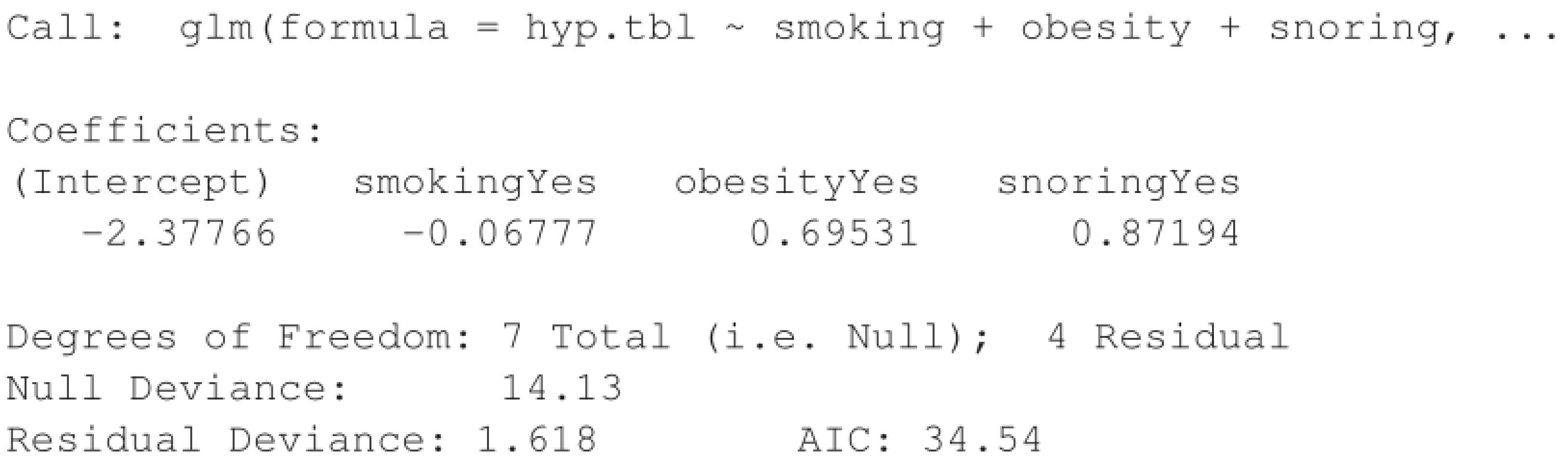


```
> prop.hyp <- n.hyp/n.tot
> glm.hyp <- glm(prop.hyp~smoking+obesity+snoring,
+                binomial,weights=n.tot)
```



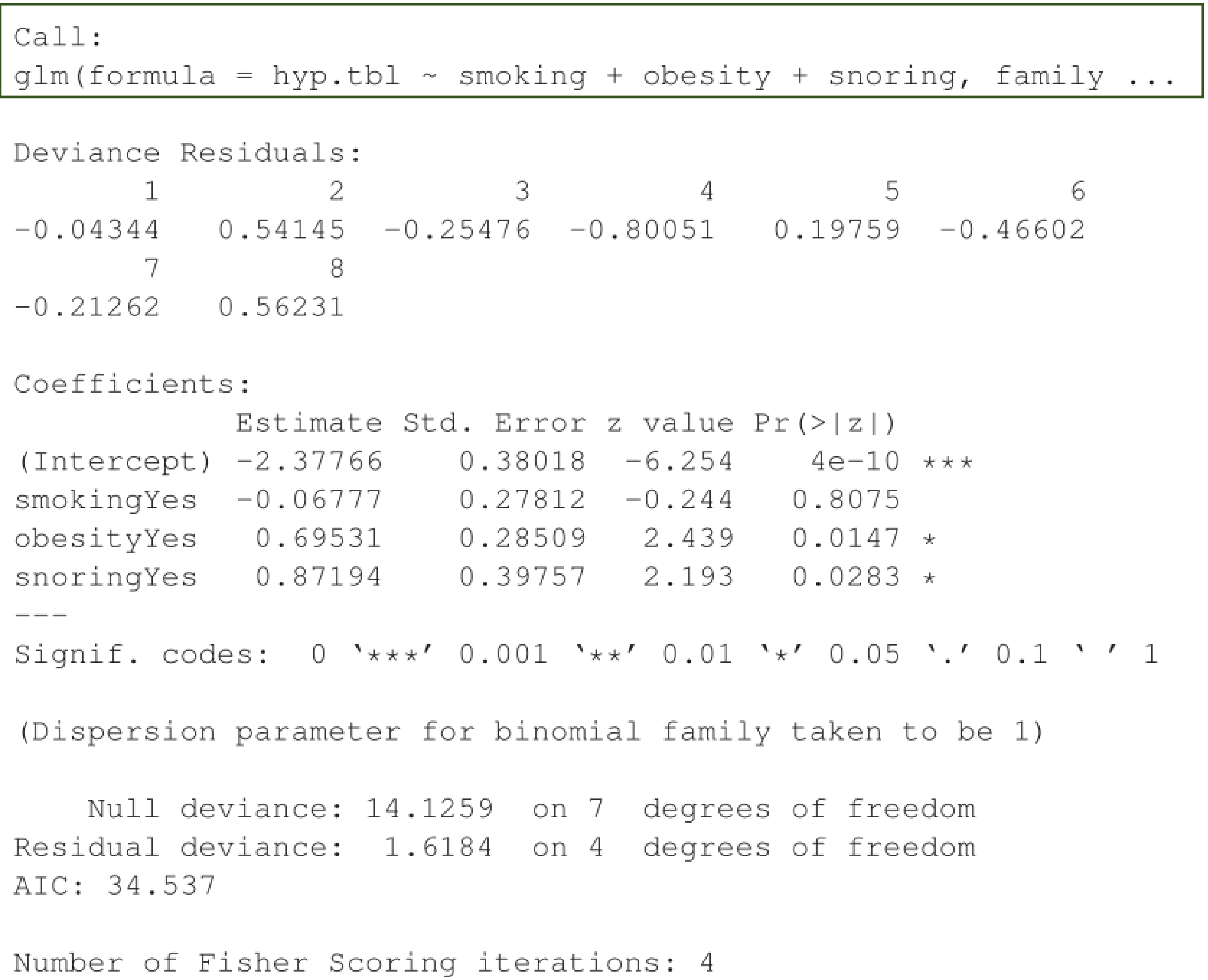
```
> glm.hyp <- glm(hyp.tbl~smoking+obesity+snoring,binomial)
> summary(glm.hyp)
```

Logistic Regression_Practice



```
Call:  glm(formula = hyp.tbl ~ smoking + obesity + snoring, ...

Coefficients:
(Intercept)   smokingYes   obesityYes   snoringYes
   -2.37766     -0.06777      0.69531      0.87194

Degrees of Freedom: 7 Total (i.e. Null);  4 Residual
Null Deviance:      14.13
Residual Deviance: 1.618        AIC: 34.54
```

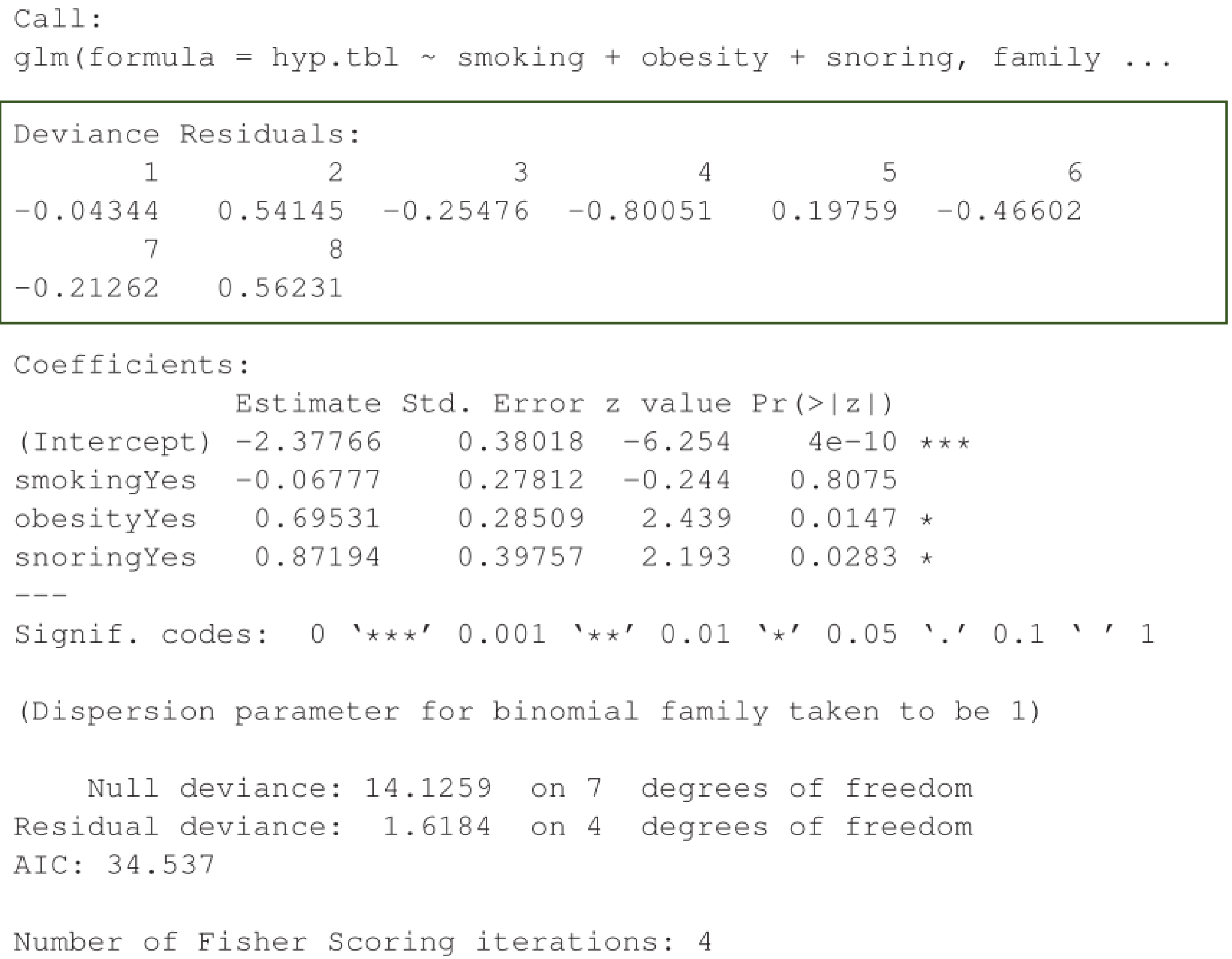
Logistic Regression_Practice



```
Call:
glm(formula = hyp.tbl ~ smoking + obesity + snoring, family ...

Deviance Residuals:
       1          2          3          4          5          6  
-0.04344   0.54145  -0.25476  -0.80051   0.19759  -0.46602  
       7          8  
-0.21262   0.56231  

Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) -2.37766    0.38018  -6.254    4e-10 ***
smokingYes  -0.06777    0.27812  -0.244   0.8075    
obesityYes   0.69531    0.28509   2.439   0.0147 *  
snoringYes   0.87194    0.39757   2.193   0.0283 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14.1259  on 7  degrees of freedom
Residual deviance:  1.6184  on 4  degrees of freedom
AIC: 34.537

Number of Fisher Scoring iterations: 4
```

❖ Repeat of the model specification

Logistic Regression_Practice

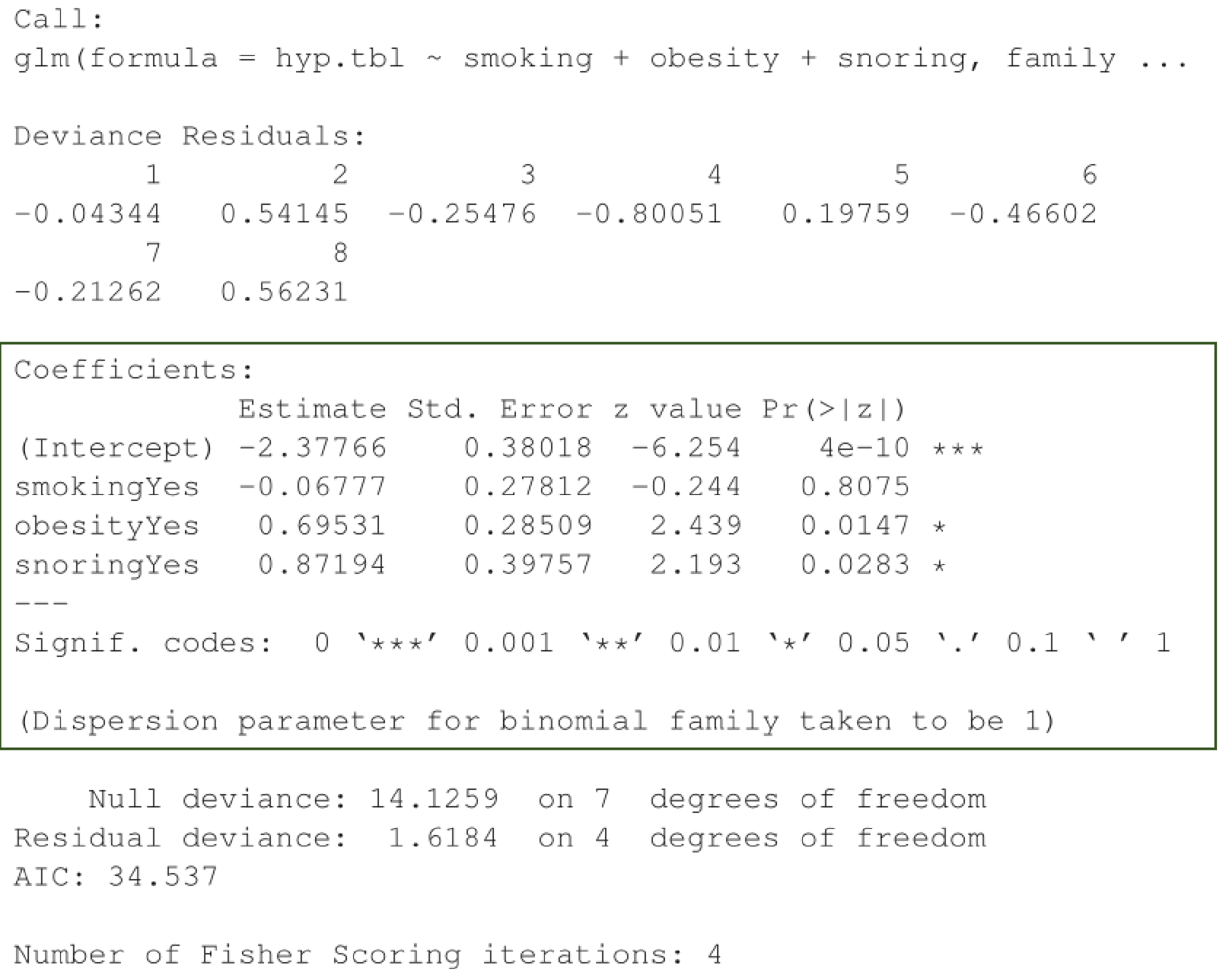


```
Call:
glm(formula = hyp.tbl ~ smoking + obesity + snoring, family ...

Deviance Residuals:
       1         2         3         4         5         6
-0.04344   0.54145  -0.25476  -0.80051   0.19759  -0.46602
       7         8
-0.21262   0.56231

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.37766    0.38018  -6.254    4e-10 ***
smokingYes  -0.06777    0.27812  -0.244   0.8075
obesityYes   0.69531    0.28509   2.439   0.0147 *
snoringYes   0.87194    0.39757   2.193   0.0283 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14.1259  on 7  degrees of freedom
Residual deviance:  1.6184  on 4  degrees of freedom
AIC: 34.537

Number of Fisher Scoring iterations: 4
```

- Contribution of each cell of the table to the deviance of the model (1부터 8까지 각각의 수치가 전체 deviance에 기여하는 정도를 나타냄)

- Corresponding the sum of squares in linear normal models

Logistic Regression_Practice

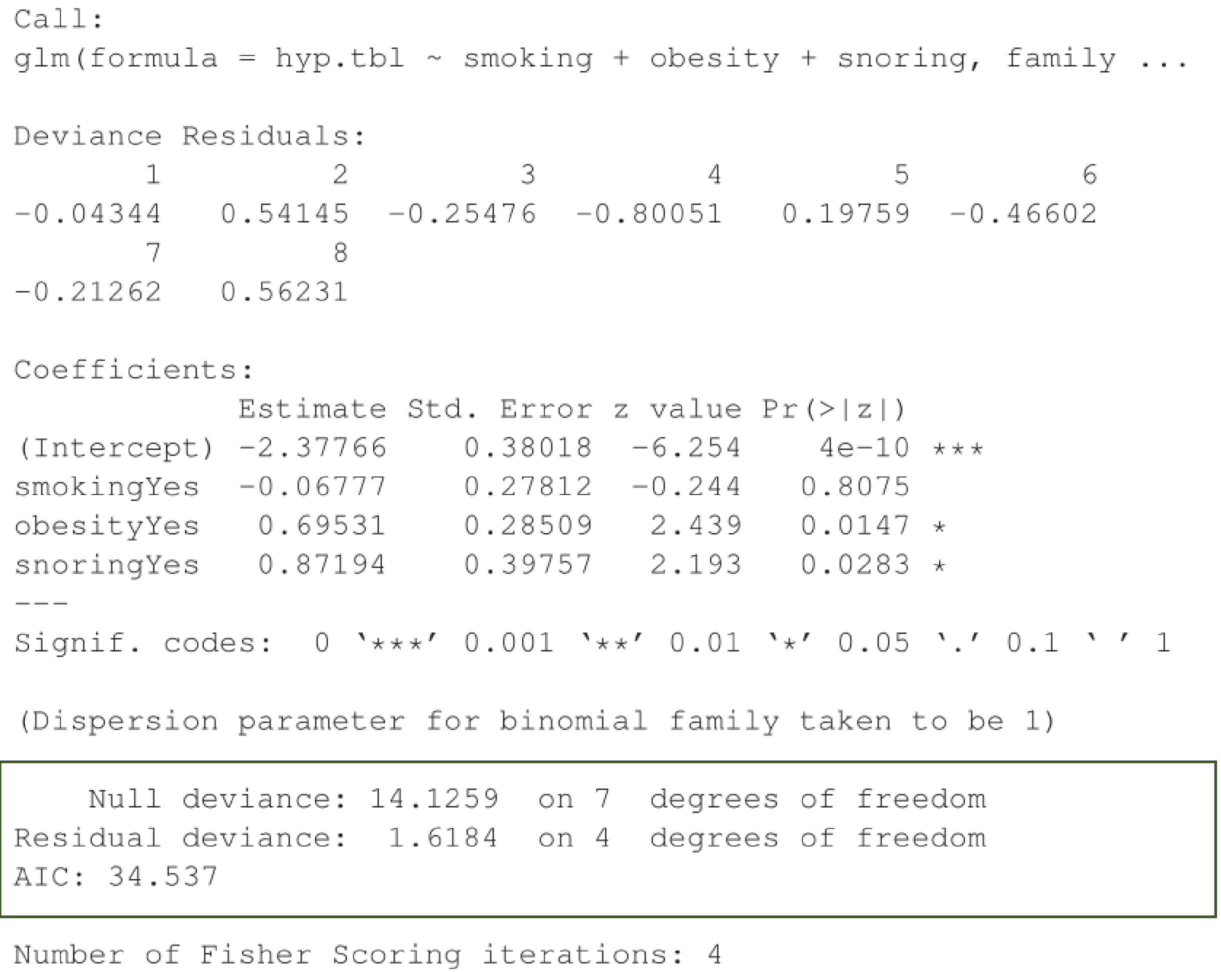


```
Call:
glm(formula = hyp.tbl ~ smoking + obesity + snoring, family ...

Deviance Residuals:
       1         2         3         4         5         6
-0.04344   0.54145  -0.25476  -0.80051   0.19759  -0.46602
       7         8
-0.21262   0.56231

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.37766    0.38018  -6.254    4e-10 ***
smokingYes  -0.06777    0.27812  -0.244   0.8075
obesityYes   0.69531    0.28509   2.439   0.0147 *
snoringYes   0.87194    0.39757   2.193   0.0283 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14.1259  on 7  degrees of freedom
Residual deviance:  1.6184  on 4  degrees of freedom
AIC: 34.537

Number of Fisher Scoring iterations: 4
```

- ❖ Estimates of the regression coefficients
- ❖ Standard errors of same
- ❖ Tests for whether each regression coefficient can be assumed to be zero

- ❖ Corresponding part of lm output

Logistic Regression_Practice



```
Call:
glm(formula = hyp.tbl ~ smoking + obesity + snoring, family ...

Deviance Residuals:
       1         2         3         4         5         6
-0.04344   0.54145  -0.25476  -0.80051   0.19759  -0.46602
       7         8
-0.21262   0.56231

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.37766    0.38018  -6.254    4e-10 ***
smokingYes  -0.06777    0.27812  -0.244   0.8075
obesityYes   0.69531    0.28509   2.439   0.0147 *
snoringYes   0.87194    0.39757   2.193   0.0283 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14.1259  on 7  degrees of freedom
Residual deviance:  1.6184  on 4  degrees of freedom
AIC: 34.537

Number of Fisher Scoring iterations: 4
```

Residual deviance

- The residual sum of squares (ordinary regression analyses)

▼

- The standard deviation

- The standard deviation of the observations is known. (in binomial models)

▼

- AIC (Akaike information criterion)
- A measure of goodness of fit

Null deviance

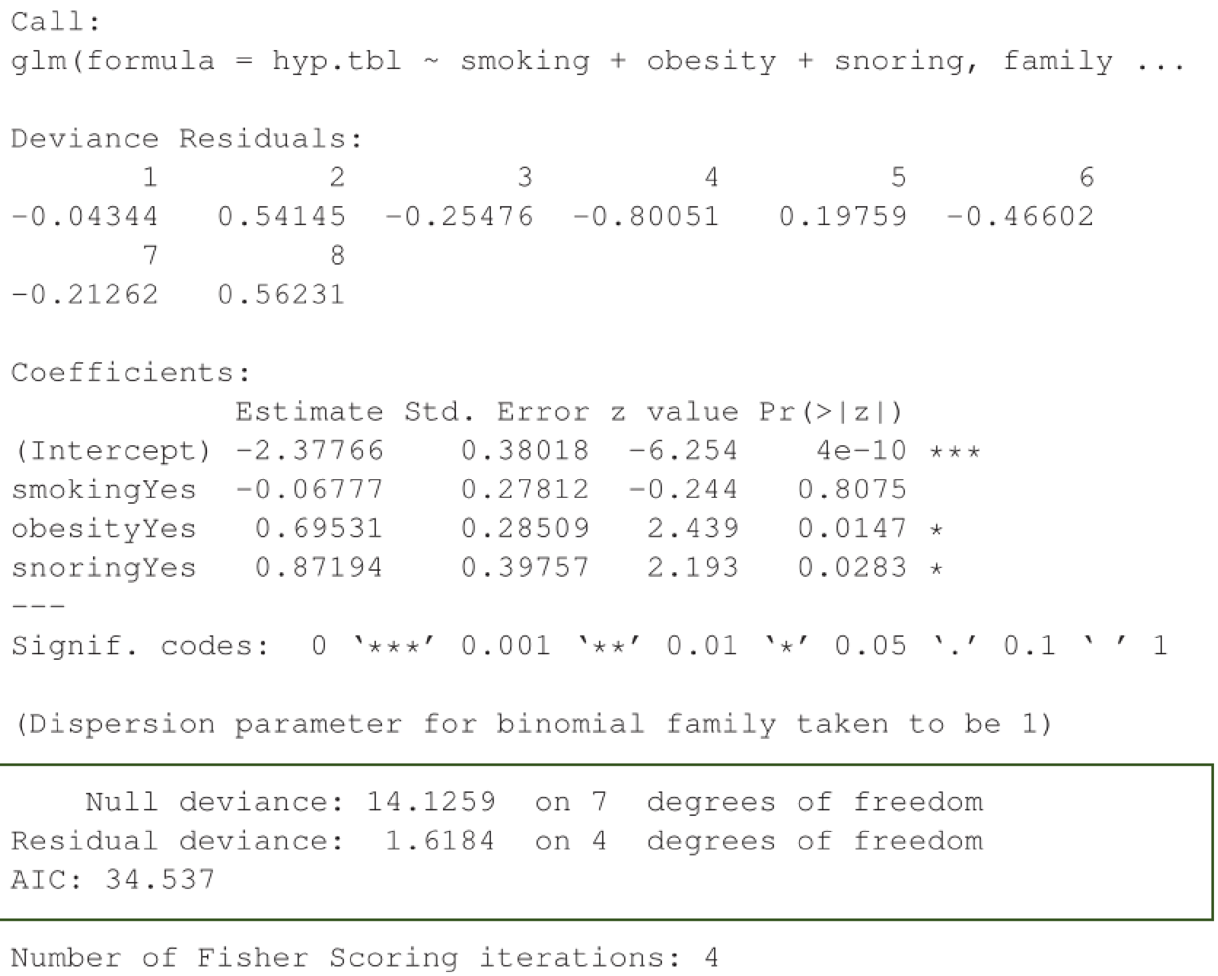
- The deviance of a model that contains only the intercept

- The difference from the residual deviance
- Used for a joint test for whether any effects are present in the model

$$14.13 - 1.62 = 12.51$$

- *P*-value of approximately 0.6%

Logistic Regression_Practice



```
Call:
glm(formula = hyp.tbl ~ smoking + obesity + snoring, family ...

Deviance Residuals:
       1          2          3          4          5          6  
-0.04344   0.54145  -0.25476  -0.80051   0.19759  -0.46602  
       7          8  
-0.21262   0.56231  

Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) -2.37766    0.38018  -6.254    4e-10 ***
smokingYes  -0.06777    0.27812  -0.244   0.8075    
obesityYes   0.69531    0.28509   2.439   0.0147 *  
snoringYes   0.87194    0.39757   2.193   0.0283 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14.1259  on 7  degrees of freedom
Residual deviance:  1.6184  on 4  degrees of freedom
AIC: 34.537

Number of Fisher Scoring iterations: 4
```

P-value

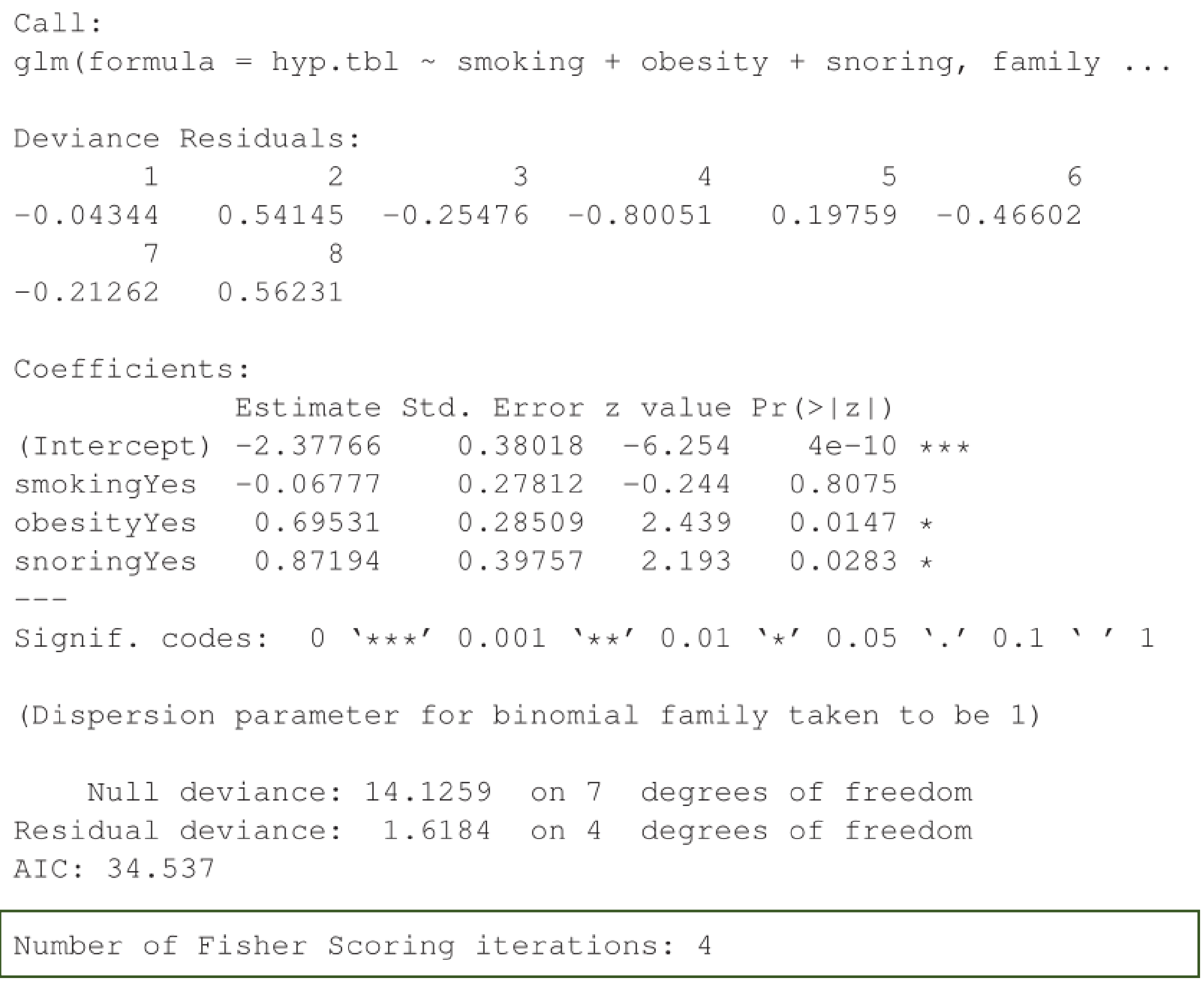
- ❖ No exact *p*-value
- ❖ Only an approximation

- ❖ The asymptotic distribution of the residual deviance

▼

- ❖ The model is wrong (?) -> nothing!

Logistic Regression_Practice



```
Call:
glm(formula = hyp.tbl ~ smoking + obesity + snoring, family ...

Deviance Residuals:
       1          2          3          4          5          6
-0.04344   0.54145  -0.25476  -0.80051   0.19759  -0.46602
       7          8
-0.21262   0.56231

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.37766    0.38018  -6.254    4e-10 ***
smokingYes  -0.06777    0.27812  -0.244   0.8075
obesityYes   0.69531    0.28509   2.439   0.0147 *
snoringYes   0.87194    0.39757   2.193   0.0283 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14.1259  on 7  degrees of freedom
Residual deviance:  1.6184  on 4  degrees of freedom
AIC: 34.537

Number of Fisher Scoring iterations: 4
```

- ❖ The actual fitting procedure
- ❖ Purely technical item
- ❖ No statistical information

- ❖ Too large -> too complex to fit (glm function -> halting the fitting procedure)

The Analysis of Deviance Table

- Corresponding ANOVA table for multiple regression analyses

- anova function

```
> glm.hyp <- glm(hyp.tbl~smoking+obesity+snoring,binomial)
> anova(glm.hyp, test="Chisq")
Analysis of Deviance Table

Model: binomial, link: logit

Response: hyp.tbl

Terms added sequentially (first to last)


        Df Deviance Resid. Df Resid. Dev P(>|Chi|)
NULL                        7    14.1259
smoking  1   0.0022         6    14.1237    0.9627
obesity  1   6.8274         5     7.2963    0.0090
snoring  1   5.6779         4     1.6184    0.0172
```

- Differences between models as variables
- χ 2 -distributed with the stated degrees of freedom

- 'snoring' variable -> significant
- 'smoking' variable -> not significant

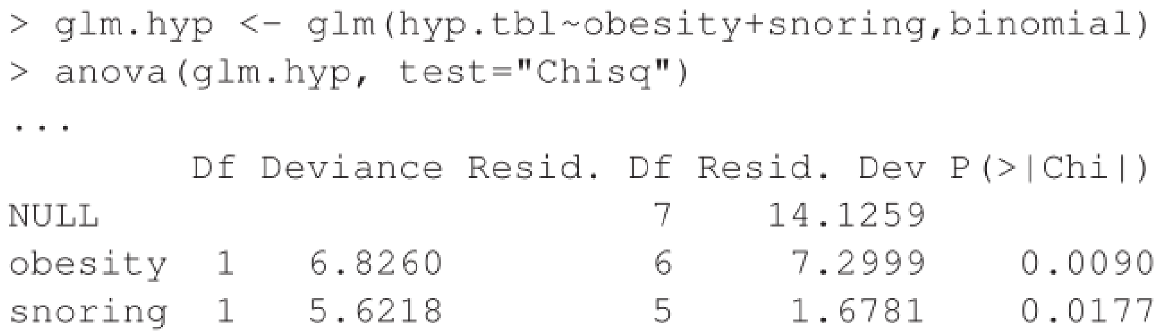
- Not be removed -> be rearranged

```
> glm.hyp <- glm(hyp.tbl~snoring+obesity+smoking,binomial)
> anova(glm.hyp, test="Chisq")
...
        Df Deviance Resid. Df Resid. Dev P(>|Chi|)
NULL                        7    14.1259
snoring  1   6.7887         6     7.3372    0.0092
obesity  1   5.6591         5     1.6781    0.0174
smoking  1   0.0597         4     1.6184    0.8069
```

- 'smoking' comes last -> removal

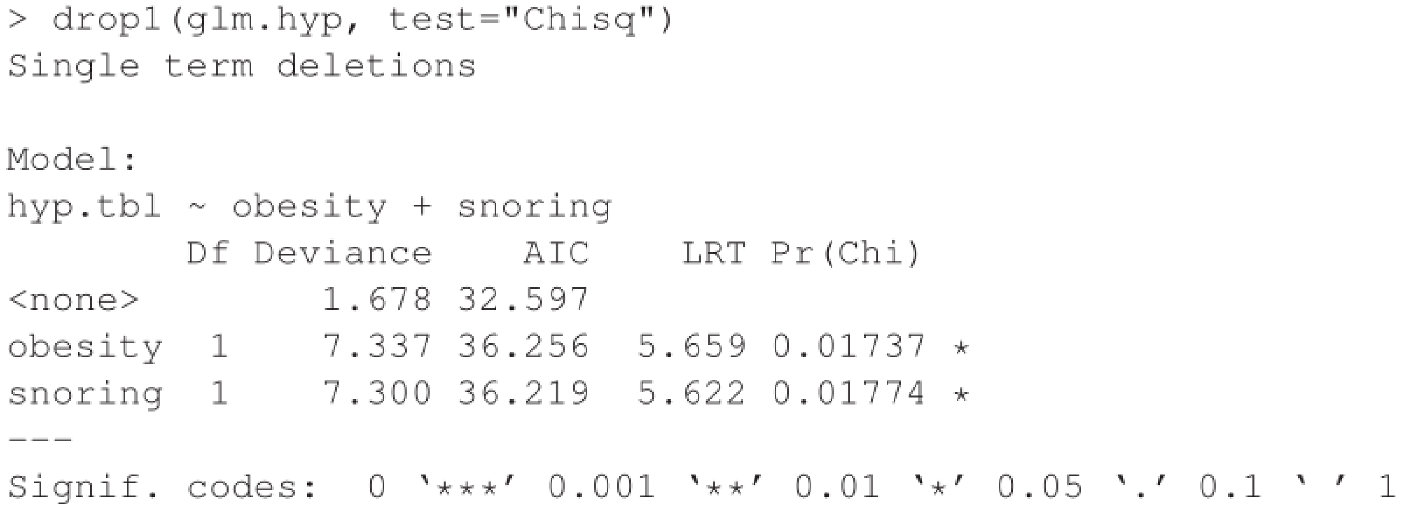
The Analysis of Deviance Table

- A test of whether snoring may be removed from a model that also contains obesity



```
> glm.hyp <- glm(hyp.tbl~obesity+snoring,binomial)
> anova(glm.hyp, test="Chisq")
...
         Df Deviance Resid. Df Resid. Dev P(>|Chi|)
NULL                         7    14.1259
obesity  1   6.8260          6     7.2999    0.0090
snoring  1   5.6218          5     1.6781    0.0177
```

- Alternative model
- drop1 function

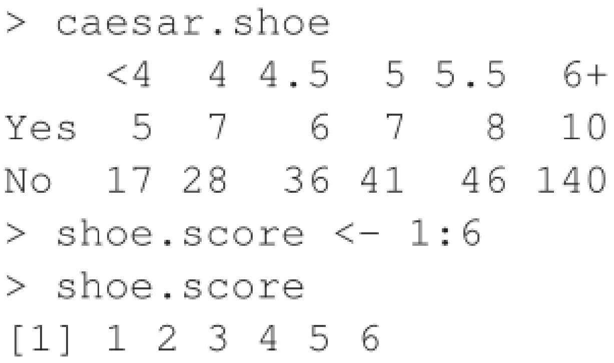


```
> drop1(glm.hyp, test="Chisq")
Single term deletions

Model:
hyp.tbl ~ obesity + snoring
        Df Deviance    AIC    LRT Pr(Chi)
<none>        1.678 32.597
obesity  1    7.337 36.256  5.659 0.01737 *
snoring  1    7.300 36.219  5.622 0.01774 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

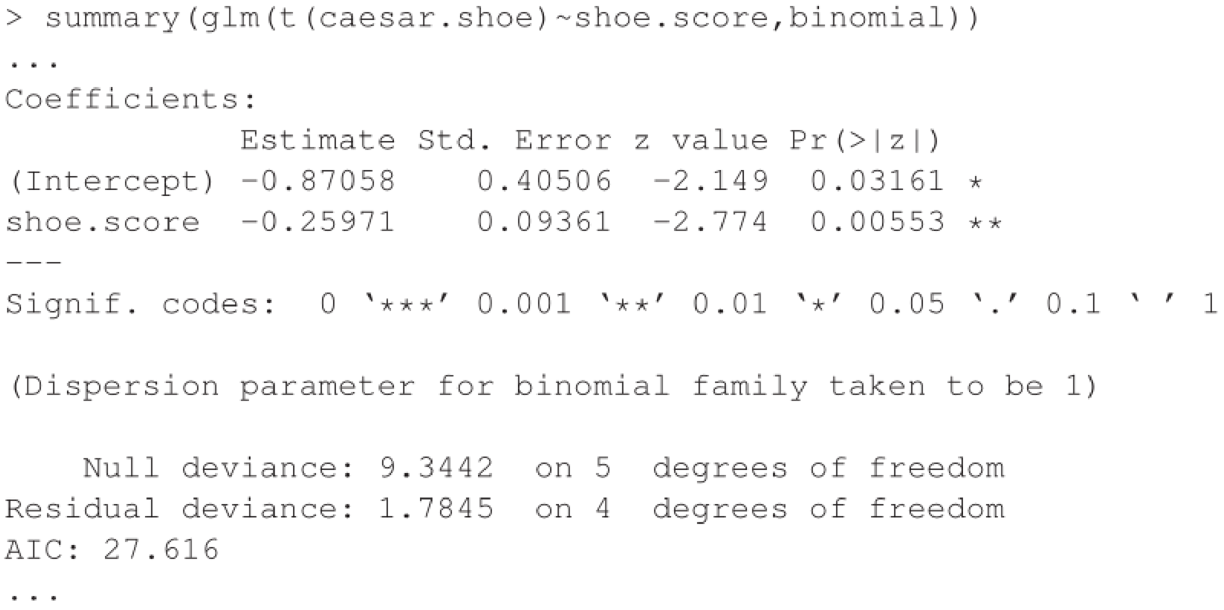
Connection to Test for Trend

❖ Tests for comparing relative frequencies

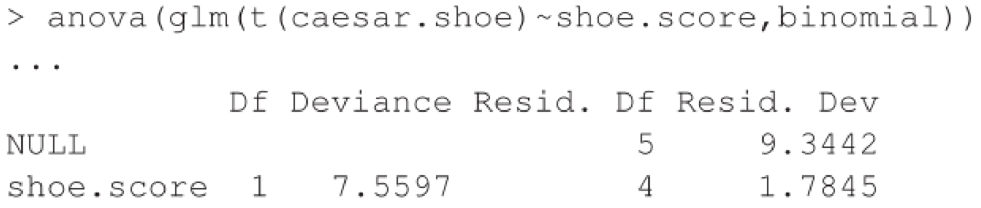
❖ prop.test ()
❖ prop.trend.test ()



```
> caesar.shoe
     <4   4 4.5   5 5.5   6+
Yes   5   7   6   7   8   10
No   17  28  36  41  46  140
> shoe.score <- 1:6
> shoe.score
[1] 1 2 3 4 5 6
```

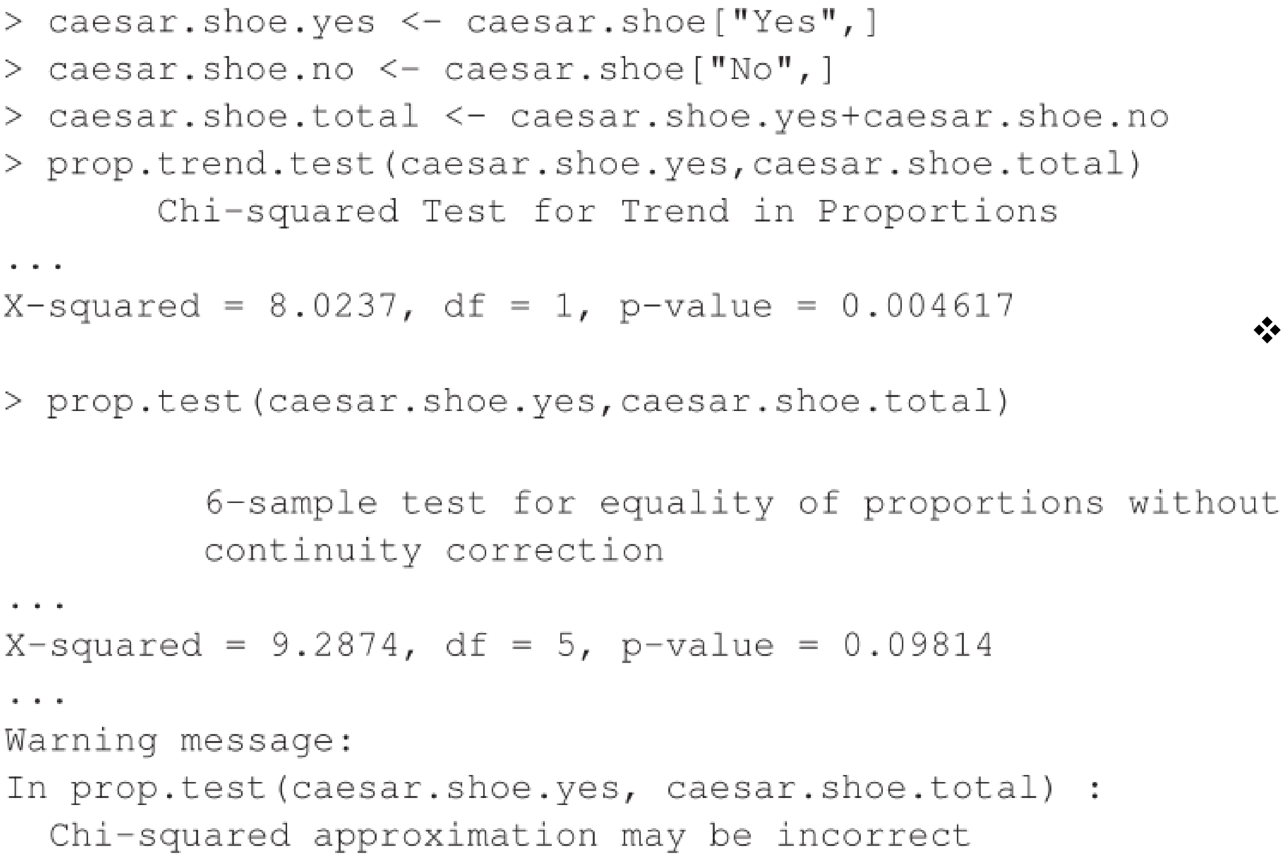


```
> summary(glm(t(caesar.shoe)~shoe.score,binomial))
...
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.87058    0.40506  -2.149  0.03161 *
shoe.score  -0.25971    0.09361  -2.774  0.00553 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9.3442  on 5  degrees of freedom
Residual deviance: 1.7845  on 4  degrees of freedom
AIC: 27.616
...
```



```
> anova(glm(t(caesar.shoe)~shoe.score,binomial))
...
           Df Deviance Resid. Df Resid. Dev
NULL                           5     9.3442
shoe.score  1   7.5597         4     1.7845
```

Connection to Test for Trend



```
> caesar.shoe.yes <- caesar.shoe["Yes",]
> caesar.shoe.no <- caesar.shoe["No",]
> caesar.shoe.total <- caesar.shoe.yes+caesar.shoe.no
> prop.trend.test(caesar.shoe.yes,caesar.shoe.total)
       Chi-squared Test for Trend in Proportions
...
X-squared = 8.0237, df = 1, p-value = 0.004617

> prop.test(caesar.shoe.yes,caesar.shoe.total)


         6-sample test for equality of proportions without
         continuity correction
...
X-squared = 9.2874, df = 5, p-value = 0.09814
...
Warning message:
In prop.test(caesar.shoe.yes, caesar.shoe.total) :
  Chi-squared approximation may be incorrect
```

- Generally almost the same

Likelihood Profiling

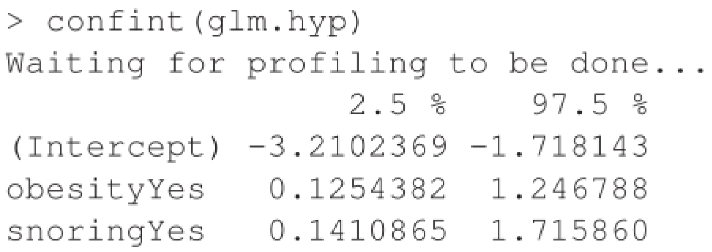
Z test
Based on Wald approximation

Wald approximation
만약 true values가 estimates와 같다면, parameter estimate의 approximate standard error를 계산하는 것

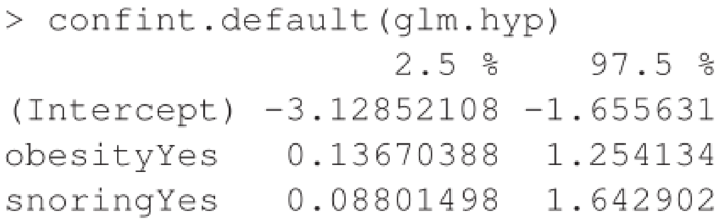
In large data sets -> no problem
In smaller data sets -> the difference between the Wald tests and the likelihood ratio test can be considerable.

▼

Affecting the calculation of confidence intervals



```
> confint(glm.hyp)
Waiting for profiling to be done...
                 2.5 %    97.5 %
(Intercept) -3.2102369 -1.718143
obesityYes   0.1254382  1.246788
snoringYes   0.1410865  1.715860
```



```
> confint.default(glm.hyp)
                  2.5 %    97.5 %
(Intercept) -3.12852108 -1.655631
obesityYes   0.13670388  1.254134
snoringYes   0.08801498  1.642902
```

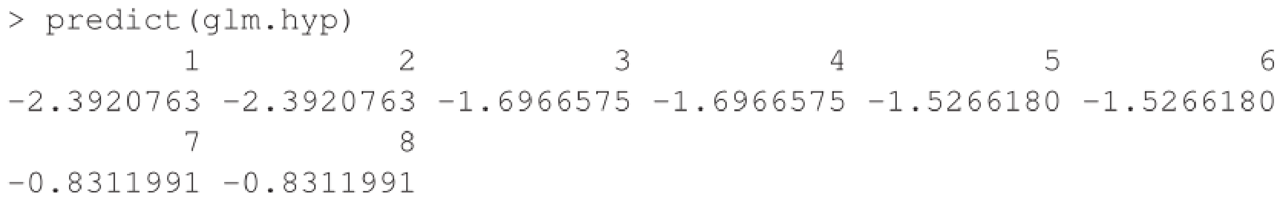
Presentation as Odds-ratio Estimates

Logistic regression using raw data

Prediction

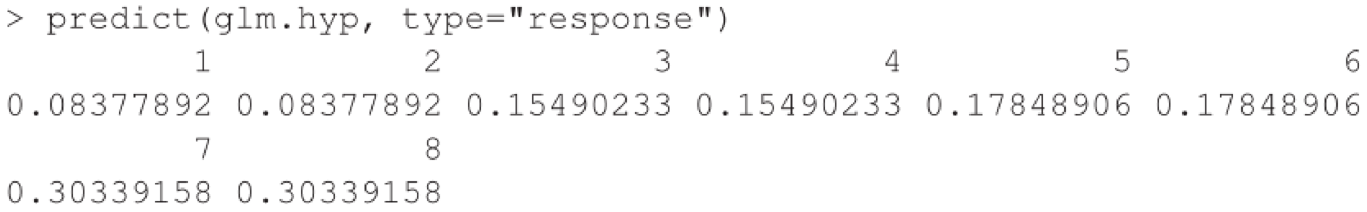
predict ()

Working for generalized linear models



```
> predict(glm.hyp)
         1          2          3          4          5          6 
-2.3920763 -2.3920763 -1.6966575 -1.6966575 -1.5266180 -1.5266180 
         7          8 
-0.8311991 -0.8311991 
```

Regression coefficient
2.392 – 1.697 = 1.527 – 0.831 = 0.695
2.392 – 1.527 = 1.697 – 0.831 = 0.866



```
> predict(glm.hyp, type="response")
         1          2          3          4          5          6 
0.08377892 0.08377892 0.15490233 0.15490233 0.17848906 0.17848906 
         7          8 
0.30339158 0.30339158 
```

Chap.14 Survival Analysis

Connection to Test for Trend

The analysis of lifetimes

- An important topic within biology and medicine

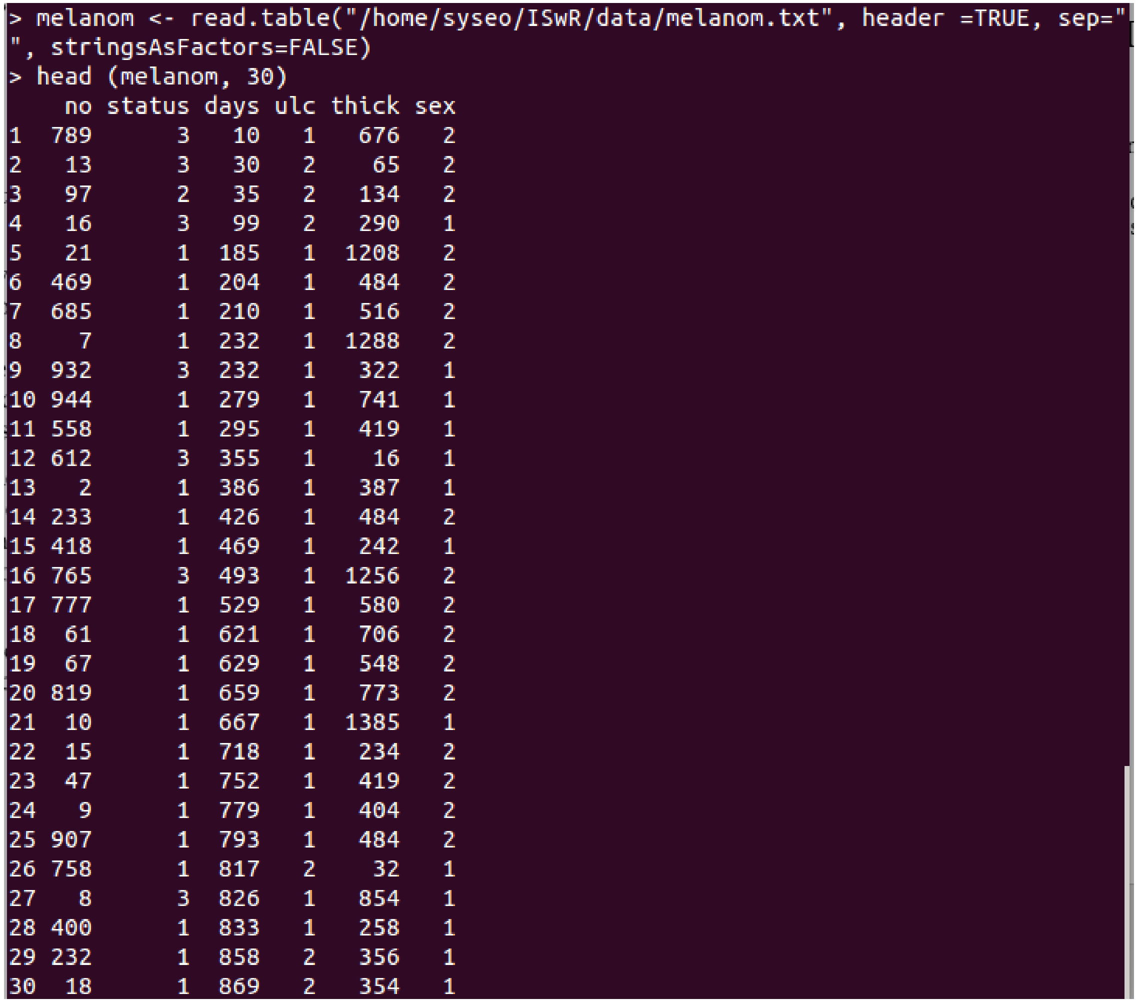
- Often highly nonnormally distributed
- Not using the standard linear models

- Often censored (the period of observation was cut off before the event of interest occurred.)

Essential concepts		
X	true lifetime	
T	censoring time	random variable fixed time depending on context noninformative for the method
The observations -> the minimum of X and T		

S (t)	survival function	probability of being alive at a given time 1 – cumulative distribution function for X (F (t))
h (t)	hazard function or force of mortality	infinitesimal risk of dying within a short interval of time t
f (t)	density of the lifetime distribution	
t	time that the subject is alive	
h (t) = f (t) / S (t) more fundamental quantity than the mean or median of survival distribution a basis for modelling		

Survival Objects



```
> melanom <- read.table("/home/syseo/ISwR/data/melanom.txt", header =TRUE, sep="
", stringsAsFactors=FALSE)
> head (melanom, 30)
    no status days ulc thick sex
1  789      3   10   1   676   2
2   13      3   30   2    65   2
3   97      2   35   2   134   2
4   16      3   99   2   290   1
5   21      1  185   1  1208   2
6  469      1  204   1   484   2
7  685      1  210   1   516   2
8    7      1  232   1  1288   2
9  932      3  232   1   322   1
10 944      1  279   1   741   1
11 558      1  295   1   419   1
12 612      3  355   1    16   1
13   2      1  386   1   387   1
14 233      1  426   1   484   2
15 418      1  469   1   242   1
16 765      3  493   1  1256   2
17 777      1  529   1   580   2
18  61      1  621   1   706   2
19  67      1  629   1   548   2
20 819      1  659   1   773   2
21  10      1  667   1  1385   1
22  15      1  718   1   234   2
23  47      1  752   1   419   2
24   9      1  779   1   404   2
25 907      1  793   1   484   2
26 758      1  817   2    32   1
27   8      3  826   1   854   1
28 400      1  833   1   258   1
29 232      1  858   2   356   1
30  18      1  869   2   354   1
```

The explanation of variables

'status'
Indicator of the patient's status by the end of the study
1 -> "dead from malignant melanoma"
2 -> "alive on January 1, 1978"
3 -> "dead from other causes"

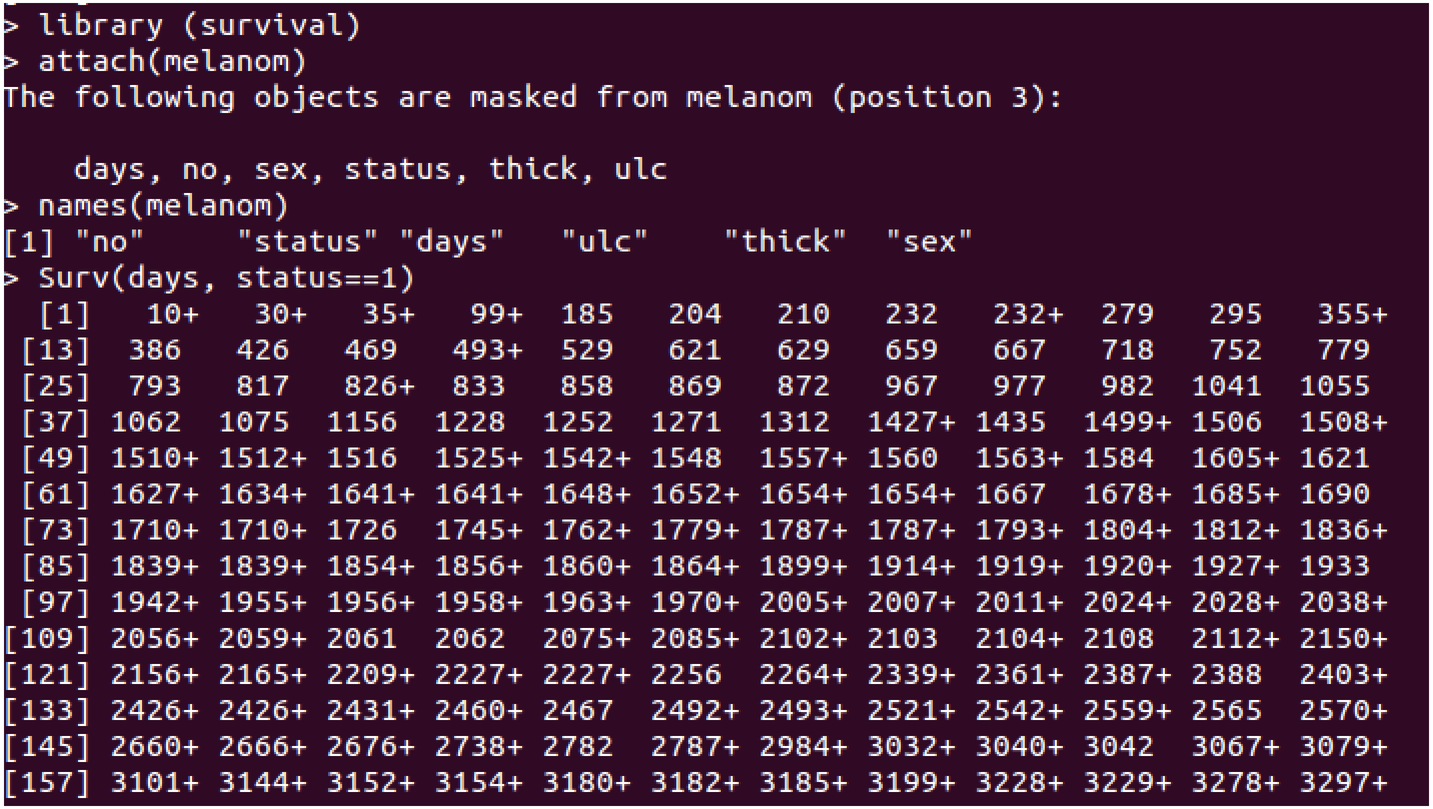
'days'
Observation time in days

'ulc' ulcerated tumor
1 -> present
2 -> absent

'thick'
Thickness in 1/100 mm

'sex'
Gender of the patient
1 -> women
2 -> men

Survival Objects



```
> library (survival)
> attach(melanom)
The following objects are masked from melanom (position 3):

    days, no, sex, status, thick, ulc
> names(melanom)
[1] "no"     "status" "days"   "ulc"    "thick"  "sex"
> Surv(days, status==1)
  [1]   10+   30+   35+   99+  185   204   210   232   232+  279   295   355+
 [13]  386   426   469   493+  529   621   629   659   667   718   752   779
 [25]  793   817   826+  833   858   869   872   967   977   982  1041  1055
 [37] 1062  1075  1156  1228  1252  1271  1312  1427+ 1435  1499+ 1506  1508+
 [49] 1510+ 1512+ 1516  1525+ 1542+ 1548  1557+ 1560  1563+ 1584  1605+ 1621
 [61] 1627+ 1634+ 1641+ 1641+ 1648+ 1652+ 1654+ 1654+ 1667  1678+ 1685+ 1690
 [73] 1710+ 1710+ 1726  1745+ 1762+ 1779+ 1787+ 1787+ 1793+ 1804+ 1812+ 1836+
 [85] 1839+ 1839+ 1854+ 1856+ 1860+ 1864+ 1899+ 1914+ 1919+ 1920+ 1927+ 1933
 [97] 1942+ 1955+ 1956+ 1958+ 1963+ 1970+ 2005+ 2007+ 2011+ 2024+ 2028+ 2038+
[109] 2056+ 2059+ 2061  2062  2075+ 2085+ 2102+ 2103  2104+ 2108  2112+ 2150+
[121] 2156+ 2165+ 2209+ 2227+ 2227+ 2256  2264+ 2339+ 2361+ 2387+ 2388  2403+
[133] 2426+ 2426+ 2431+ 2460+ 2467  2492+ 2493+ 2521+ 2542+ 2559+ 2565  2570+
[145] 2660+ 2666+ 2676+ 2738+ 2782  2787+ 2984+ 3032+ 3040+ 3042  3067+ 3079+
[157] 3101+ 3144+ 3152+ 3154+ 3180+ 3182+ 3185+ 3199+ 3228+ 3229+ 3278+ 3297+
```

Surv objects

Print method that displays the objects in the format above, with a '+'
marking censored observations
status ==1 -> logical vector, (TRUE = died of malignant melanoma)

10+ -> not die from melanoma within 10 days
died from other causes

185 -> died from the disease

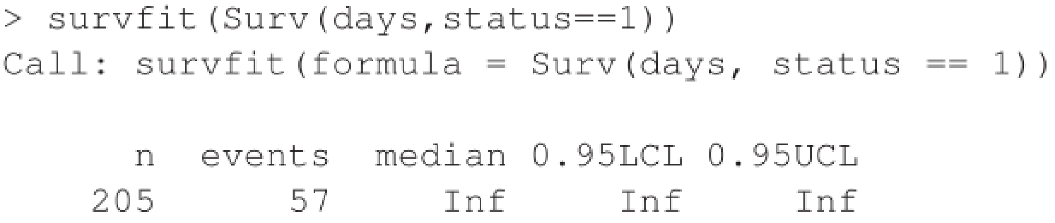
Kaplan–Meier Estimates

Computation of an estimated survival function in the presence of right-censoring

Product-limit estimator
Multiplying together conditional survival curves for intervals in which there are either no censored observations or no deaths

Step function
Reducing the estimated survival by a factor (1-1/R_t)
T: Death time
R_t: Still alive and uncensored at that time

survfit ()
Only single argument, Surv object



```
> survfit(Surv(days,status==1))
Call: survfit(formula = Surv(days, status == 1))

      n  events  median 0.95LCL 0.95UCL
    205      57     Inf     Inf     Inf
```

Couple of summary statistics
Estimate of the median survival

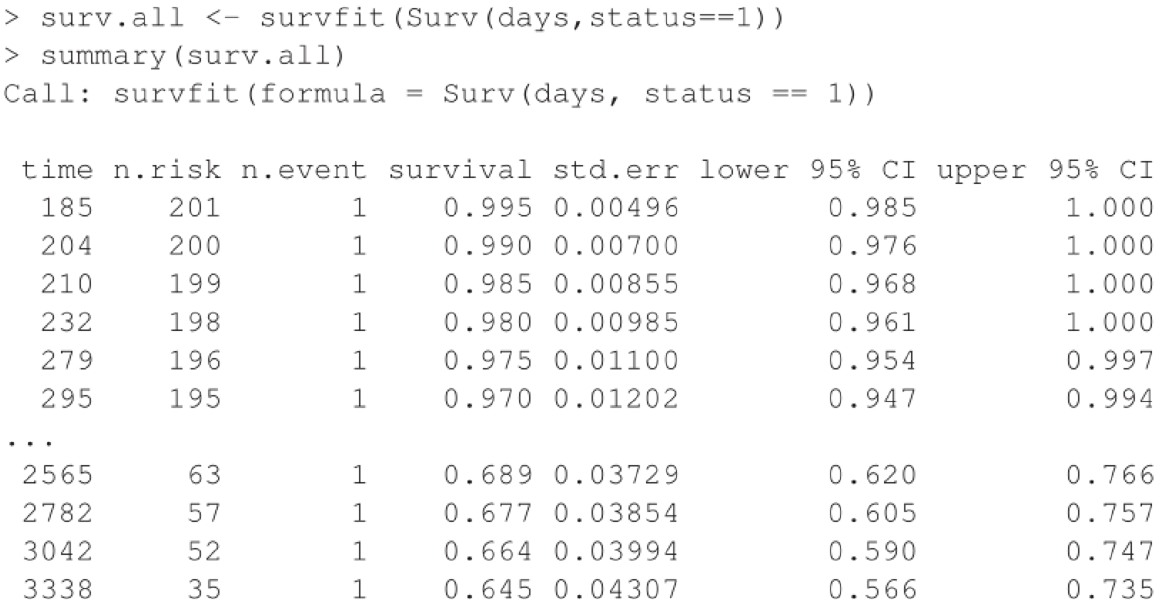
Not informative
Not even interesting <- infinite

Kaplan–Meier Estimates

To see the actual Kaplan–Meier estimate

Using summary on the survfit object

surv.all -> the raw survival function for all patients without regard of patient characteristic



```
> surv.all <- survfit(Surv(days,status==1))
> summary(surv.all)
Call: survfit(formula = Surv(days, status == 1))

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
  185    201       1    0.995 0.00496        0.985        1.000
  204    200       1    0.990 0.00700        0.976        1.000
  210    199       1    0.985 0.00855        0.968        1.000
  232    198       1    0.980 0.00985        0.961        1.000
  279    196       1    0.975 0.01100        0.954        0.997
  295    195       1    0.970 0.01202        0.947        0.994
...
 2565     63       1    0.689 0.03729        0.620        0.766
 2782     57       1    0.677 0.03854        0.605        0.757
 3042     52       1    0.664 0.03994        0.590        0.747
 3338     35       1    0.645 0.04307        0.566        0.735
```

Values of the survival function at the event times

Step function

1. Jump points are given in time
2. Values right after a jump are given in survival

Kaplan–Meier Estimates_Practice

```
> plot(surv.all)
```

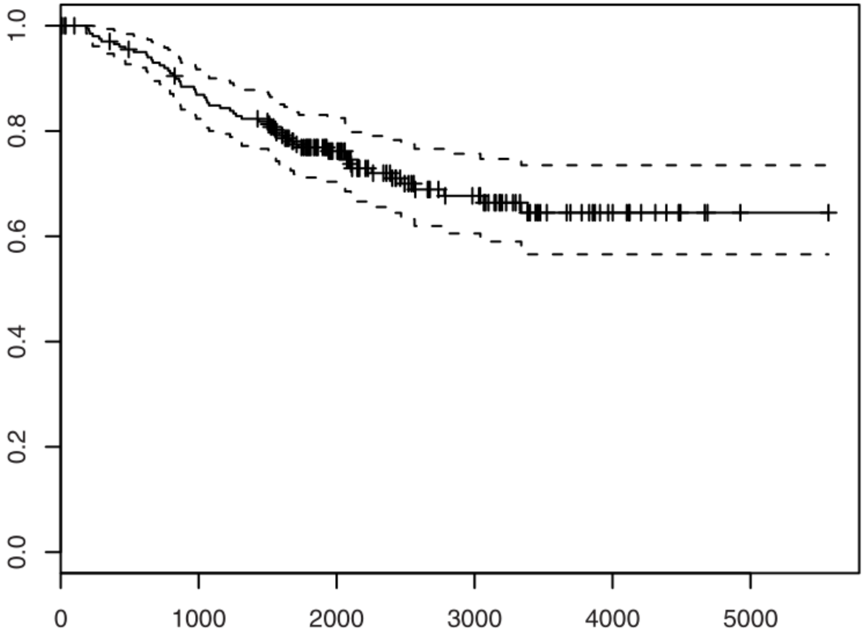


Figure 14.1. Kaplan–Meier plot for melanoma data (all observations).

Markings on the curve -> censoring times
Bands -> approximate confidence intervals

Symmetric interval on the log scale

```
> surv.bysex <- survfit(Surv(days,status==1)~sex)
> plot(surv.bysex)
```

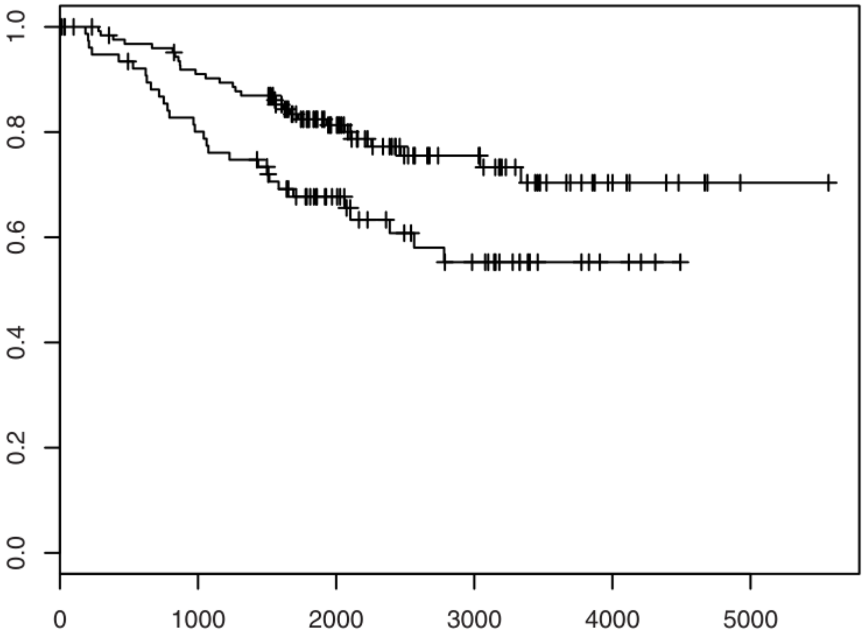


Figure 14.2. Kaplan–Meier plots for melanoma data, grouped by gender.

Splitting by gender

No confidence intervals on the curves

The Log-rank Test

To test whether two or more survival curves are identical
Hypothesis test to compare the survival distributions of two samples

Nonparametric test
Appropriate to use when the data are right skewed and censored (non-informative)

Establishing the efficacy of a new treatment in comparison with a control treatment

Computing the observed and expected number of events in one of the groups at each observed event time
Adding these to obtain an overall summary across all-time points

Comparing estimates of the hazard functions of the two groups at each observed event time

The same test as the score test from the Cox proportional hazard model

survdiff function

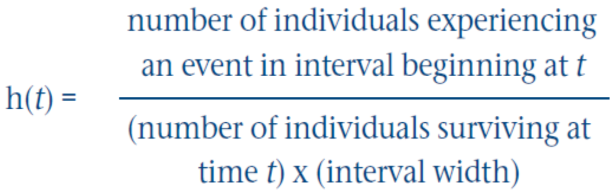
The Cox Proportional Hazards Model

Survival models

Analysis of survival data by regression models

Fitted via the maximization of Cox's likelihood

Hazard function
How the risk of event per time unit changes over time at baseline levels of covariates

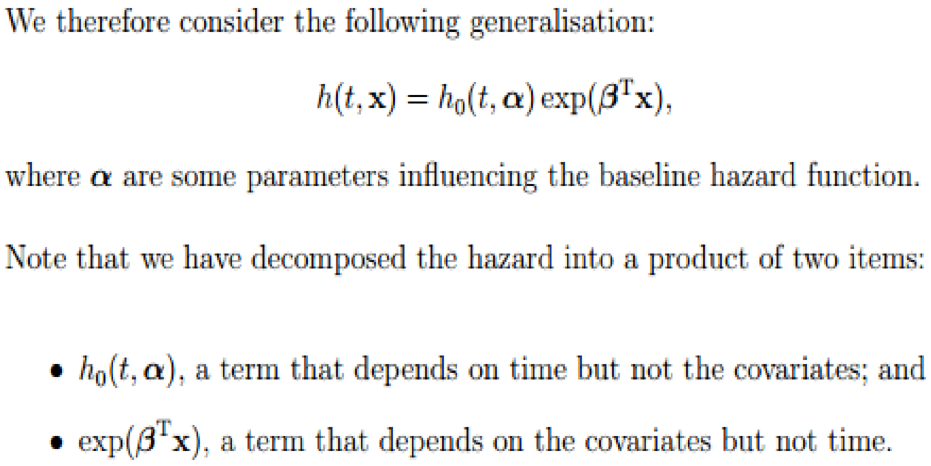


$$h(t) = \frac{\text{number of individuals experiencing an event in interval beginning at } t}{(\text{number of individuals surviving at time } t) \text{ x (interval width)}}$$

Proportional hazards condition
Covariates -> multiplicatively related to the hazard
Not restricted to binary predictors

The effect parameters
How the hazard varies in response to explanatory covariates

Cox partial likelihood
Obtained by using Breslow's estimate of the baseline hazard function
Plugging it into the full likelihood
Observing that the result is a product of two factors.



We therefore consider the following generalisation:

$$h(t, \mathbf{x}) = h_0(t, \boldsymbol{\alpha}) \exp(\boldsymbol{\beta}^{\mathrm{T}} \mathbf{x}),$$

where $\boldsymbol{\alpha}$ are some parameters influencing the baseline hazard function.

Note that we have decomposed the hazard into a product of two items:

- $h_0(t, \boldsymbol{\alpha})$, a term that depends on time but not the covariates; and
- $\exp(\boldsymbol{\beta}^{\mathrm{T}} \mathbf{x})$, a term that depends on the covariates but not time.

Thank you for your attention