Biostatistics (MATH11230), 2022/2023

Vanda Inácio

In this document I show how to obtain the maximum likelihood estimates of β_0 and β_1 using both the `optim` function and the `glm` function with the argument `family` set to `binomial`. The dataset `wcgsdata` is also available on Learn and it has several (exposure) variables. For now, we will only use the weight and CHD variables.

```
require(readxl)
data_wchs <- read_excel("wcgsdata.xls")
names(data_wchs)
```

```
##  [1] "Id"      "Age0"    "Height0" "Weight0" "Sbp0"    "Dbp0"    "Chol0"
##  [8] "Behpat0" "Ncigs0"  "Dibpat0" "Chd69"   "Typechd" "Time169" "Arcus0"
```

```
head(data_wchs)
```

```
## # A tibble: 6 x 14
##      Id  Age0 Height0 Weight0  Sbp0  Dbp0 Chol0 Behpat0 Ncigs0 Dibpat0 Chd69
##   <dbl> <dbl>   <dbl>   <dbl> <dbl> <dbl> <chr>   <dbl>  <dbl>   <dbl> <dbl>
## 1  2001    49      73     150   110    76 225         2     25       1     0
## 2  2002    42      70     160   154    84 177         2     20       1     0
## 3  2003    42      69     160   110    78 181         3      0       0     0
## 4  2004    41      68     152   124    78 132         4     20       0     0
## 5  2005    59      70     150   144    86 255         3     20       0     1
## 6  2006    44      72     204   150    90 182         4      0       0     0
## # ... with 3 more variables: Typechd <dbl>, Time169 <dbl>, Arcus0 <chr>
```

```
#Log likelihood function logistic regression model (slide 12)
log_like <- function(param, d, x){
beta0 <- param[1]
beta1 <- param[2]
sum((d*(beta0+beta1*x)) - log(1+exp(beta0+beta1*x)))
}

res_optim <- optim(c(0, 0), log_like,
                   d = data_wchs$Chd69, x = data_wchs$Weight0,
                   control = list(fnscale = -1), hessian = TRUE)
res_optim
```

```
## $par
## [1] -4.21609890  0.01043202
##
## $value
## [1] -884.4688
##
## $counts
## function gradient
##      101       NA
```

```
##
## $convergence
## [1] 0
##
## $message
## NULL
##
## $hessian
##             [,1]        [,2]
## [1,]   -235.0617   -40999.69
## [2,] -40999.6914 -7268930.98
```

```
#Inverse of the observed Fisher information matrix
# (the diagonal extract the standard errors of beta0 and beta1)
sqrt(diag(solve(-1*res_optim$hessian)))
```

```
## [1] 0.512516688 0.002914498
```

```
#Results using the glm function
res_glm <- glm(Chd69 ~ Weight0, family = "binomial",
               data = data_wchs)
out_glm <- summary(res_glm)
out_glm
```

```
##
## Call:
## glm(formula = Chd69 ~ Weight0, family = "binomial", data = data_wchs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7283  -0.4292  -0.3982  -0.3693   2.4146
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.21471    0.51206  -8.231  < 2e-16 ***
## Weight0      0.01042    0.00292   3.570 0.000356 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1781.2  on 3153  degrees of freedom
## Residual deviance: 1768.9  on 3152  degrees of freedom
## AIC: 1772.9
##
## Number of Fisher Scoring iterations: 5
```

```
exp(res_glm$coefficients)
```

```
## (Intercept)     Weight0
##  0.01477667  1.01047871
```

```
exp(confint.default(res_glm, level = 0.95))
```

```
##                  2.5 %     97.5 %
## (Intercept) 0.00541636 0.04031303
## Weight0     1.00471301 1.01627750
```

```r
alpha <- 0.05
exp(out_glm$coefficients[2, 1] - qnorm(1-(alpha/2))* out_glm$coefficients[2, 2])
```

```
## [1] 1.004713
```

```r
exp(out_glm$coefficients[2, 1] + qnorm(1-(alpha/2))* out_glm$coefficients[2, 2])
```

```
## [1] 1.016278
```