Logistic Regression Tutorial

Unnamed

2018/05/24

http://ww2.coastal.edu/kingw/statistics/R-tutorials/logistic.html

$$http://ww2.coastal.edu/kingw/statistics/R-tutorials/logistic.html$$

Example 1: In the "MASS" library there is a data set called "menarche" (Milicer, H. and Szczotka, F., 1966, Age at Menarche in Warsaw girls in 1965, Human Biology, 38, 199-203), in which there are three variables: "Age" (average age of age homogeneous groups of girls), "Total" (number of girls in each group), and "Menarche" (number of girls in the group who have reached menarche).

Logistic Regression: One Numeric Predictor

```
library("MASS")
help("menarche")
```

```
## starting httpd help server ... done
```

```
data("menarche")
str(menarche)
```

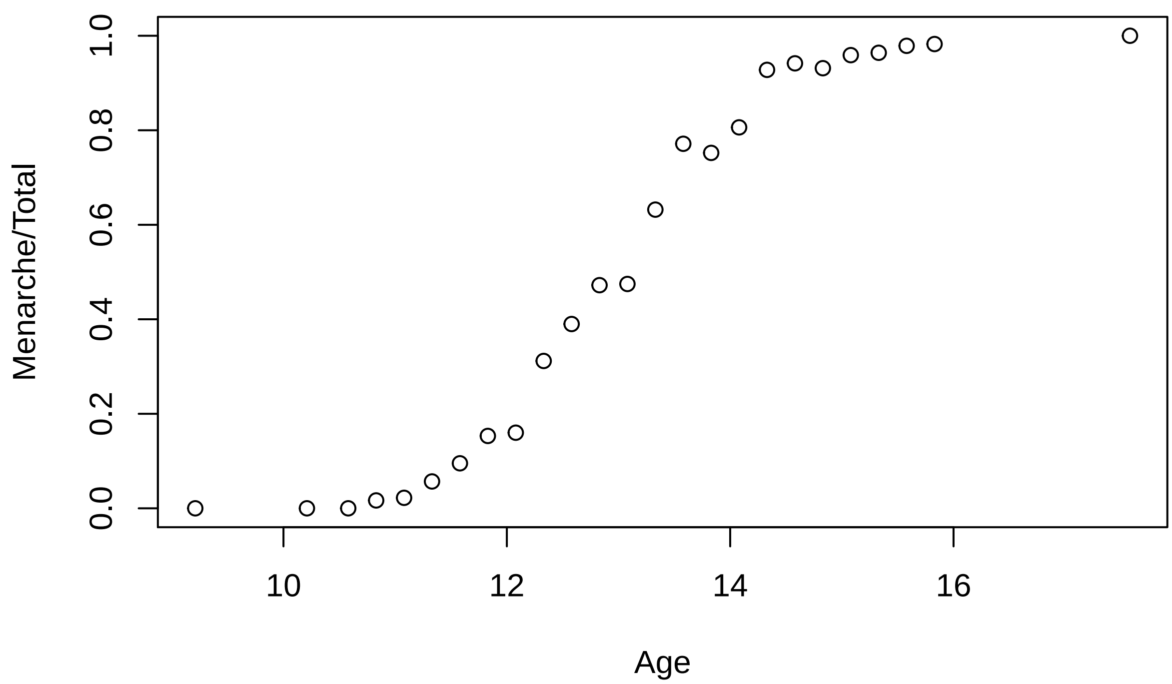
```
## 'data.frame':    25 obs. of  3 variables:
##  $ Age     : num  9.21 10.21 10.58 10.83 11.08 ...
##  $ Total   : num  376 200 93 120 90 88 105 111 100 93 ...
##  $ Menarche: num  0 0 0 2 2 5 10 17 16 29 ...
```

```
?str()
```

```
summary(menarche)
```

```
##       Age            Total           Menarche
##  Min.   : 9.21   Min.   :  88.0   Min.   :   0.00
##  1st Qu.:11.58   1st Qu.:  98.0   1st Qu.:  10.00
##  Median :13.08   Median : 105.0   Median :  51.00
##  Mean   :13.10   Mean   : 156.7   Mean   :  92.32
##  3rd Qu.:14.58   3rd Qu.: 117.0   3rd Qu.:  92.00
##  Max.   :17.58   Max.   :1049.0   Max.   :1049.00
```

```
plot(Menarche/Total ~ Age, data=menarche)
```




```
menarche$Total
```

```
##  [1]  376  200   93  120   90   88  105  111  100   93  100  108   99  106
## [15]  105  117   98   97  120  102  122  111   94  114 1049
```

```
cbind(menarche$Menarche, menarche$Total-menarche$Menarche)
```

```
##       [,1] [,2]
##  [1,]    0  376
```

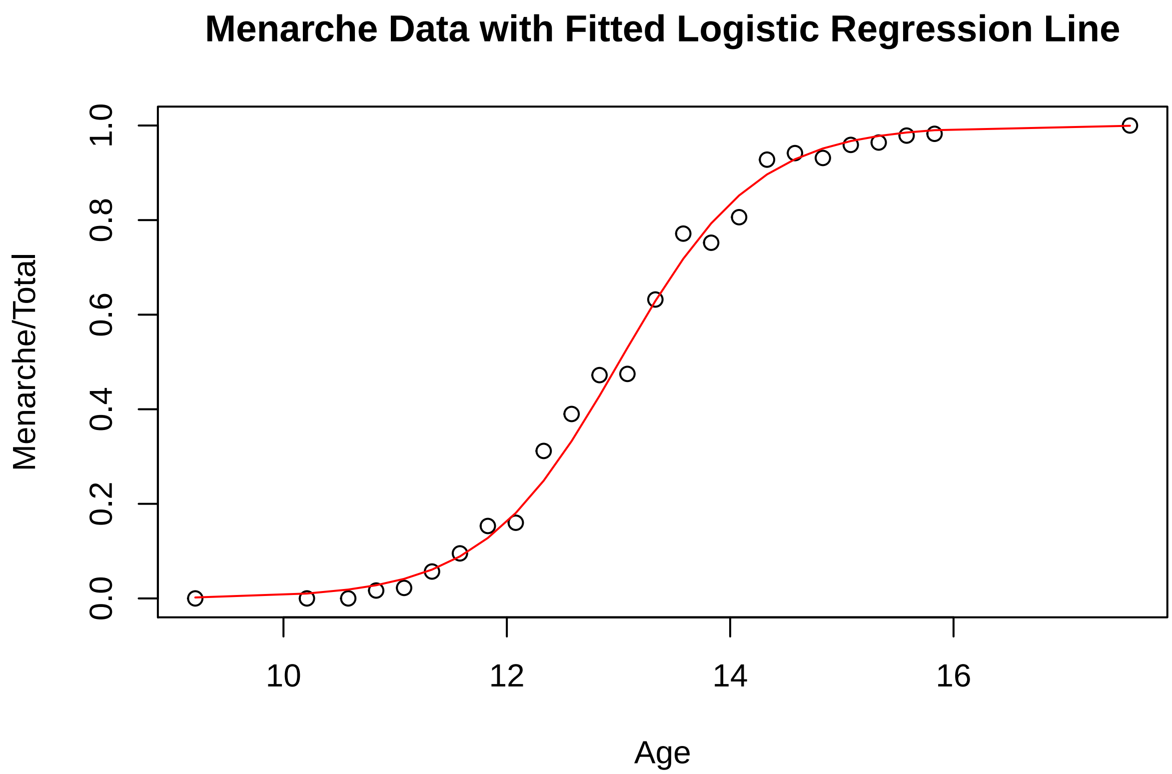
```
##  [2,]    0  200
##  [3,]    0   93
##  [4,]    2  118
##  [5,]    2   88
##  [6,]    5   83
##  [7,]   10   95
##  [8,]   17   94
##  [9,]   16   84
## [10,]   29   64
## [11,]   39   61
## [12,]   51   57
## [13,]   47   52
## [14,]   67   39
## [15,]   81   24
## [16,]   88   29
## [17,]   79   19
## [18,]   90    7
## [19,]  113    7
## [20,]   95    7
## [21,]  117    5
## [22,]  107    4
## [23,]   92    2
## [24,]  112    2
## [25,] 1049    0
```

```r
glm.out = glm(cbind(Menarche, Total-Menarche) ~ Age, family=binomial(logit), data=menarche)
summary(glm.out)
```

```
##
## Call:
## glm(formula = cbind(Menarche, Total - Menarche) ~ Age, family = binomial(logit),
##     data = menarche)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0363  -0.9953  -0.4900   0.7780   1.3675
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -21.22639    0.77068  -27.54   <2e-16 ***
```

```
## Age           1.63197    0.05895   27.68   <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: 3693.884  on 24  degrees of freedom
## Residual deviance:   26.703  on 23  degrees of freedom
## AIC: 114.76
##
## Number of Fisher Scoring iterations: 4
```

```
plot(Menarche/Total ~ Age, data=menarche)
lines(menarche$Age, glm.out$fitted, type="l", col="red")
title(main="Menarche Data with Fitted Logistic Regression Line")
```



https://stats.idre.ucla.edu/r/dae/logit-regression/

$$https://stats.idre.ucla.edu/r/dae/logit-regression/$$

Example 2. A researcher is interested in how variables, such as GRE (Graduate Record Exam scores), GPA (grade point average) and prestige of the undergraduate institution, effect admission into graduate school. The response variable, admit/don't admit, is a binary variable.

```
mydata <- read.csv("https://stats.idre.ucla.edu/stat/data/binary.csv")
## view the first few rows of the data
head(mydata)
```

```
##   admit gre  gpa rank
## 1     0 380 3.61    3
## 2     1 660 3.67    3
## 3     1 800 4.00    1
## 4     1 640 3.19    4
## 5     0 520 2.93    4
## 6     1 760 3.00    2
```

```
summary(mydata)
```

```
##      admit             gre             gpa             rank
##  Min.   :0.0000   Min.   :220.0   Min.   :2.260   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:520.0   1st Qu.:3.130   1st Qu.:2.000
##  Median :0.0000   Median :580.0   Median :3.395   Median :2.000
##  Mean   :0.3175   Mean   :587.7   Mean   :3.390   Mean   :2.485
##  3rd Qu.:1.0000   3rd Qu.:660.0   3rd Qu.:3.670   3rd Qu.:3.000
##  Max.   :1.0000   Max.   :800.0   Max.   :4.000   Max.   :4.000
```

```
sd(mydata$admit)
```

```
## [1] 0.4660867
```

```
help(sapply)
sapply(mydata, sd)
```

```
##       admit         gre         gpa        rank
##   0.4660867 115.5165364   0.3805668   0.9444602
```

```r
## two-way contingency table of categorical outcome and predictors we want
## to make sure there are not 0 cells
help(xtabs)
xtabs(~admit + rank, data = mydata)
```

```
##      rank
## admit  1  2  3  4
##     0 28 97 93 55
##     1 33 54 28 12
```

```r
mydata$rank <- factor(mydata$rank)
mydata$rank
```

```
##   [1] 3 3 1 4 4 2 1 2 3 2 4 1 1 2 1 3 4 3 2 1 3 2 4 4 2 1 1 4 2 1 4 3 3 3 1
##  [36] 2 1 3 2 3 2 2 2 3 2 3 2 4 4 3 3 4 4 2 3 3 3 3 2 4 2 4 3 3 3 2 4 1 1 1
##  [71] 3 4 4 2 4 3 3 3 1 1 4 2 2 4 3 2 2 2 1 2 2 1 2 2 2 2 4 2 2 3 3 3 4 3 2
## [106] 2 1 2 3 2 4 4 3 1 3 3 2 2 1 3 2 2 3 3 3 4 1 4 2 4 2 2 2 3 2 3 4 3 2 1
## [141] 2 4 4 3 4 3 2 3 1 1 1 2 2 3 3 4 2 1 2 3 2 2 2 2 2 1 4 3 3 3 3 3 3 2 4
## [176] 2 2 3 3 3 3 4 2 2 4 2 3 2 2 2 2 3 3 4 2 2 3 4 3 4 3 2 1 4 1 3 1 1 3 2
## [211] 4 2 2 3 2 3 1 1 1 2 3 3 1 3 2 3 2 4 2 2 4 3 2 3 1 2 2 2 4 3 2 1 3 2 1
## [246] 3 2 2 3 3 4 4 2 4 4 3 2 3 2 2 2 2 3 3 3 3 4 3 2 3 2 3 2 1 2 2 3 1 4 2
## [281] 2 3 4 4 2 4 1 4 4 4 2 2 2 1 1 3 1 2 2 3 2 3 2 2 3 4 1 2 2 3 3 2 3 4 4
## [316] 2 2 4 4 1 3 2 4 2 3 1 2 2 2 4 3 3 1 3 3 1 3 4 1 3 4 3 4 2 3 3 2 2 2 2
## [351] 2 3 3 2 2 1 2 1 3 3 1 1 2 2 1 3 3 3 1 2 2 3 1 1 2 4 2 2 3 2 2 2 2 1 2
## [386] 1 2 2 2 2 2 2 3 2 3 2 3 2 2 3
## Levels: 1 2 3 4
```

```r
mylogit <- glm(admit ~ gre + gpa + rank, data = mydata, family = "binomial")
```

```r
summary(mylogit)
```

```
##
## Call:
## glm(formula = admit ~ gre + gpa + rank, family = "binomial",
##     data = mydata)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6268  -0.8662  -0.6388   1.1490   2.0790
```

```
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.989979   1.139951  -3.500 0.000465 ***
## gre          0.002264   0.001094   2.070 0.038465 *
## gpa          0.804038   0.331819   2.423 0.015388 *
## rank2       -0.675443   0.316490  -2.134 0.032829 *
## rank3       -1.340204   0.345306  -3.881 0.000104 ***
## rank4       -1.551464   0.417832  -3.713 0.000205 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 499.98  on 399  degrees of freedom
## Residual deviance: 458.52  on 394  degrees of freedom
## AIC: 470.52
##
## Number of Fisher Scoring iterations: 4
```

```
## CIs using profiled log-likelihood
confint(mylogit)
```

```
## Waiting for profiling to be done...
```

```
##                     2.5 %       97.5 %
## (Intercept) -6.2716202334 -1.792547080
## gre          0.0001375921  0.004435874
## gpa          0.1602959439  1.464142727
## rank2       -1.3008888002 -0.056745722
## rank3       -2.0276713127 -0.670372346
## rank4       -2.4000265384 -0.753542605
```

```
## CIs using standard errors
confint.default(mylogit)
```

```
##                     2.5 %       97.5 %
## (Intercept) -6.2242418514 -1.755716295
## gre          0.0001202298  0.004408622
## gpa          0.1536836760  1.454391423
```

```
## rank2       -1.2957512650 -0.055134591
## rank3       -2.0169920597 -0.663415773
## rank4       -2.3703986294 -0.732528724
```

```r
library(aod)
library(ggplot2)
```

```r
wald.test(b = coef(mylogit), Sigma = vcov(mylogit), Terms = 4:6)
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 20.9, df = 3, P(> X2) = 0.00011
```

```r
## odds ratios only
coef(mylogit)
```

```
##  (Intercept)          gre          gpa        rank2        rank3
## -3.989979073  0.002264426  0.804037549 -0.675442928 -1.340203916
##        rank4
## -1.551463677
```

```r
## odds ratios only
exp(coef(mylogit))
```

```
## (Intercept)         gre         gpa       rank2       rank3       rank4
##   0.0185001   1.0022670   2.2345448   0.5089310   0.2617923   0.2119375
```

```r
## odds ratios and 95% CI
exp(cbind(OR = coef(mylogit), confint(mylogit)))
```

```
## Waiting for profiling to be done...
```

```
##                    OR       2.5 %    97.5 %
## (Intercept) 0.0185001 0.001889165 0.1665354
## gre         1.0022670 1.000137602 1.0044457
## gpa         2.2345448 1.173858216 4.3238349
## rank2       0.5089310 0.272289674 0.9448343
## rank3       0.2617923 0.131641717 0.5115181
## rank4       0.2119375 0.090715546 0.4706961
```

```r
newdata1 <- with(mydata, data.frame(gre = mean(gre), gpa = mean(gpa), rank = factor(1:4)))

## view data frame
newdata1
```

```
##     gre    gpa rank
## 1 587.7 3.3899    1
## 2 587.7 3.3899    2
## 3 587.7 3.3899    3
## 4 587.7 3.3899    4
```

```r
newdata1$rankP <- predict(mylogit, newdata = newdata1, type = "response")
newdata1
```

```
##     gre    gpa rank     rankP
## 1 587.7 3.3899    1 0.5166016
## 2 587.7 3.3899    2 0.3522846
## 3 587.7 3.3899    3 0.2186120
## 4 587.7 3.3899    4 0.1846684
```

```r
newdata2 <- with(mydata, data.frame(gre = rep(seq(from = 200, to = 800, length.out = 100),
    4), gpa = mean(gpa), rank = factor(rep(1:4, each = 100))))
```

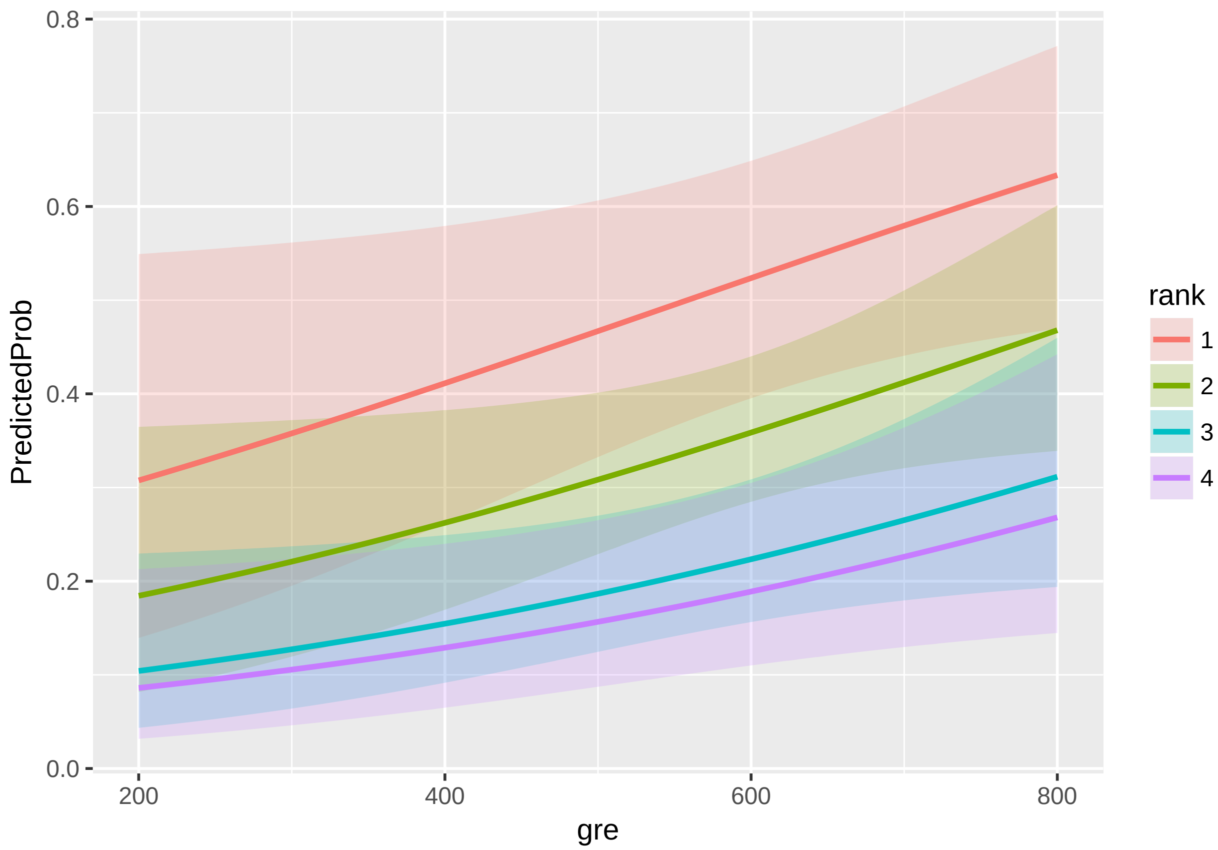
```r
newdata3 <- cbind(newdata2, predict(mylogit, newdata = newdata2, type = "link",
    se = TRUE))
newdata3 <- within(newdata3, {
    PredictedProb <- plogis(fit)
    LL <- plogis(fit - (1.96 * se.fit))
    UL <- plogis(fit + (1.96 * se.fit))
})

## view first few rows of final dataset
head(newdata3)
```

```
##        gre    gpa rank        fit    se.fit residual.scale        UL
## 1 200.0000 3.3899    1 -0.8114870 0.5147714              1 0.5492064
## 2 206.0606 3.3899    1 -0.7977632 0.5090986              1 0.5498513
## 3 212.1212 3.3899    1 -0.7840394 0.5034491              1 0.5505074
## 4 218.1818 3.3899    1 -0.7703156 0.4978239              1 0.5511750
```

```
## 5 224.2424 3.3899    1 -0.7565919 0.4922237            1 0.5518545
## 6 230.3030 3.3899    1 -0.7428681 0.4866494            1 0.5525464
##          LL PredictedProb
## 1 0.1393812     0.3075737
## 2 0.1423880     0.3105042
## 3 0.1454429     0.3134499
## 4 0.1485460     0.3164108
## 5 0.1516973     0.3193867
## 6 0.1548966     0.3223773
```

```r
ggplot(newdata3, aes(x = gre, y = PredictedProb)) + geom_ribbon(aes(ymin = LL,
    ymax = UL, fill = rank), alpha = 0.2) + geom_line(aes(colour = rank),
    size = 1)
```



Back to

$http://ww2.coastal.edu/kingw/statistics/R-tutorials/logistic.html$

Visualising Categorical Data https://rstudio-pubs-static.s3.amazonaws.com/300645_f342587e10674aebafd57e94d1527f
html

$https://rstudio-pubs-static.s3.amazonaws.com/300645_f342587e10674aebafd57e94d1527f20.html$

Example 3: Logistic Regression: Categorical Predictors

In the UCBAdmissions dataset, when we look at the Admit and Gender variables, there appears to be bias towards the number of men being admitted, with women having a lower acceptance rate overall. When we compare Admit and Gender with Dept, this bias disappears and we can see that the admission rates are similar for males and females in most departments, except A.

```r
ftable(UCBAdmissions, col.vars="Admit")
```

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

```r
dimnames(UCBAdmissions)
```

```
## $Admit
## [1] "Admitted" "Rejected"
##
## $Gender
## [1] "Male"   "Female"
##
## $Dept
## [1] "A" "B" "C" "D" "E" "F"
```

```r
margin.table(UCBAdmissions, c(2,1))
```

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

```r
margin.table(UCBAdmissions, c(3,1))
```

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

```r
margin.table(UCBAdmissions, c(2,3))
```

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

```r
### begin copying here
ucb.df = data.frame(gender=rep(c("Male","Female"),c(6,6)),
                    dept=rep(LETTERS[1:6],2),
                    yes=c(512,353,120,138,53,22,89,17,202,131,94,24),
                    no=c(313,207,205,279,138,351,19,8,391,244,299,317))
### end copying here and paste into the R Console

ucb.df
```

```
##    gender dept yes  no
## 1    Male    A 512 313
## 2    Male    B 353 207
## 3    Male    C 120 205
## 4    Male    D 138 279
```

```
## 5     Male     E  53 138
## 6     Male     F  22 351
## 7   Female     A  89  19
## 8   Female     B  17   8
## 9   Female     C 202 391
## 10 Female     D 131 244
## 11 Female     E  94 299
## 12 Female     F  24 317
```

```r
mod.form = "cbind(yes,no) ~ gender * dept"      # mind the quotes here!
glm.out = glm(mod.form, family=binomial(logit), data=ucb.df)
```

```r
anova(glm.out, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: cbind(yes, no)
##
## Terms added sequentially (first to last)
##
##
##             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                           11     877.06
## gender       1    93.45        10     783.61 < 2.2e-16 ***
## dept         5   763.40         5      20.20 < 2.2e-16 ***
## gender:dept  5    20.20         0       0.00  0.001144 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(glm.out)
```

```
##
## Call:
## glm(formula = mod.form, family = binomial(logit), data = ucb.df)
##
## Deviance Residuals:
##  [1]  0  0  0  0  0  0  0  0  0  0  0  0
##
```

```
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)         1.5442     0.2527   6.110 9.94e-10 ***
## genderMale         -1.0521     0.2627  -4.005 6.21e-05 ***
## deptB              -0.7904     0.4977  -1.588  0.11224
## deptC              -2.2046     0.2672  -8.252  < 2e-16 ***
## deptD              -2.1662     0.2750  -7.878 3.32e-15 ***
## deptE              -2.7013     0.2790  -9.682  < 2e-16 ***
## deptF              -4.1250     0.3297 -12.512  < 2e-16 ***
## genderMale:deptB    0.8321     0.5104   1.630  0.10306
## genderMale:deptC    1.1770     0.2996   3.929 8.53e-05 ***
## genderMale:deptD    0.9701     0.3026   3.206  0.00135 **
## genderMale:deptE    1.2523     0.3303   3.791  0.00015 ***
## genderMale:deptF    0.8632     0.4027   2.144  0.03206 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8.7706e+02  on 11  degrees of freedom
## Residual deviance: 1.0791e-13  on  0  degrees of freedom
## AIC: 92.94
##
## Number of Fisher Scoring iterations: 3
```

```r
exp(-1.0521)
```

```
## [1] 0.3492037
```

```r
1/exp(-1.0521)
```

```
## [1] 2.863658
```

```r
exp(-2.2046)
```

```
## [1] 0.1102946
```

```r
exp(-2.2046) / exp(-2.1662)          # C:A / D:A leaves C:D
```

```
## [1] 0.9623279
```

```r
mod.form="cbind(yes,no) ~ dept + gender"
glm.out=glm(mod.form, family=binomial(logit), data=ucb.df)
anova(glm.out, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: cbind(yes, no)
##
## Terms added sequentially (first to last)
##
##
##        Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                      11     877.06
## dept    5   855.32         6      21.74   <2e-16 ***
## gender  1     1.53         5      20.20   0.2159
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(glm.out)
```

```
##
## Call:
## glm(formula = mod.form, family = binomial(logit), data = ucb.df)
##
## Deviance Residuals:
##       1        2        3        4        5        6        7        8
## -1.2487  -0.0560   1.2533   0.0826   1.2205  -0.2076   3.7189   0.2706
##       9       10       11       12
## -0.9243  -0.0858  -0.8509   0.2052
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.68192    0.09911   6.880 5.97e-12 ***
## deptB       -0.04340    0.10984  -0.395    0.693
## deptC       -1.26260    0.10663 -11.841  < 2e-16 ***
## deptD       -1.29461    0.10582 -12.234  < 2e-16 ***
## deptE       -1.73931    0.12611 -13.792  < 2e-16 ***
```

```
## deptF       -3.30648    0.16998 -19.452  < 2e-16 ***
## genderMale  -0.09987    0.08085  -1.235    0.217
## ---
## 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: 103.14
##
## Number of Fisher Scoring iterations: 4
```