

Duration of Unemployment - Different Codings of Covariables

February 1, 2012

The unemployment data represent a contingency table with rows referring to gender and columns to duration of unemployment.

```
> unemployment <- matrix(c(403, 238, 167, 175), nrow=2, ncol=2)
> rownames(unemployment) <- c("male", "female")
> colnames(unemployment) <- c("<6 month", ">6 month")
> unemployment
```

| | <6 month | >6 month |
|--------|----------|----------|
| male | 403 | 167 |
| female | 238 | 175 |

```
> rowSums(unemployment)
```

| male | female |
|------|--------|
| 570 | 413 |

Calculation of odds and log-odds.

```
> ( odds_m <- 403/167 )
```

```
[1] 2.413174
```

```
> ( odds_w <- 238/175 )
```

```
[1] 1.36
```

```
> ( log_odds_m <- log(403/167) )
```

```
[1] 0.8809427
```

```
> ( log_odds_w <- log(238/175) )
```

```
[1] 0.3074847
```

For the fitting of a logit-model an alternative dataset is generated. First (0-1)-coding is considered

```
> gender <- c(rep(1, 403+167), rep(0, 238+175))
> unemp <- c(rep(1, 403), rep(0, 167), rep(1, 238), rep(0, 175))
```

For control, one can compute the crosstabulation of the generated data.

```
> table(gender, unemp)
```

```
      unemp
gender  0   1
  0 175 238
  1 167 403
```

Fit of a logit model.

```
> bin <- glm(unemp ~ gender, family=binomial)
> summary(bin)
```

Call:

```
glm(formula = unemp ~ gender, family = binomial)
```

Deviance Residuals:

```
      Min       1Q   Median       3Q      Max
-1.5669  -1.3105   0.8327   0.8327   1.0499
```

Coefficients:

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.30748    0.09958   3.088 0.00202 **
gender        0.57346    0.13559   4.229 2.34e-05 ***
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

```
Null deviance: 1270.3  on 982  degrees of freedom
Residual deviance: 1252.4  on 981  degrees of freedom
AIC: 1256.4
```

Number of Fisher Scoring iterations: 4

```
> bin$coef
```

```
(Intercept)      gender
  0.3074847    0.5734580
```

```
> exp(bin$coef)
```

```
(Intercept)      gender
  1.360000    1.774392
```

Now a dataset in effect-coding is created.

```
> gender_effect <- c(rep(1, 403+167), rep(-1,238+175))
```

For control, one can compute the crosstabulation of the generated data.

```
> table(gender_effect, unemp)
```

```

      unemp
gender_effect  0  1
             -1 175 238
              1  167 403

```

Fit a logit model.

```

> bin_effect <- glm(unemp ~ gender_effect, family=binomial)
> summary(bin_effect)

```

Call:

```
glm(formula = unemp ~ gender_effect, family = binomial)
```

Deviance Residuals:

```

      Min       1Q   Median       3Q      Max
-1.5669  -1.3105   0.8327   0.8327   1.0499

```

Coefficients:

```

              Estimate Std. Error z value Pr(>|z|)
(Intercept)    0.5942     0.0678   8.765 < 2e-16 ***
gender_effect  0.2867     0.0678   4.229 2.34e-05 ***
---

```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

```

      Null deviance: 1270.3  on 982  degrees of freedom
Residual deviance: 1252.4  on 981  degrees of freedom
AIC: 1256.4

```

Number of Fisher Scoring iterations: 4

```
> bin_effect$coef
```

```

(Intercept) gender_effect
  0.5942137   0.2867290

```

```
> exp(bin_effect$coef)
```

```

(Intercept) gender_effect
  1.811606   1.332063

```

Now we consider education level as explanatory variable.

```

> unemp_level <- matrix(c(202, 307, 87, 45,
+                          96, 162, 66, 18), nrow=4, ncol=2)
> colnames(unemp_level) <- c("Short term", "Long term")
> unemp_level

```

```

      Short term Long term
[1,]        202        96
[2,]        307       162
[3,]         87        66
[4,]         45        18

```

```
> rowSums(unemp_level)
```

```
[1] 298 469 153 63
```

For the fitting of a logit-model a new dataset is generated. First (0-1)-coding is considered.

```
> level <- factor(c(rep(1, 202+96), rep(2,307+162), rep(3,87+66), rep(4,45+18)))
> unemp_l <- c(rep(1, 202), rep(0, 96), rep(1, 307), rep(0, 162),
+             rep(1, 87), rep(0, 66), rep(1, 45), rep(0, 18))
```

For control, one can compute the crosstabulation of the generated data.

```
> table(level, unemp_l)
```

```
      unemp_l
level  0    1
1     96 202
2    162 307
3     66  87
4     18  45
```

Fit a logit model on the data. Define the variable level as a factor with the reference category 4.

```
> level <- relevel(level, ref=4)
> bin_l <- glm(unemp_l ~ level, family=binomial)
> summary(bin_l)
```

Call:

```
glm(formula = unemp_l ~ level, family = binomial)
```

Deviance Residuals:

| | Min | 1Q | Median | 3Q | Max |
|--|---------|---------|--------|--------|--------|
| | -1.5829 | -1.4581 | 0.8819 | 0.9206 | 1.0626 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|------------|
| (Intercept) | 0.9163 | 0.2789 | 3.286 | 0.00102 ** |
| level1 | -0.1724 | 0.3052 | -0.565 | 0.57222 |
| level2 | -0.2770 | 0.2953 | -0.938 | 0.34818 |
| level3 | -0.6400 | 0.3231 | -1.981 | 0.04763 * |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 1270.3 on 982 degrees of freedom
Residual deviance: 1263.8 on 979 degrees of freedom
AIC: 1271.8

Number of Fisher Scoring iterations: 4

Now additionally quasi-variances can be computed. Therefore the function "qvcalc" from the "qvcalc"-library is used.

```
> library(qvcalc)
> qv<-qvcalc(bin_1,"level")
> summary(qv)
```

```
Model call: glm(formula = unemp_l ~ level, family = binomial)
Factor name: level
```

| | estimate | SE | quasiSE | quasiVar |
|---|------------|-----------|------------|-------------|
| 4 | 0.0000000 | 0.0000000 | 0.27888650 | 0.077777678 |
| 1 | -0.1723712 | 0.3051964 | 0.12396432 | 0.015367154 |
| 2 | -0.2770393 | 0.2953097 | 0.09710904 | 0.009430166 |
| 3 | -0.6400374 | 0.3231462 | 0.16323531 | 0.026645768 |

```
Worst relative errors in SEs of simple contrasts (%): 0 0
```

```
Worst relative errors over *all* contrasts (%): 0 0
```

```
> plot(qv)
```

