Contrasting Contrasts

Livio Finos

18 / 11 / 2019

Contents

NOTA la prima bozza di questo materiale è stato presentato il 18/11/2019 agli incontri regolari del gruppo di Psicostat http://ip146179.psy.unipd.it/psicostat/web.

Calendario incontri: http://ip146179.psy.unipd.it/psicostat/web/psicostat3.html

Ulteriore materiale didattico alla mia pagine https://livioivil.github.io/students/Teaching_material.html

Abstract Richiamo l'importanza dell'uso di contrasti a somma zero per le variabili categoriali (ad es i fattori di un disegno sperimentale) rispetto all'usuale codifica in variabili dummy. Il problema rimane analogo per le variabili quantitative. Affronto il problema con un dataset sintetico e un modello lineare con un fattore (= variabile categriale), una variabile quantitiva e la loro interazione.

1 Introduzione

Inizialmente pensavo di parlare dell'importanza dei contrasti ortogonali nei disegni sperimentali. Mentre iniziavo a lavorarci, capivo che in venti minuti non sarei riuscito a lasciare molto di più del profumo del problema.
Nella speranza che questo stimoli comunque l'appetito e lanci il branco alla caccia della sua preda.

2 The data + EDA

Challenge: Costruiamo un dataset dove sono noti gli effetti, riuscite ad analizzarlo in modo adeguato?

Il modello proposto è un semplice modello ANCOVA:
- risposta normale (lm) con varianza dei residui pari ad 1
- modello lineare con predittori: + due gruppi (`A` e `B`), + una variabile continua e + la loro interazione - Effetti: Intercetta e gruppo. La variabile continua non ha relazione con la risposta per il gruppo `A`, ce l'ha invece nel il gruppo `B` (interazione).

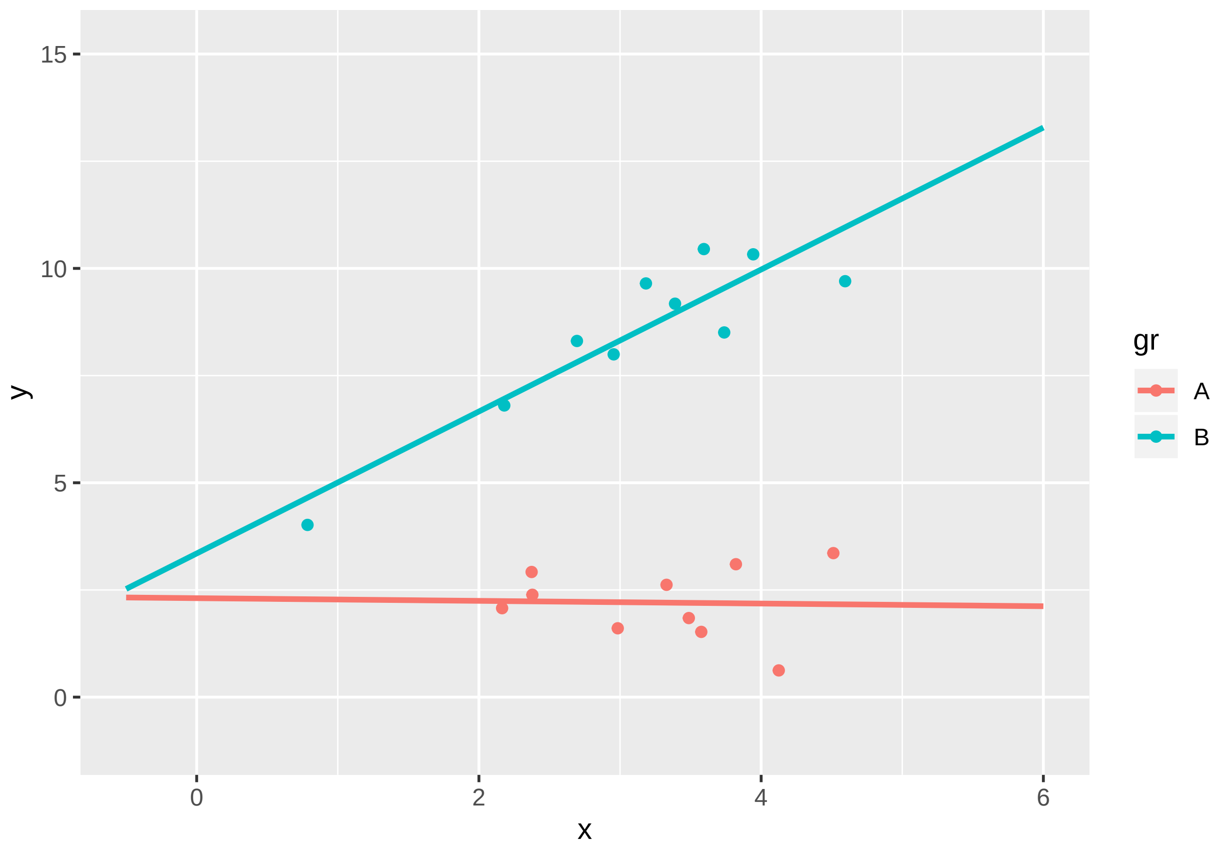
Questi sono i dati creati e la loro rappresentazione.

```
set.seed(1)
n0=10
D=data.frame(gr=as.factor(rep(LETTERS[1:2],n0)),
             x=rnorm(n0*2)+3)
mu=2+(D$gr=="B")*.5+D$x*(D$gr=="B")*2
D$y= mu+rnorm(n0*2)

D
```

```
##    gr         x          y
## 1   A 2.3735462  2.9189774
## 2   B 3.1836433  9.6494229
## 3   A 2.1643714  2.0745650
## 4   B 4.5952808  9.7012099
## 5   A 3.3295078  2.6198257
## 6   B 2.1795316  6.8029345
## 7   A 3.4874291  1.8442045
## 8   B 3.7383247  8.5058970
## 9   A 3.5757814  1.5218499
## 10  B 2.6946116  8.3071648
## 11  A 4.5117812  3.3586796
## 12  B 3.3898432  9.1768987
## 13  A 2.3787594  2.3876716
## 14  B 0.7853001  4.0167952
## 15  A 4.1249309  0.6229404
## 16  B 2.9550664  7.9951382
## 17  A 2.9838097  1.6057100
## 18  B 3.9438362 10.3283590
## 19  A 3.8212212  3.1000254
## 20  B 3.5939013 10.4509784
```

```
library(ggplot2)
ggplot(D,aes(x=x,y=y,color=gr))+geom_point()+
  geom_smooth(method = "lm", fill = NA,fullrange = TRUE)+xlim(-.5, 6)
```



3 Modelli lineari

3.1 Un modello lineare

```
modDU=lm(y~gr*x,data=D)
summary(modDU)
```

```
##
## Call:
## lm(formula = y ~ gr * x, data = D)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.55585 -0.61528 -0.00131  0.54234  1.19203
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.30818    1.24368   1.856  0.08198 .
## grB          1.04292    1.53659   0.679  0.50701
## x           -0.03137    0.37016  -0.085  0.93352
## grB:x        1.68703    0.46200   3.652  0.00215 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8778 on 16 degrees of freedom
## Multiple R-squared:  0.9481, Adjusted R-squared:  0.9384
## F-statistic: 97.49 on 3 and 16 DF,  p-value: 1.708e-10
```

Le variabili usate nel modello lineare

```r
(mm <- model.matrix(~gr*x,data=D))
```

```
##    (Intercept) grB         x     grB:x
## 1            1   0 2.3735462 0.0000000
## 2            1   1 3.1836433 3.1836433
## 3            1   0 2.1643714 0.0000000
## 4            1   1 4.5952808 4.5952808
## 5            1   0 3.3295078 0.0000000
## 6            1   1 2.1795316 2.1795316
## 7            1   0 3.4874291 0.0000000
## 8            1   1 3.7383247 3.7383247
## 9            1   0 3.5757814 0.0000000
## 10           1   1 2.6946116 2.6946116
## 11           1   0 4.5117812 0.0000000
## 12           1   1 3.3898432 3.3898432
## 13           1   0 2.3787594 0.0000000
## 14           1   1 0.7853001 0.7853001
## 15           1   0 4.1249309 0.0000000
## 16           1   1 2.9550664 2.9550664
## 17           1   0 2.9838097 0.0000000
## 18           1   1 3.9438362 3.9438362
## 19           1   0 3.8212212 0.0000000
## 20           1   1 3.5939013 3.5939013
## attr(,"assign")
## [1] 0 1 2 3
## attr(,"contrasts")
## attr(,"contrasts")$gr
## [1] "contr.treatment"
```

Notate le correlazioni tra predittori e il Multiple R-squared delle prime tre colonne per spiegare la colonna dell'interazione:

```r
cor(mm)
```

```
## Warning in cor(mm): the standard deviation is zero
```

```
##             (Intercept)         grB           x     grB:x
## (Intercept)           1          NA          NA        NA
## grB                  NA  1.00000000 -0.09503104 0.9094708
## x                    NA -0.09503104  1.00000000 0.2451776
## grB:x                NA  0.90947082  0.24517762 1.0000000
```

```r
summary(lm(mm[,2]~mm[,-2]+0))
```

```
##
## Call:
## lm(formula = mm[, 2] ~ mm[, -2] + 0)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.243736 -0.060709  0.005904  0.071867  0.265952
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## mm[, -2](Intercept)  0.65509    0.11529   5.682 2.70e-05 ***
## mm[, -2]x           -0.19006    0.03590  -5.294 5.95e-05 ***
## mm[, -2]grB:x        0.29060    0.01871  15.528 1.79e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1385 on 17 degrees of freedom
## Multiple R-squared:  0.9674, Adjusted R-squared:  0.9616
## F-statistic:   168 on 3 and 17 DF,  p-value: 7.853e-13
```

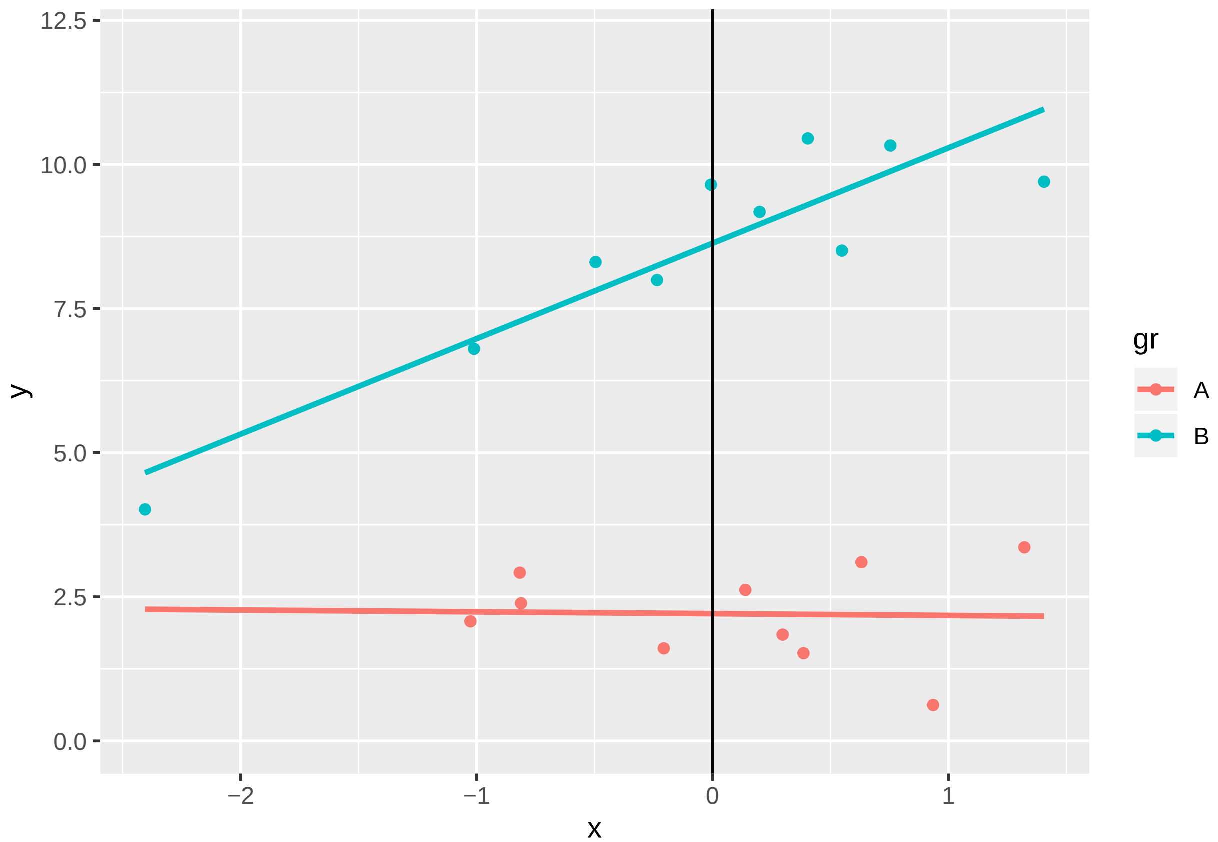
3.2 Un secondo modello lineare (x 0-centrata)

```r
D2=D
D2$x=D$x-mean(D$x)
modDUC=lm(y~gr*x,data=D2)
summary(modDUC)
```

```
##
## Call:
## lm(formula = y ~ gr * x, data = D2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.55585 -0.61528 -0.00131  0.54234  1.19203
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.20810    0.27933   7.905 6.47e-07 ***
## grB          6.42543    0.39449  16.288 2.21e-11 ***
## x           -0.03137    0.37016  -0.085  0.93352
## grB:x        1.68703    0.46200   3.652  0.00215 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8778 on 16 degrees of freedom
## Multiple R-squared:  0.9481, Adjusted R-squared:  0.9384
## F-statistic: 97.49 on 3 and 16 DF,  p-value: 1.708e-10
```

Osservate la scala lo 0 nella scala delle ascisse: è chiaro che la differenza nei due gruppi c'è.

```r
ggplot(D2,aes(x=x,y=y,color=gr))+geom_point()+
  geom_smooth(method = "lm", fill = NA,fullrange = TRUE)+ geom_vline(xintercept = 0)
```



NOTA Notate che il test F (e i vari R-squares) sono identici al primo modello (e così sarà per tutti i successivi).

Correlazioni tra predittori

E' utile anche valutare le correlazioni tra i predittori.

I predittori:

```r
(mm <- model.matrix(~gr*x,data=D2))
```

```
##    (Intercept) grB            x        grB:x
## 1            1   0 -0.816977687  0.000000000
## 2            1   1 -0.006880552 -0.006880552
## 3            1   0 -1.026152489  0.000000000
## 4            1   1  1.404756926  1.404756926
## 5            1   0  0.138983896  0.000000000
## 6            1   1 -1.010992260 -1.010992260
## 7            1   0  0.296905176  0.000000000
## 8            1   1  0.547800829  0.547800829
## 9            1   0  0.385257475  0.000000000
## 10           1   1 -0.495912263 -0.495912263
## 11           1   0  1.321257292  0.000000000
## 12           1   1  0.199319360  0.199319360
```

```
## 13            1   0 -0.811764457  0.000000000
## 14            1   1 -2.405223763 -2.405223763
## 15            1   0  0.934407042  0.000000000
## 16            1   1 -0.235457485 -0.235457485
## 17            1   0 -0.206714139  0.000000000
## 18            1   1  0.753312335  0.753312335
## 19            1   0  0.630697319  0.000000000
## 20            1   1  0.403377445  0.403377445
## attr(,"assign")
## [1] 0 1 2 3
## attr(,"contrasts")
## attr(,"contrasts")$gr
## [1] "contr.treatment"
```

Notate la correlazione tra predittori e il Multiple R-squared delle prime tre colonne per spiegare la colonna dell'interazione:

```r
cor(mm)
```

```
## Warning in cor(mm): the standard deviation is zero
```

```
##             (Intercept)         grB           x       grB:x
## (Intercept)           1          NA          NA          NA
## grB                  NA  1.00000000 -0.09503104 -0.05946962
## x                    NA -0.09503104  1.00000000  0.80181388
## grB:x                NA -0.05946962  0.80181388  1.00000000
```

```r
summary(lm(mm[,2]~mm[,-2]+0))
```

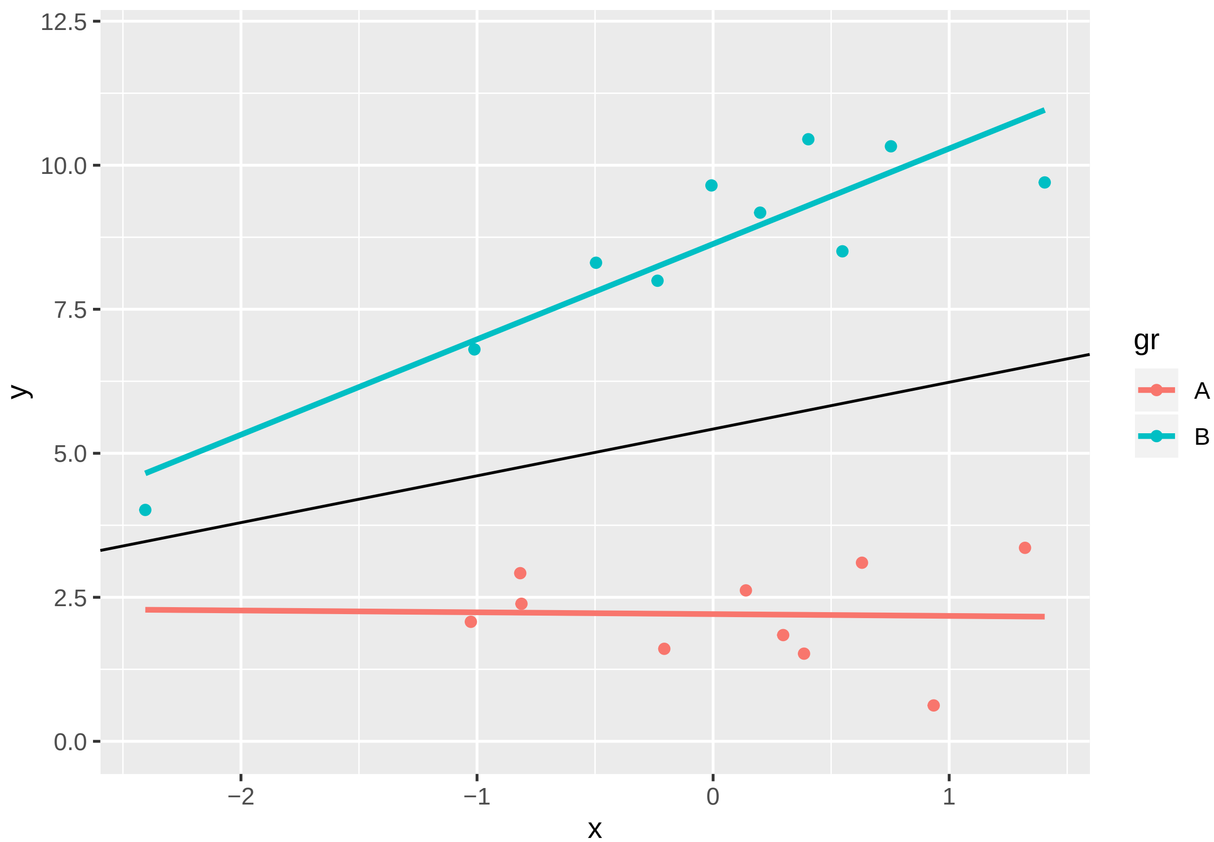
```
## 
## Call:
## lm(formula = mm[, 2] ~ mm[, -2] + 0)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.57782 -0.48222 -0.00215  0.50046  0.55697 
## 
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)    
## mm[, -2](Intercept)  0.50139    0.12127   4.135 0.000693 ***
## mm[, -2]x           -0.07448    0.22686  -0.328 0.746694    
## mm[, -2]grB:x        0.03293    0.28393   0.116 0.909022    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.5397 on 17 degrees of freedom
## Multiple R-squared:  0.5049, Adjusted R-squared:  0.4175 
## F-statistic: 5.779 on 3 and 17 DF,  p-value: 0.006501
```

3.3 ... e un terzo (x 0-centrata + gr 0-centrata)

```r
D3=D2
contrasts(D3$gr)=contr.sum(2)
modS0=lm(y~gr*x,data=D3)
summary(modS0)
```

```
##
## Call:
## lm(formula = y ~ gr * x, data = D3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.55585 -0.61528 -0.00131  0.54234  1.19203
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.4208     0.1972  27.483 6.80e-15 ***
## gr1          -3.2127     0.1972 -16.288 2.21e-11 ***
## x             0.8121     0.2310   3.516  0.00287 **
## gr1:x        -0.8435     0.2310  -3.652  0.00215 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8778 on 16 degrees of freedom
## Multiple R-squared:  0.9481, Adjusted R-squared:  0.9384
## F-statistic: 97.49 on 3 and 16 DF,  p-value: 1.708e-10
```

```r
ggplot(D3,aes(x=x,y=y,color=gr))+geom_point()+
  geom_smooth(method = "lm", fill = NA,fullrange = TRUE)+ geom_abline(intercept = coef(modS0)[1], slope
```



La linea nera nel grafico rappresenta l'equazione: `Y=` 5.4208153 + 0.8121474 `X` come stimati dal modello. In effetti questo è l'effetto per i soggetti con `gr==0` che non esistono nella verà, ma rappresentano un valore intermedio (in qualche modo nullo); sono quindi una stima *al netto di* `gr`.

Osserviamo ora come è cambiata la matrice dei predittori (in particolare l'interazione):

```
(mm <- model.matrix(~gr*x,data=D3))
```

```
##    (Intercept) gr1            x        gr1:x
## 1            1   1 -0.816977687 -0.816977687
## 2            1  -1 -0.006880552  0.006880552
## 3            1   1 -1.026152489 -1.026152489
## 4            1  -1  1.404756926 -1.404756926
## 5            1   1  0.138983896  0.138983896
## 6            1  -1 -1.010992260  1.010992260
## 7            1   1  0.296905176  0.296905176
## 8            1  -1  0.547800829 -0.547800829
## 9            1   1  0.385257475  0.385257475
## 10           1  -1 -0.495912263  0.495912263
## 11           1   1  1.321257292  1.321257292
## 12           1  -1  0.199319360 -0.199319360
## 13           1   1 -0.811764457 -0.811764457
## 14           1  -1 -2.405223763  2.405223763
## 15           1   1  0.934407042  0.934407042
## 16           1  -1 -0.235457485  0.235457485
## 17           1   1 -0.206714139 -0.206714139
```

```
## 18           1  -1  0.753312335 -0.753312335
## 19           1   1  0.630697319  0.630697319
## 20           1  -1  0.403377445 -0.403377445
## attr(,"assign")
## [1] 0 1 2 3
## attr(,"contrasts")
## attr(,"contrasts")$gr
##   [,1]
## A    1
## B   -1
```

Notate il Multiple R-squared delle prime tre colonne per spiegare la colonna dell'interazione:

```r
summary(lm(mm[,2]~mm[,-2]+0))
```

```
##
## Call:
## lm(formula = mm[, 2] ~ mm[, -2] + 0)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.11394 -1.00093  0.00431  0.96444  1.15564
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## mm[, -2](Intercept) -0.002786   0.242535  -0.011    0.991
## mm[, -2]x            0.116024   0.282650   0.410    0.687
## mm[, -2]gr1:x        0.032933   0.283935   0.116    0.909
##
## Residual standard error: 1.079 on 17 degrees of freedom
## Multiple R-squared:  0.009814,   Adjusted R-squared:  -0.1649
## F-statistic: 0.05617 on 3 and 17 DF,  p-value: 0.9819
```

4 Una simulazione

Cosa mi direbbero i tre modelli se potessi ripetere molte volte l'esperimento?

```r
res_sim=replicate(1000,
          {
            D$y <- D2$y <- D3$y <- mu+rnorm(n0*2)
            modDU=lm(y~gr*x,data=D)
            p_DU=summary(modDU)$coeff[,4]
            modDUC=lm(y~gr*x,data=D2)
            p_DUC=summary(modDUC)$coeff[,4]
            modS0=lm(y~gr*x,data=D3)
            p_S0=summary(modS0)$coeff[,4]
            c(DU=p_DU,DUC=p_DUC,S0=p_S0)
          })

res_sim=t(res_sim)
```

Potenza stimata:

```r
library(r41sqrt10)
## modello Dummy
summaryResSim(res_sim[,1:4])
```

```
##                  <=0.01 <=0.05 <=0.1 <=0.5 <=0.75
## DU.(Intercept)    0.095  0.244 0.364 0.773  0.904
## DU.grB            0.012  0.046 0.098 0.517  0.742
## DU.x              0.006  0.043 0.100 0.496  0.747
## DU.grB:x          0.783  0.940 0.974 0.999  1.000
```

```r
## modello Dummy +  Centered X
summaryResSim(res_sim[,5:8])
```

```
##                   <=0.01 <=0.05 <=0.1 <=0.5 <=0.75
## DUC.(Intercept)    0.997  1.000 1.000 1.000  1.000
## DUC.grB            1.000  1.000 1.000 1.000  1.000
## DUC.x              0.006  0.043 0.100 0.496  0.747
## DUC.grB:x          0.783  0.940 0.974 0.999  1.000
```

```r
## modello S0 +  Centered X
summaryResSim(res_sim[,9:12])
```

```
##                  <=0.01 <=0.05 <=0.1 <=0.5 <=0.75
## S0.(Intercept)    1.000  1.000 1.000 1.000      1
## S0.gr1            1.000  1.000 1.000 1.000      1
## S0.x              0.781  0.934 0.971 0.997      1
## S0.gr1:x          0.783  0.940 0.974 0.999      1
```

```r
# prova anche con
# D3=D2
# contrasts(D3$gr)=contr.sum(2)
# modS0=lm(y~gr*x,data=D3)
# X non centrata e contr.sum(2)
```

APPROFONDIMENTO abbiamo supposto un effetto più piccolo per `gr` rispetto all'effetto della sua interazione con `x`. La potenza di `gr` però risulta maggiore. Questa apparente contraddizione si dissolve considerando che la potenza di un test dipende dalla varianza dello stimatore su cui si basa. In questo caso la varianza dello stimatore dell'interazione è molto maggiore di quella dell'effetto gruppo. Ma questo merita un ulteriore approfondimento e non verrà trattato qui.

5 Conclusioni

La definizione di un effetto dipende in modo cruciale dal modo in cui codifichiamo le variabili (fattori o continue che siano).

La loro stima dipende in modo cruciale dalla correlazione tra i predittori. Questo aspetto è cruciale soprattutto nelle interazioni, dove le correlazioni sono spesso elevate per natura (sono definite come prodotto delle colonne del disegno sperimentale).

Quando è possibile (e sensato), la raccomandazione è quella di centrare i contrasti intorno allo 0 (vedi uso di `contr.sum()`).