# Statistical Methods and Data Analysis in Developmental Psychology

Exam Simulation

Prof. Antonio Calcagnì, Dr. Filippo Gambarota

The dataset `anxiety.rda` contains data about adolescents with or without anxiety disorders. The dataset contains the following variables:

- `anx`: having (1) or not (0) the anxiety disorder
- `selfesteem`: the self-esteem score from 0 (low self-esteem) to 10 (high self-esteem)
- `socialnetwork`: the estimated size of the social network from 0 (very small) to 100 (big)
- `age`: the age in years
- `family`: If the family had ("yes") or not ("no") history of anxiety disorders

The dataset can be loaded using `load()`. Make sure to check the dataset structure and whether categorical variables are interpreted as factors from R.

## 1 Problem: Identify the number of statistical units n and the type of variables

a) 190 observations, 2 categorical variables and 2 numeric variables
b) 200 observations, 1 categorical variables and 4 numeric variables
c) 205 observations, 1 categorical variables and 4 numeric variables
d) 200 observations, 5 categorical variables and 0 numeric variables

**Solution**: **b**

## 2 Problem: Make an appropriate plot of univariate distributions of predictors response variable

```r
par(mfrow = c(2,3)) # we have 5 variables

# numerical variables
boxplot(anxiety$selfesteem, main = "Self Esteem") # or histogram
boxplot(anxiety$socialnetwork, main = "Social Network") # or histogram
boxplot(anxiety$age, main = "Social Network") # or histogram

# categorical variables
barplot(table(anxiety$anx), main = "anxiety")
barplot(table(anxiety$family), main = "Family History anxiety")
```

# 3 Problem: Calculate from observed data the odds ratio of having anxiety as a function of family anxiety history

```
pfamily_yes <- mean(anxiety$anx[anxiety$family == "yes"])
pfamily_no <- mean(anxiety$anx[anxiety$family == "no"])

(pfamily_yes / (1 - pfamily_yes)) / (pfamily_no / (1 - pfamily_no))
```

```
## [1] 2.501385
```

# 4 Problem: Define and fit an appropriate additive model to predict the probability of having anxiety as a function of `selfesteem` and `family`. Intepret the results.

```
fit <- glm(anx ~ family + selfesteem, family = binomial(link = "logit"), data = anxiety)
summary(fit)
```

```
## 
## Call:
## glm(formula = anx ~ family + selfesteem, family = binomial(link = "logit"), 
##     data = anxiety)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.2923  -0.5675  -0.4407  -0.2754   2.7491  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept)   1.2714     0.7453   1.706   0.0880 .  
## familyyes     0.9697     0.4278   2.267   0.0234 *  
## selfesteem   -0.5027     0.1202  -4.181  2.9e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 179.15  on 199  degrees of freedom
## Residual deviance: 153.23  on 197  degrees of freedom
## AIC: 159.23
## 
## Number of Fisher Scoring iterations: 5
```

The `(Intercept)` ($\beta_{(Intercept)} = 1.271$, $SE = 0.745$, $z = 0.745$, $p = 0.088$) is the log odds of having anxiety for adolescents without family history of anxiety and self esteem equal to 0. Converting to probabilities using the inverse of the link function (`plogis()`) the expected probability of anxiety is 0.7809878.

The `familyyes`($\beta_{familyyes} = 0.970$, $SE = 0.428$, $z = 0.428$, $p = 0.023$) is the log odds ratio comparing the odds of having anxiety for people with and without family history, controlling for other predictors. Converting to the probability scale using `exp()` the odds ratio suggest that people with familiarity in anxiety disorders (compared to no familiarity) have 2.6371868 the odds of having anxiety.

The `selfesteem` ($\beta_{selfesteem} = -0.503$, $SE = 0.120$, $z = 0.120$, $p =< 0.001$) is the increase in the log odds of having anxiety disorders for a unit increase in self esteem, controlling for other predictors. Converting to the probability scale, for a unit increase in self esteem the odds of having anxiety disorders decrease by a factor of 0.6048933.

In summary, having familiarity of anxiety disorders significantly increase the probability of having anxiety and as the self esteem increase, the probability of having anxiety decrease.

# 5 Problem: From the fitted model, find the probability and the 95% confidence interval that a subject without anxiety familiarity and self esteem = 3 has axiety disorders.

We need to use the `predict()` function, extract the expected value and the standard error on the logit scale, calculate the confidence interval using the quantiles of the normal distribution and apply the inverse logit function.

```
# dataset for predictions
prs <- data.frame(
    family = "no",
    selfesteem = 3
)

preds <- predict(fit, prs, se.fit = TRUE)
ci <- plogis(preds$fit + qnorm(c(0.025, 0.975)) * preds$se.fit)

c(p = plogis(preds$fit), lower95 = ci[1], upper95 = ci[2])
```

```
##       p.1   lower95   upper95
## 0.4411053 0.2551370 0.6452088
```

# 6 Problem: fit a model including also the social network effect and intepret the parameters of numerical predictors using the divide by 4 rule.

```
fit2 <- update(fit, . ~ . + socialnetwork)
coefs <- coef(fit2)[c("selfesteem", "socialnetwork")] # only numerical variables
coefs / 4
```

```
##    selfesteem socialnetwork
##   -0.08168504   -0.00941893
```

The divide-by-4 rule is a way to quickly estimate the maximal difference in probability for a unit-increase of predictors. Thus the maximal effect of `selfesteem` is -0.081685 and the maximal effect of `socialnetwork` is -0.081685

# 7 Problem: perform a statistical test to compare the residual deviance of the null model, model with and model without the `socialnetwork` predictor and intepret the result.

```
fit0 <- glm(anx ~ 1, family = binomial(link = "logit"), data = anxiety)
anova(fit0, fit, fit2, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: anx ~ 1
## Model 2: anx ~ family + selfesteem
## Model 3: anx ~ family + selfesteem + socialnetwork
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       199     179.15
## 2       197     153.23  2  25.9218 2.35e-06 ***
## 3       196     147.94  1   5.2911  0.02143 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

The test to compare the residual deviance of two models is the likelihood ratio test and can be performed in R using the `anova(test = "LRT")` function. The ratio between deviances under the null hypothesis is distributed as a $\chi^2$ with *df* the difference between the residual degrees of freedom of the models under comparison. The reduction in deviance for `fit` is 25.9217991 and the p value is $2.3504599 \times 10^{-6}$. When including `socialnetwork` the further reduction in deviance is 5.2911195 and the p value is 0.0214344. Thus including `socialnetwork` significantly reduce the residual deviance.

# 8 Problem: extract the DFBETAs value from the model created in the previous step. Identify (if any) problematic observations (i.e., marked as outlier in all coefficients but the Intercept) using a cut-off of $2/\sqrt{n}$ ($n$ is the sample size) and intepret the results. In case of problematic observations re-fit the model without that observations and comment the results.

```
cutoff <- 2 / sqrt(nrow(anxiety))
infl <- infl_measure(fit2) # from utils-glm.R
is_out <- abs(infl[, 2]) > cutoff & abs(infl[, 3]) > cutoff & abs(infl[, 4]) > cutoff # outlier in all

anxiety[is_out, ]
```

```
##     anx selfesteem socialnetwork age family
## 36    1          6            50  15    yes
## 140   1          5            42  13    yes
## 191   1          3            35  16    yes
```

```r
fit_no_out <- update(fit2, . ~ ., data = anxiety[!is_out, ])

car::compareCoefs(fit2, fit_no_out, pvals = TRUE)
```

```
## Calls:
## 1: glm(formula = anx ~ family + selfesteem + socialnetwork, family = binomial(link = "logit"),
##   data = anxiety)
## 2: glm(formula = anx ~ family + selfesteem + socialnetwork, family = binomial(link = "logit"),
##   data = anxiety[!is_out, ])
##
##               Model 1 Model 2
## (Intercept)     1.404   1.203
## SE              0.745   0.760
## Pr(>|z|)       0.0593  0.1135
##
## familyyes       1.078   0.819
## SE              0.442   0.465
## Pr(>|z|)       0.0147  0.0781
##
## selfesteem     -0.327  -0.244
## SE              0.138   0.143
## Pr(>|z|)       0.0180  0.0886
##
## socialnetwork -0.0377 -0.0476
## SE             0.0168  0.0179
## Pr(>|z|)       0.0252  0.0078
##
```

DFBETAs quantifies the impact of removing a single observation in the estimated coefficients and standard errors. Observation with high DFBETA suggest that removing that observation impact the estimation of the parameter. Firstly we extract the DFBETAs using the `infl_measure()` from the `utils-glm.R` file. Then we compare each observation across coefficients and we mark the observation as problematic if the absolute value is greater than the cutoff in all coefficients (no Intercept). Then we fitted the model with and without that outliers assessing the estimates and p values. Clearly, removing the observations has a great impact on estimates and p values because the `family` and `selfesteem` effects are no longer statistically significant.

# 9 Problem: The classification accuracy of the model created at Problem 6 (`anx ~ family + selfesteem + socialnetwork`) is:

a) 0
b) 0.65
c) 33
d) 0.85

**Solution**: d

```r
1 - error_rate(fit2) # from utils-glm.R
```

```
## [1] 0.85
```

```r
# or manually
pi <- ifelse(predict(fit2, type = "response") > 0.5, 1, 0)
yi <- anxiety$anx

mean((yi == 1 & pi == 1) | (yi == 0 & pi == 0))
```

```
## [1] 0.85
```

The classification accuracy is defined as the proportions of correct classifications of a logistic regression model using a threshold of 0.5. The `error_rate()` function compute the proportion of incorrect classifications.

# 10 Problem: Fit the same model considered in the previous problem but using the median-centered self-esteem. How parameters intepretation change?

```r
anxiety$selfesteem0 <- anxiety$selfesteem - median(anxiety$selfesteem)

fit2_cen <- glm(anx ~ family + selfesteem0 + socialnetwork,
                family = binomial(link = "logit"), data = anxiety)
summary(fit2_cen)
```

```
##
## Call:
## glm(formula = anx ~ family + selfesteem0 + socialnetwork, family = binomial(link = "logit"),
##     data = anxiety)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2900  -0.5728  -0.4002  -0.2228   2.7145
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -0.88308    0.64231  -1.375   0.1692
## familyyes      1.07833    0.44182   2.441   0.0147 *
## selfesteem0   -0.32674    0.13808  -2.366   0.0180 *
## socialnetwork -0.03768    0.01683  -2.239   0.0252 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 179.15  on 199  degrees of freedom
## Residual deviance: 147.93  on 196  degrees of freedom
## AIC: 155.93
##
## Number of Fisher Scoring iterations: 5
```

```
car::compareCoefs(fit2, fit2_cen)
```

```
## Calls:
## 1: glm(formula = anx ~ family + selfesteem + socialnetwork, family = binomial(link = "logit"),
##   data = anxiety)
## 2: glm(formula = anx ~ family + selfesteem0 + socialnetwork, family = binomial(link = "logit"),
##    data = anxiety)
##
##               Model 1 Model 2
## (Intercept)     1.404  -0.883
## SE              0.745   0.642
##
## familyyes       1.078   1.078
## SE              0.442   0.442
##
## selfesteem     -0.327
## SE              0.138
##
## socialnetwork -0.0377 -0.0377
## SE             0.0168  0.0168
##
## selfesteem0            -0.327
## SE                      0.138
##
```

Median centering does not affect the overall model fit or regression coefficients but only the Intercept. Now the Intercept $\beta_{(Intercept)} = -0.883$, $SE = 0.642$, $z = 0.642$, $p = 0.169$ is the expected log odds of having anxiety for people without familiarity, with social network 0 and a median value of self-esteem.