

# ICC1 and ICC2 for binary data using rptR

Narayan Sharma

5 December 2017

# Repeatability

- ▶ Repeatability relates to the way of quantifying the *reliability* of measurements
- ▶ Offers insights into the components contributing to *variability* in the data
- ▶ Describes the variance into within-group and between-group sources of variance and is more generally referred to as the *intra-class correlation* (ICC)

# Introduction $R$

- ▶ Estimating repeatabilities is the mixed effects model framework
- ▶  $R$  is calculated as the variance among group means *group-level variance*  $V_G$  over the sum of group-level and
- ▶ Data-level *residual variance*  $V_G$
- ▶  $R = V_G / (V_G + V_R)$

# Introduction $R$ (ICC1)

- ▶ Normally distributed data and linear models

$$R = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_e^2}$$

- ▶ Binary data and generalised linear models

$$R = \frac{\sigma_u^2}{\sigma_u^2 + \pi^2/3}$$

- ▶ ICC1 ranges between 0 and 1
- ▶ If ICC1 is higher than 0.05, there is variation (Snijders & Bosker, 2012)

# Introduction $R$ (ICC2)

- ▶ A useful index of the quality of the test with the number of items and the average correlation is the Signal/Noise ratio  $\lambda_j$
- ▶ Normally

$$\lambda_j = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_e^2/k}$$

- ▶  $k$  = number of units or groups
- ▶ Binary data and generalised linear models

$$\lambda_j = \frac{\sigma_u^2}{\sigma_u^2 + \pi^2/3 * (1/k)}$$

# GLMM-based Repeatability Estimation for Binary Data

Estimates repeatability from a generalized linear mixed-effects models fitted by restricted maximum likelihood (REML)

- Usage

```
rptBinary(BV~ (1|groups), grname, data, link = c("logit", "probit"),  
CI = 0.95, nboot = 1000, npermut = 0, parallel = FALSE, ncores  
= NULL, ratio = TRUE, adjusted = TRUE, expect = "meanobs",  
rptObj = NULL, update = FALSE)
```

# Arguments

- ▶ `gname` = A character string or vector of character strings giving the name(s) of the grouping factor(s)
- ▶ `data` = A dataframe that contains the variables
- ▶ `link` = Link function. logit and probit are allowed
- ▶ `nboot` = Number of parametric bootstraps for interval estimation
- ▶ `npermut` = Number of permutations used when calculating asymptotic p-values
- ▶ `parallel` = Boolean to express if parallel computing should be applied
- ▶ `ncores` = Specifying the number of cores to use for parallelization
- ▶ `ratio` = Boolean to express if variances or ratios of variance should be estimated
- ▶ `adjusted` = Boolean to express if adjusted or unadjusted repeatabilities should be estimated.

# Arguments

- ▶ `expect` = A character string specifying the method for estimating the expectation in Poisson models with log link and in Binomial models with logit link (in all other cases the argument is ignored), default is “meanobs”,
- ▶ `expect` = “liability” for binary cases to estimate the distribution specific variance
- ▶ `rptObj` = The output of a `rptR` function. Can be specified in combination with `update` = TRUE to update bootstraps and permutations
- ▶ `update` = If TRUE, the `rpt` object to be updated has to be inputted with the `rptObj` argument



## Example from BeetlesMale dataset

##	Population	Container	Sex	Habitat	Treatment	Colour
## 1	1	2	Male	A	Cont	0
## 2	1	2	Male	A	Exp	0
## 3	1	2	Male	A	Cont	0
## 4	1	2	Male	A	Exp	1
## 5	1	2	Male	B	Cont	0
## 6	1	2	Male	B	Exp	0

# ICC1 with CI

```
beetle.ratio<-rptBinary(Colour ~ (1|Population),  
  grname=c("Population"),data=BeetlesMale, nboot=100,  
  npermut=0, expect = "liability",ratio = T)
```

- ▶ beetle.ratio\$R # gives ICC1 with link scale
- ▶ beetle.ratio\$CI\_emp[[2]][1] # Lower
- ▶ beetle.ratio\$CI\_emp[[2]][2] # Upper

```
## Bootstrap Progress:
```

```
##           Population  
## R_org      0.1858041  
## R_link     0.2232949  
  
##           2.5%  
## Population 0.05066133  
  
##           97.5%  
## Population 0.3637154
```

## ICC2 with CI

```
beetle.ratio.var<-rptBinary(Colour ~ (1|Population),  
grname=c("Population"),data=BeetlesMale, nboot=100,  
npermut=0, expect = "liability",ratio = F)
```

- ▶ `k=12` # numbers of groups/units
- ▶ `round(ab/(ab+ (pi^2/3*(1/k))), 3)# icc2`
- ▶ `round(ab_l/(ab_l+ (pi^2/3*(1/k))), 3) # icc2_lower`
- ▶ `round(ab_u/(ab_u+ (pi^2/3*(1/k))), 3) # icc2_upper`

```
## Bootstrap Progress:
```

```
## [1] 0.775
```

```
##           2.5%
```

```
## Population 0.518
```

```
##           97.5%
```

```
## Population 0.881
```

## ICC1 and ICC2 manually

```
fit1<-glmer(Colour ~ (1|Population), BeetlesMale,  
family="binomial")
```

- ▶ `vc <- VarCorr(fit1)`
- ▶ `v<-vc$Population[1,1]`
- ▶ `icc<- v/(v+ (pi^2/3))`
- ▶ `icc2<- v/(v+(pi^2/3*(1/k)))# ICC2`

```
## [1] 0.2232949
```

```
## [1] 0.7752748
```



**THANK YOU**  
for your  
**ATTENTION!**