Formulaire R – LBRAI2222

**read.csv(“**mycsvsheet.csv**”)**  To read csv sheets

**head(**mydata**)** To visualize my data.frame.

**table(**mydata[,”factors of interest”]**)** Build a contingency table of the counts at each combination of factor levels.

**pander(summary(**mydata**))** To visualize data: min, max, mean, 1st Qu., median, 3rd Qu. or class and mode if categorial

**ggplot()** To plot data

**+ geom\_point()**

**+ labs()**

mod **<- lm(**Y ~ a \* b, mydata**)** To create a GLM with all interactions

**anova(**mod**)** To create an ANOVA with a GLM model

mod **<- lme4::lmer(**Y **~** 1 **+ (1|**a**) + (1|**b:a**),** mydata**)**

To create a linear mixed model. Random effects are in “()”. Here in the example, b is hierarchized to a (“b:a”).

**summary(**mod**)** To get the estimations of each term of a GLM model

**plot(**mod**)** To check ANOVA hypothesis (Residuals vs Fitted ; Normal Q-Q plot ; Scale-location ; Residuals vs Leverage)

**predict(**mod, **newdata** **=** xpred, **interval =** ""**)** To predict values from a GLM model (xpred = data.frame with the values we want to predict)

**pander(model.matrix(**mod**))** To get the X matrix used by R for the GLM model

mydata$myfactor **<- as.factor(**mydata$myfactor**)** To indicate to R that my factor is a factor rather than a continuous variable

**nrow(**mydata**)** To get the number of row in a data.frame

**unique(**mydata**)** To get the same data.frame without any duplicate

**emmeans(**mod**, ~ 1)** To get the confidence intervalle on the mean of a linear mixed model

**show\_test()** A function which operates on anova tables and LS-means tables makes it possible to see exactly which functions of the coefficients are being tested. This is helpful when differences between type I, II and III anova tables are being considered and discussed

**lmerTest::contest()** Tests of vector or matrix contrasts for lmer model fits

**pairs(emmeans(**mod**, ~**myfactor**))** Paiwise t-test between all the level of a specific factor

**qt(.**x**, df =** df**)** Find the t-score of the xth quantile (ex: .99 for the 99th quantile) of the Student t distribution

**powerSim(**mod**, nsim =)** To calculate the power of our model based on a number of simulations (nsim).

**expand.grid()** To create a dataframe from all combinations of the supplied vectors or factors

mod**@beta[1]** To get the intercept of a mixed model

**VarCorr(**mod**)** To get the standard deviation of random effects of a mixed model

**makeLmer()** To create a fitted lmer model from Std. deviation of random effects, S², fixed effect estimates and plan of a previous mixed model. Usefull to use powerSim.

**powerCurve ()** This function runs powerSim over a range of sample sizes

**corDiGGer()** To create a design

**interplot()** To produce a plot of the coefficient estimates of one variable in a two-way interaction conditional on the values of the other variable in the interaction term

**contour()** Create a contour plot, or add contour lines to an existing plot (for surface optimization)

**rsm()** Fit a linear model with a response-surface component, and produce appropriate analyses and summaries