

R companion for the lmerTest tutorial

Per Bruun Brockhoff

Tuesday, June 23, 2015

This is an R Markdown version of the introduction to mixed models in R.

```
# Set working Directory: (to whatever matches your local setting)
# setwd("C:/perbb/RioCourseJune15/Lavras2dayCourse/Day1/lmerTest")

# Import data:
sensintro <- read.table("introexample.csv",header=TRUE,sep="," ,dec=".")

# Make relevant variables into factors:
sensintro$assessor=factor(sensintro$assessor)
```

```
# Check your data:
head(sensintro)
summary(sensintro)
```

```
# Check your data:
head(sensintro)
```

```
##  assessor product sweet
## 1         1      A  10.4
## 2         2      A  10.6
## 3         3      A  10.2
## 4         4      A  10.1
## 5         5      A  10.3
## 6         6      A  10.7
```

```
summary(sensintro)
```

```
##  assessor product      sweet
## 1      : 2  A:20  Min.   : 9.60
## 2      : 2  B:20  1st Qu.:10.03
## 3      : 2      Median :10.25
## 4      : 2      Mean   :10.28
## 5      : 2  3rd Qu.:10.50
## 6      : 2  Max.   :11.00
## (Other):28      NA's   :10
```

```
# Make a data file with only the first 10 assessors
sensintro10=subset(sensintro,as.integer(assessor)<11)
```

```
# Fixed effects analysis:
anova(lm(sweet~assessor+product,data=sensintro))
```

```
## Analysis of Variance Table
```

```
##
## Response: sweet
##           Df Sum Sq Mean Sq F value Pr(>F)
## assessor  19 3.7230 0.195947  4.4931 0.01288 *
## product    1 0.0125 0.012500  0.2866 0.60537
## Residuals   9 0.3925 0.043611
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Same as paired t-test on the 10 assessors with no missings:
with(sensintro10, t.test(sweet~product,paired=TRUE))
```

```
##
## Paired t-test
##
## data:  sweet by product
## t = -0.53537, df = 9, p-value = 0.6054
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
## -0.2612693  0.1612693
## sample estimates:
## mean of the differences
## -0.05
```

```
# The mixed model version of it:
# We first just load the lme4 package
# If first time: first install the package
library(lme4)
```

```
## Loading required package: Matrix
## Loading required package: Rcpp
```

```
# Analysing the 10 first assessors:
lmer0 <- lmer(sweet~product+(1|assessor), data=sensintro10)
summary(lmer0)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: sweet ~ product + (1 | assessor)
## Data: sensintro10
##
## REML criterion at convergence: 14
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.28851 -0.50128 -0.03985  0.52348  1.82403
##
## Random effects:
## Groups Name Variance Std.Dev.
## assessor (Intercept) 0.08933  0.2989
## Residual            0.04361  0.2088
## Number of obs: 20, groups: assessor, 10
##
```

```
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept) 10.34000   0.11530   89.68
## productB    0.05000   0.09339    0.54
##
## Correlation of Fixed Effects:
##           (Intr)
## productB -0.405
```

```
anova(lmer0)
```

```
## Analysis of Variance Table
##           Df Sum Sq Mean Sq F value
## product   1 0.0125  0.0125  0.2866
```

```
# Analysing all 20 assessors:
```

```
lmer1 <- lmer(sweet~product+(1|assessor), data=sensintro)
summary(lmer1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: sweet ~ product + (1 | assessor)
## Data: sensintro
##
## REML criterion at convergence: 24.7
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.16659 -0.43803  0.00315  0.43635  1.84478
##
## Random effects:
## Groups Name Variance Std.Dev.
## assessor (Intercept) 0.10192  0.3192
## Residual              0.04347  0.2085
## Number of obs: 30, groups: assessor, 20
##
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept) 10.21175   0.09259  110.29
## productB    0.07211   0.08697   0.83
##
## Correlation of Fixed Effects:
##           (Intr)
## productB -0.470
```

```
anova(lmer1)
```

```
## Analysis of Variance Table
##           Df Sum Sq Mean Sq F value
## product   1 0.029887 0.029887  0.6875
```

```

# We next load the lmerTest package,
# which actually also loads the lme4-package
# So in the future you don't need the step above
# If first time: first install the package
library(lmerTest)

```

```

##
## Attaching package: 'lmerTest'
##
## The following object is masked from 'package:lme4':
##
##   lmer
##
## The following object is masked from 'package:stats':
##
##   step

```

```

# Now check the anova-output:
lmer1 <- lmer(sweet-product+(1|assessor), data=sensintro)
anova(lmer1)

```

```

## Analysis of Variance Table of type III with Satterthwaite
## approximation for degrees of freedom
##           Sum Sq Mean Sq NumDF DenDF F.value Pr(>F)
## product 0.029887 0.029887     1 11.696 0.68746 0.4236

```

```

# Or to get (almost) everything -
# use the step-function of lmerTest:
# We use it now WITHOUT the automated model reduction feature
s <- step(lmer1,reduce.fixed = FALSE, reduce.random = FALSE)
s

```

```

##
## Random effects:
##           Chi.sq Chi.DF p.value
## assessor   5.95     1 0.0148
##
## Fixed effects:
##           Sum Sq Mean Sq NumDF DenDF F.value Pr(>F)
## product 0.0299 0.0299     1 11.7 0.6875 0.4236
##
## Least squares means:
##           product Estimate Standard Error  DF t-value Lower CI Upper CI
## product A      1.0 10.2117      0.0926 25.8 110.2900    10.0    10.4
## product B      2.0 10.2839      0.0926 25.8 111.0700    10.1    10.5
##           p-value
## product A <2e-16 ***
## product B <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Differences of LSMEANS:

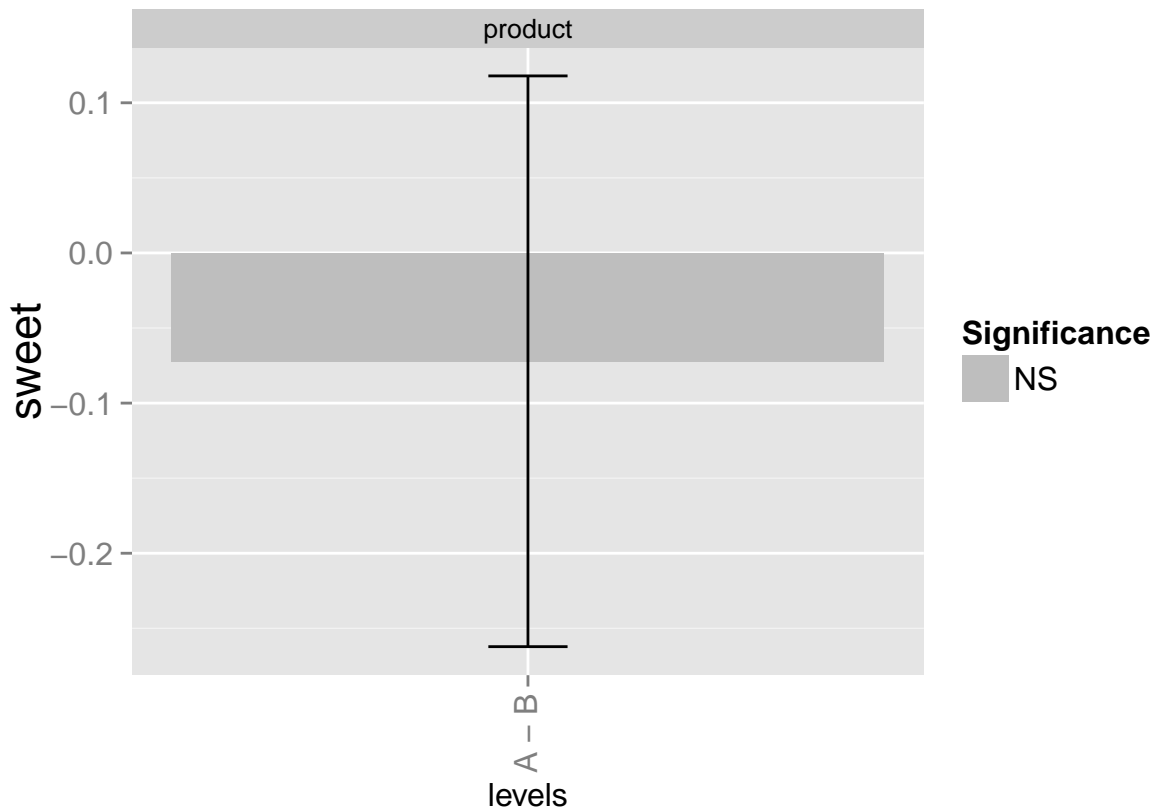
```

```
##           Estimate Standard Error   DF t-value Lower CI Upper CI
## product A - B      -0.1         0.0870 11.7   -0.83   -0.262   0.118
##           p-value
## product A - B       0.4
##
## Final model:
## lme4::lmer(formula = sweet ~ product + (1 | assessor), data = sensintro,
##           REML = reml.lmerTest.private, contrasts = l.lmerTest.private.contrast,
##           devFunOnly = devFunOnly.lmerTest.private)
```

```
# To check what the R-names of the various output is:
# str(s)
s$diffs.lsmmeans.table
```

```
##           Estimate Standard Error   DF t-value Lower CI Upper CI
## product A - B  -0.0721         0.087 11.7   -0.83  -0.2622   0.1179
##           p-value
## product A - B   0.4236
```

```
plot(s)
```



```
# Now the ham data:
data(ham)
modelHam<-lmer(Informed.liking ~ Product*Information*Gender*Age+
(1|Consumer) +(1|Product:Consumer) + (1|Information:Consumer), data=ham)
```

```
stepHam<-step(modelHam)
```

```
# Final model
```

```
stepHam$model
```

```
## Linear mixed model fit by REML ['merModLmerTest']
## Formula: Informed.liking ~ Product + Information + (1 | Consumer) + (1 |
##   Product:Consumer)
##   Data: ham
## REML criterion at convergence: 2709.074
## Random effects:
##   Groups          Name          Std.Dev.
##   Product:Consumer (Intercept) 1.7759
##   Consumer         (Intercept) 0.6112
##   Residual                1.2977
## Number of obs: 648, groups: Product:Consumer, 324; Consumer, 81
## Fixed Effects:
##   (Intercept)      Product2      Product3      Product4      Information2
##           5.7083      -0.7037      0.2840      0.1173      0.2006
```

```
# Overview of random effects
```

```
stepHam$rand.table
```

```
##           Chi.sq Chi.DF elim.num      p.value
## Information:Consumer  1.538043      1      1 2.149094e-01
## Consumer            2.941389      1      kept 8.633646e-02
## Product:Consumer    165.560425      1      kept 6.900611e-38
```

```
# Overview of fixed effects
```

```
stepHam$anova.table
```

```
##           Sum Sq      Mean Sq NumDF      DenDF
## Product:Information:Gender:Age  7.206530890  2.402176963      3 307.99892
## Product:Gender:Age            0.623362524  0.207787508      3 231.00147
## Product:Information:Gender    5.878938234  1.959646078      3 311.00058
## Product:Gender                0.915467117  0.305155706      3 234.00091
## Product:Information:Age       7.214105057  2.404701686      3 314.00069
## Product:Age                   4.033448707  1.344482902      3 236.99994
## Product:Information          10.387345679  3.462448560      3 317.00066
## Information:Gender:Age       5.341456607  5.341456607      1 320.00022
## Information:Age              0.008425032  0.008425032      1 321.00020
## Gender:Age                   1.201674926  1.201674926      1  77.00000
## Age                          0.025389297  0.025389297      1  78.00021
## Information:Gender           1.407453516  1.407453516      1 322.00088
## Gender                       1.480262640  1.480262640      1  78.99996
## Product                     19.346565058  6.448855019      3 240.00064
## Information                   6.520061728  6.520061728      1 323.00067
##           F.value elim.num      Pr(>F)
## Product:Information:Gender:Age  1.460648189      1 0.22530100
## Product:Gender:Age            0.125786650      2 0.94477581
## Product:Information:Gender    1.186295201      3 0.31505508
```

```

## Product:Gender          0.184401451      4 0.90693822
## Product:Information:Age  1.453128611      5 0.22736708
## Product:Age             0.808983592      6 0.48997996
## Product:Information     2.083376538      7 0.10231772
## Information:Gender:Age   3.181672113      8 0.07541605
## Information:Age         0.004984545      9 0.94375903
## Gender:Age              0.713157001     10 0.40101396
## Age                    0.015067763     11 0.90262015
## Information:Gender      0.835280165     12 0.36143397
## Gender                 0.878938345     13 0.35135000
## Product                3.829148969    kept 0.01047589
## Information            3.871429512    kept 0.04996999

```

```
# Differences of LSMEANS for final model
```

```
stepHam$diffs.lsmeans.table
```

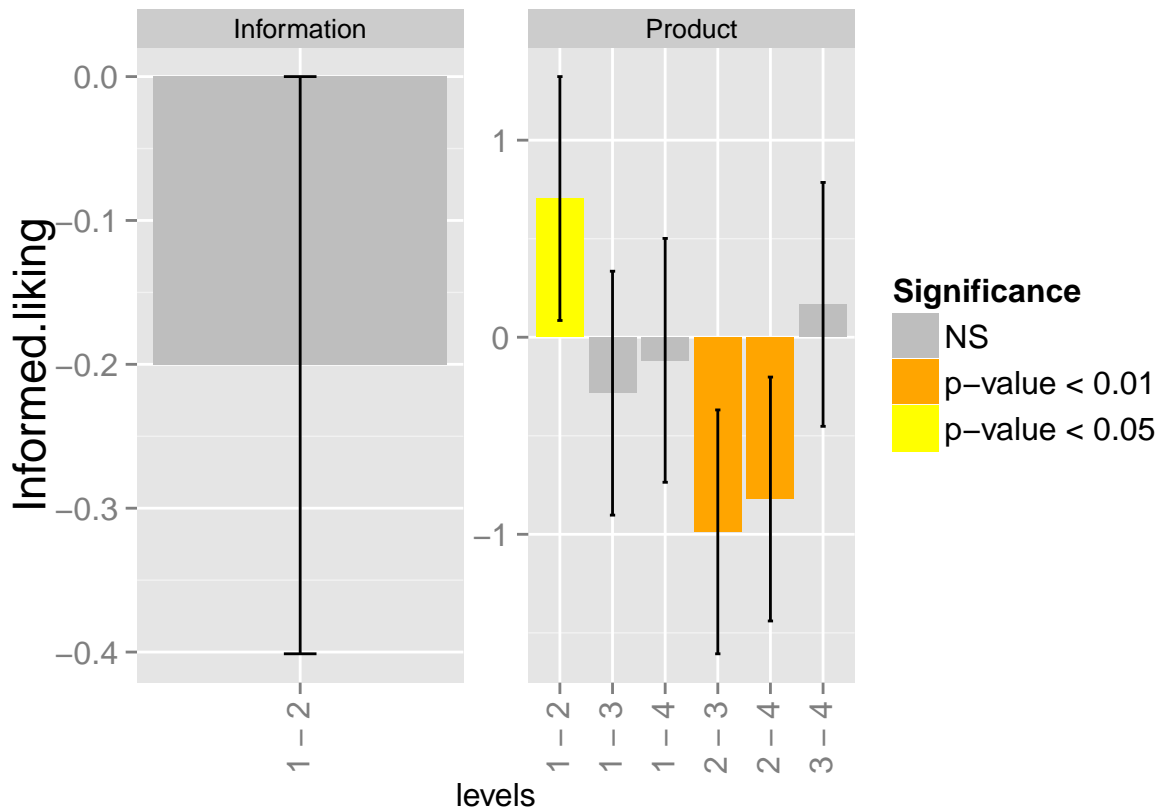
```

##          Estimate Standard Error  DF t-value Lower CI Upper CI
## Product 1 - 2      0.7037      0.3141 240    2.24   0.0849   1.3225
## Product 1 - 3     -0.2840      0.3141 240   -0.90  -0.9027   0.3348
## Product 1 - 4     -0.1173      0.3141 240   -0.37  -0.7361   0.5015
## Product 2 - 3     -0.9877      0.3141 240   -3.14  -1.6064  -0.3689
## Product 2 - 4     -0.8210      0.3141 240   -2.61  -1.4398  -0.2022
## Product 3 - 4      0.1667      0.3141 240    0.53  -0.4521   0.7854
## Information 1 - 2 -0.2006      0.1020 323   -1.97  -0.4012   0.0000
##          p-value
## Product 1 - 2      0.0260
## Product 1 - 3      0.3669
## Product 1 - 4      0.7092
## Product 2 - 3      0.0019
## Product 2 - 4      0.0095
## Product 3 - 4      0.5962
## Information 1 - 2  0.0500

```

```
# Look at the (default) plots
```

```
plot(stepHam)
```



```
# From Kuznetsova et al (2015), page 34.
# Automated analysis of TVbo data set in R
# Running the largest possible model:
modelTVbo <- lmer(Colourbalance ~ TVset*Picture + (1 | Assessor) +
  (1 | Assessor:TVset) + (1 | Assessor:Picture) +
  (1 | Assessor:TVset:Picture) + (1 | Repeat) +
  (1 | Repeat:TVset) + (1 | Repeat:Picture) +
  (1 | Repeat:TVset:Picture), data = TVbo)

# Running the step-function of lmerTest with default options:
stepTVbo <- step(modelTVbo)

# Look at (all) the results
stepTVbo
```

```
##
## Random effects:
##
```

	Chi.sq	Chi.DF	elim.num	p.value
## Repeat:Picture	0.00	1	1	1.0000
## Repeat:TVset:Picture	0.00	1	2	1.0000
## Repeat	0.07	1	3	0.7924
## Assessor:TVset:Picture	0.21	1	4	0.6439
## Assessor	0.28	1	5	0.5990
## Assessor:TVset	66.78	1	kept	<1e-07
## Assessor:Picture	3.92	1	kept	0.0478


```

## Repeat:TVset          5.36      1      kept  0.0206
##
## Fixed effects:
##           Sum Sq Mean Sq NumDF  DenDF F.value elim.num Pr(>F)
## TVset      15.2315  7.6157   2   15.90  3.8988     kept 0.0419
## Picture     8.0123  2.6708   3   20.59  1.3673     kept 0.2806
## TVset:Picture 52.3508  8.7251   6  135.35  4.4668     kept 0.0004
##
## Least squares means:
##           TVset Picture Estimate Standard Error  DF t-value
## TVset TV1          1.0      NA      8.380          0.654 16.6  12.81
## TVset TV2          2.0      NA      6.614          0.654 16.6  10.11
## TVset TV3          3.0      NA      9.098          0.654 16.6  13.91
## Picture 1           NA      1.0      7.865          0.452 29.1  17.38
## Picture 2           NA      2.0      8.300          0.452 29.1  18.34
## Picture 3           NA      3.0      8.298          0.452 29.1  18.34
## Picture 4           NA      4.0      7.660          0.452 29.1  16.93
## TVset:Picture TV1 1  1.0      1.0      9.006          0.738 26.7  12.20
## TVset:Picture TV2 1  2.0      1.0      5.869          0.738 26.7   7.95
## TVset:Picture TV3 1  3.0      1.0      8.719          0.738 26.7  11.81
## TVset:Picture TV1 2  1.0      2.0      7.763          0.738 26.7  10.51
## TVset:Picture TV2 2  2.0      2.0      7.469          0.738 26.7  10.11
## TVset:Picture TV3 2  3.0      2.0      9.669          0.738 26.7  13.09
## TVset:Picture TV1 3  1.0      3.0      8.350          0.738 26.7  11.31
## TVset:Picture TV2 3  2.0      3.0      7.463          0.738 26.7  10.11
## TVset:Picture TV3 3  3.0      3.0      9.081          0.738 26.7  12.30
## TVset:Picture TV1 4  1.0      4.0      8.400          0.738 26.7  11.38
## TVset:Picture TV2 4  2.0      4.0      5.656          0.738 26.7   7.66
## TVset:Picture TV3 4  3.0      4.0      8.925          0.738 26.7  12.09
##
##           Lower CI Upper CI p-value
## TVset TV1          7.00   9.76 <2e-16 ***
## TVset TV2          5.23   8.00 <2e-16 ***
## TVset TV3          7.72  10.48 <2e-16 ***
## Picture 1          6.94   8.79 <2e-16 ***
## Picture 2          7.37   9.23 <2e-16 ***
## Picture 3          7.37   9.22 <2e-16 ***
## Picture 4          6.74   8.59 <2e-16 ***
## TVset:Picture TV1 1  7.49  10.52 <2e-16 ***
## TVset:Picture TV2 1  4.35   7.38 <2e-16 ***
## TVset:Picture TV3 1  7.20  10.23 <2e-16 ***
## TVset:Picture TV1 2  6.25   9.28 <2e-16 ***
## TVset:Picture TV2 2  5.95   8.98 <2e-16 ***
## TVset:Picture TV3 2  8.15  11.18 <2e-16 ***
## TVset:Picture TV1 3  6.83   9.87 <2e-16 ***
## TVset:Picture TV2 3  5.95   8.98 <2e-16 ***
## TVset:Picture TV3 3  7.57  10.60 <2e-16 ***
## TVset:Picture TV1 4  6.88   9.92 <2e-16 ***
## TVset:Picture TV2 4  4.14   7.17 <2e-16 ***
## TVset:Picture TV3 4  7.41  10.44 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Differences of LSMEANS:
##           Estimate Standard Error  DF t-value

```

## TVset TV1 - TV2	1.8	0.9156	15.9	1.93
## TVset TV1 - TV3	-0.7	0.9156	15.9	-0.78
## TVset TV2 - TV3	-2.5	0.9156	15.9	-2.71
## Picture 1 - 2	-0.4	0.3879	20.6	-1.12
## Picture 1 - 3	-0.4	0.3879	20.6	-1.12
## Picture 1 - 4	0.2	0.3879	20.6	0.53
## Picture 2 - 3	0.0	0.3879	20.6	0.01
## Picture 2 - 4	0.6	0.3879	20.6	1.65
## Picture 3 - 4	0.6	0.3879	20.6	1.64
## TVset:Picture TV1 1 - TV2 1	3.1	1.0107	23.5	3.10
## TVset:Picture TV1 1 - TV3 1	0.3	1.0107	23.5	0.28
## TVset:Picture TV1 1 - TV1 2	1.2	0.5597	75.7	2.22
## TVset:Picture TV1 1 - TV2 2	1.5	1.0443	26.7	1.47
## TVset:Picture TV1 1 - TV3 2	-0.7	1.0443	26.7	-0.63
## TVset:Picture TV1 1 - TV1 3	0.7	0.5597	75.7	1.17
## TVset:Picture TV1 1 - TV2 3	1.5	1.0443	26.7	1.48
## TVset:Picture TV1 1 - TV3 3	-0.1	1.0443	26.7	-0.07
## TVset:Picture TV1 1 - TV1 4	0.6	0.5597	75.7	1.08
## TVset:Picture TV1 1 - TV2 4	3.4	1.0443	26.7	3.21
## TVset:Picture TV1 1 - TV3 4	0.1	1.0443	26.7	0.08
## TVset:Picture TV2 1 - TV3 1	-2.8	1.0107	23.5	-2.82
## TVset:Picture TV2 1 - TV1 2	-1.9	1.0443	26.7	-1.81
## TVset:Picture TV2 1 - TV2 2	-1.6	0.5597	75.7	-2.86
## TVset:Picture TV2 1 - TV3 2	-3.8	1.0443	26.7	-3.64
## TVset:Picture TV2 1 - TV1 3	-2.5	1.0443	26.7	-2.38
## TVset:Picture TV2 1 - TV2 3	-1.6	0.5597	75.7	-2.85
## TVset:Picture TV2 1 - TV3 3	-3.2	1.0443	26.7	-3.08
## TVset:Picture TV2 1 - TV1 4	-2.5	1.0443	26.7	-2.42
## TVset:Picture TV2 1 - TV2 4	0.2	0.5597	75.7	0.38
## TVset:Picture TV2 1 - TV3 4	-3.1	1.0443	26.7	-2.93
## TVset:Picture TV3 1 - TV1 2	1.0	1.0443	26.7	0.92
## TVset:Picture TV3 1 - TV2 2	1.2	1.0443	26.7	1.20
## TVset:Picture TV3 1 - TV3 2	-1.0	0.5597	75.7	-1.70
## TVset:Picture TV3 1 - TV1 3	0.4	1.0443	26.7	0.35
## TVset:Picture TV3 1 - TV2 3	1.3	1.0443	26.7	1.20
## TVset:Picture TV3 1 - TV3 3	-0.4	0.5597	75.7	-0.65
## TVset:Picture TV3 1 - TV1 4	0.3	1.0443	26.7	0.31
## TVset:Picture TV3 1 - TV2 4	3.1	1.0443	26.7	2.93
## TVset:Picture TV3 1 - TV3 4	-0.2	0.5597	75.7	-0.37
## TVset:Picture TV1 2 - TV2 2	0.3	1.0107	23.5	0.29
## TVset:Picture TV1 2 - TV3 2	-1.9	1.0107	23.5	-1.89
## TVset:Picture TV1 2 - TV1 3	-0.6	0.5597	75.7	-1.05
## TVset:Picture TV1 2 - TV2 3	0.3	1.0443	26.7	0.29
## TVset:Picture TV1 2 - TV3 3	-1.3	1.0443	26.7	-1.26
## TVset:Picture TV1 2 - TV1 4	-0.6	0.5597	75.7	-1.14
## TVset:Picture TV1 2 - TV2 4	2.1	1.0443	26.7	2.02
## TVset:Picture TV1 2 - TV3 4	-1.2	1.0443	26.7	-1.11
## TVset:Picture TV2 2 - TV3 2	-2.2	1.0107	23.5	-2.18
## TVset:Picture TV2 2 - TV1 3	-0.9	1.0443	26.7	-0.84
## TVset:Picture TV2 2 - TV2 3	0.0	0.5597	75.7	0.01
## TVset:Picture TV2 2 - TV3 3	-1.6	1.0443	26.7	-1.54
## TVset:Picture TV2 2 - TV1 4	-0.9	1.0443	26.7	-0.89
## TVset:Picture TV2 2 - TV2 4	1.8	0.5597	75.7	3.24
## TVset:Picture TV2 2 - TV3 4	-1.5	1.0443	26.7	-1.39

## TVset:Picture	TV3 2 - TV1 3	1.3	1.0443	26.7	1.26
## TVset:Picture	TV3 2 - TV2 3	2.2	1.0443	26.7	2.11
## TVset:Picture	TV3 2 - TV3 3	0.6	0.5597	75.7	1.05
## TVset:Picture	TV3 2 - TV1 4	1.3	1.0443	26.7	1.21
## TVset:Picture	TV3 2 - TV2 4	4.0	1.0443	26.7	3.84
## TVset:Picture	TV3 2 - TV3 4	0.7	0.5597	75.7	1.33
## TVset:Picture	TV1 3 - TV2 3	0.9	1.0107	23.5	0.88
## TVset:Picture	TV1 3 - TV3 3	-0.7	1.0107	23.5	-0.72
## TVset:Picture	TV1 3 - TV1 4	0.0	0.5597	75.7	-0.09
## TVset:Picture	TV1 3 - TV2 4	2.7	1.0443	26.7	2.58
## TVset:Picture	TV1 3 - TV3 4	-0.6	1.0443	26.7	-0.55
## TVset:Picture	TV2 3 - TV3 3	-1.6	1.0107	23.5	-1.60
## TVset:Picture	TV2 3 - TV1 4	-0.9	1.0443	26.7	-0.90
## TVset:Picture	TV2 3 - TV2 4	1.8	0.5597	75.7	3.23
## TVset:Picture	TV2 3 - TV3 4	-1.5	1.0443	26.7	-1.40
## TVset:Picture	TV3 3 - TV1 4	0.7	1.0443	26.7	0.65
## TVset:Picture	TV3 3 - TV2 4	3.4	1.0443	26.7	3.28
## TVset:Picture	TV3 3 - TV3 4	0.2	0.5597	75.7	0.28
## TVset:Picture	TV1 4 - TV2 4	2.7	1.0107	23.5	2.71
## TVset:Picture	TV1 4 - TV3 4	-0.5	1.0107	23.5	-0.52
## TVset:Picture	TV2 4 - TV3 4	-3.3	1.0107	23.5	-3.23
##					
		Lower CI	Upper CI	p-value	
## TVset	TV1 - TV2	-0.1764	3.708	0.072	.
## TVset	TV1 - TV3	-2.6608	1.223	0.444	
## TVset	TV2 - TV3	-4.4264	-0.542	0.015	*
## Picture	1 - 2	-1.2431	0.372	0.275	
## Picture	1 - 3	-1.2410	0.374	0.277	
## Picture	1 - 4	-0.6035	1.012	0.604	
## Picture	2 - 3	-0.8056	0.810	0.996	
## Picture	2 - 4	-0.1681	1.447	0.114	
## Picture	3 - 4	-0.1702	1.445	0.116	
## TVset:Picture	TV1 1 - TV2 1	1.0493	5.226	0.005	**
## TVset:Picture	TV1 1 - TV3 1	-1.8007	2.376	0.778	
## TVset:Picture	TV1 1 - TV1 2	0.1289	2.359	0.029	*
## TVset:Picture	TV1 1 - TV2 2	-0.6065	3.682	0.153	
## TVset:Picture	TV1 1 - TV3 2	-2.8065	1.482	0.531	
## TVset:Picture	TV1 1 - TV1 3	-0.4586	1.771	0.245	
## TVset:Picture	TV1 1 - TV2 3	-0.6003	3.688	0.151	
## TVset:Picture	TV1 1 - TV3 3	-2.2190	2.069	0.943	
## TVset:Picture	TV1 1 - TV1 4	-0.5086	1.721	0.282	
## TVset:Picture	TV1 1 - TV2 4	1.2060	5.494	0.004	**
## TVset:Picture	TV1 1 - TV3 4	-2.0628	2.225	0.939	
## TVset:Picture	TV2 1 - TV3 1	-4.9382	-0.762	0.010	**
## TVset:Picture	TV2 1 - TV1 2	-4.0378	0.250	0.081	.
## TVset:Picture	TV2 1 - TV2 2	-2.7148	-0.485	0.005	**
## TVset:Picture	TV2 1 - TV3 2	-5.9440	-1.656	0.001	**
## TVset:Picture	TV2 1 - TV1 3	-4.6253	-0.337	0.025	*
## TVset:Picture	TV2 1 - TV2 3	-2.7086	-0.479	0.006	**
## TVset:Picture	TV2 1 - TV3 3	-5.3565	-1.068	0.005	**
## TVset:Picture	TV2 1 - TV1 4	-4.6753	-0.387	0.022	*
## TVset:Picture	TV2 1 - TV2 4	-0.9023	1.327	0.705	
## TVset:Picture	TV2 1 - TV3 4	-5.2003	-0.912	0.007	**
## TVset:Picture	TV3 1 - TV1 2	-1.1878	3.100	0.368	
## TVset:Picture	TV3 1 - TV2 2	-0.8940	3.394	0.242	

```

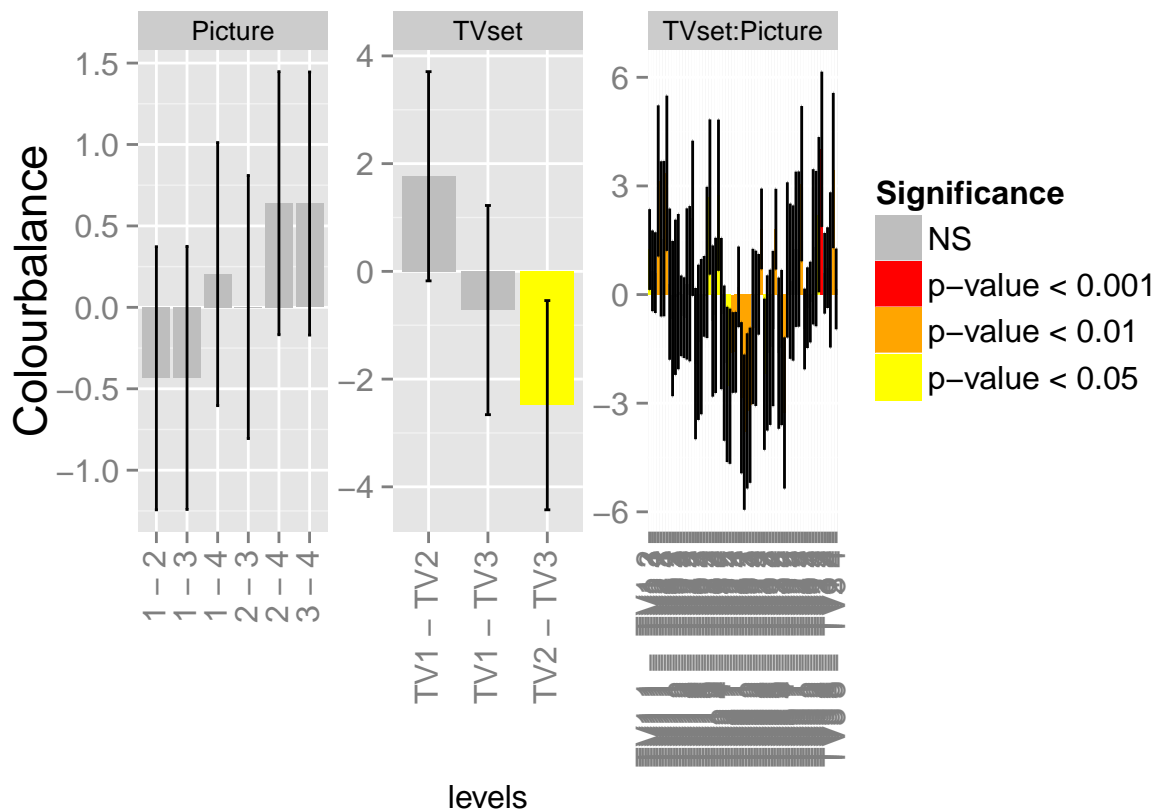
## TVset:Picture TV3 1 - TV3 2 -2.0648 0.165 0.094 .
## TVset:Picture TV3 1 - TV1 3 -1.7753 2.513 0.727
## TVset:Picture TV3 1 - TV2 3 -0.8878 3.400 0.240
## TVset:Picture TV3 1 - TV3 3 -1.4773 0.752 0.519
## TVset:Picture TV3 1 - TV1 4 -1.8253 2.463 0.763
## TVset:Picture TV3 1 - TV2 4 0.9185 5.207 0.007 **
## TVset:Picture TV3 1 - TV3 4 -1.3211 0.909 0.714
## TVset:Picture TV1 2 - TV2 2 -1.7945 2.382 0.774
## TVset:Picture TV1 2 - TV3 2 -3.9945 0.182 0.072 .
## TVset:Picture TV1 2 - TV1 3 -1.7023 0.527 0.297
## TVset:Picture TV1 2 - TV2 3 -1.8440 2.444 0.776
## TVset:Picture TV1 2 - TV3 3 -3.4628 0.825 0.218
## TVset:Picture TV1 2 - TV1 4 -1.7523 0.477 0.258
## TVset:Picture TV1 2 - TV2 4 -0.0378 4.250 0.054 .
## TVset:Picture TV1 2 - TV3 4 -3.3065 0.982 0.276
## TVset:Picture TV2 2 - TV3 2 -4.2882 -0.112 0.040 *
## TVset:Picture TV2 2 - TV1 3 -3.0253 1.263 0.406
## TVset:Picture TV2 2 - TV2 3 -1.1086 1.121 0.991
## TVset:Picture TV2 2 - TV3 3 -3.7565 0.531 0.134
## TVset:Picture TV2 2 - TV1 4 -3.0753 1.213 0.380
## TVset:Picture TV2 2 - TV2 4 0.6977 2.927 0.002 **
## TVset:Picture TV2 2 - TV3 4 -3.6003 0.688 0.175
## TVset:Picture TV3 2 - TV1 3 -0.8253 3.463 0.218
## TVset:Picture TV3 2 - TV2 3 0.0622 4.350 0.044 *
## TVset:Picture TV3 2 - TV3 3 -0.5273 1.702 0.297
## TVset:Picture TV3 2 - TV1 4 -0.8753 3.413 0.235
## TVset:Picture TV3 2 - TV2 4 1.8685 6.157 7e-04 ***
## TVset:Picture TV3 2 - TV3 4 -0.3711 1.859 0.188
## TVset:Picture TV1 3 - TV2 3 -1.2007 2.976 0.389
## TVset:Picture TV1 3 - TV3 3 -2.8195 1.357 0.476
## TVset:Picture TV1 3 - TV1 4 -1.1648 1.065 0.929
## TVset:Picture TV1 3 - TV2 4 0.5497 4.838 0.016 *
## TVset:Picture TV1 3 - TV3 4 -2.7190 1.569 0.587
## TVset:Picture TV2 3 - TV3 3 -3.7070 0.469 0.123
## TVset:Picture TV2 3 - TV1 4 -3.0815 1.206 0.377
## TVset:Picture TV2 3 - TV2 4 0.6914 2.921 0.002 **
## TVset:Picture TV2 3 - TV3 4 -3.6065 0.681 0.173
## TVset:Picture TV3 3 - TV1 4 -1.4628 2.825 0.520
## TVset:Picture TV3 3 - TV2 4 1.2810 5.569 0.003 **
## TVset:Picture TV3 3 - TV3 4 -0.9586 1.271 0.781
## TVset:Picture TV1 4 - TV2 4 0.6555 4.832 0.012 *
## TVset:Picture TV1 4 - TV3 4 -2.6132 1.563 0.608
## TVset:Picture TV2 4 - TV3 4 -5.3570 -1.181 0.004 **
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Final model:
## lme4::lmer(formula = Colourbalance ~ TVset + Picture + (1 | Assessor:TVset) +
## (1 | Assessor:Picture) + (1 | Repeat:TVset) + TVset:Picture,
## data = TVbo, REML = reml.lmerTest.private, contrasts = l.lmerTest.private.contrast,
## devFunOnly = devFunOnly.lmerTest.private)

```

```

# Look at the (default) plots
plot(stepTVbo)

```



```
# Showing ALL the (default)options for the step function:
stepTVbo=step(modelTVbo, ddf="Satterthwaite", type=3, alpha.random = 0.1,
  alpha.fixed = 0.05, reduce.fixed = TRUE, reduce.random = TRUE,
  lsmeans.calc=TRUE, difflsmeans.calc=TRUE, test.effs=NULL,
  keep.effs = NULL, method.grad="simple")
# Check the details:
# (choose the help info regarding the lmerTest version of step )
?step
```

```
# And finally the carrots data:
modelCarrots <- lmer(Preference~sens2*sens1*Homesize*Age+
  (1 | product) + (1 + sens1 + sens2 | Consumer),
  data=carrots)
stepCarrots=step(modelCarrots)
stepCarrots
```

```
##
## Random effects:
##           Chi.sq Chi.DF elim.num p.value
## sens1:Consumer    2.45     3         1 0.4853
## product           17.34     1       kept    0
## sens2:Consumer    8.23     2       kept 0.0163
##
```

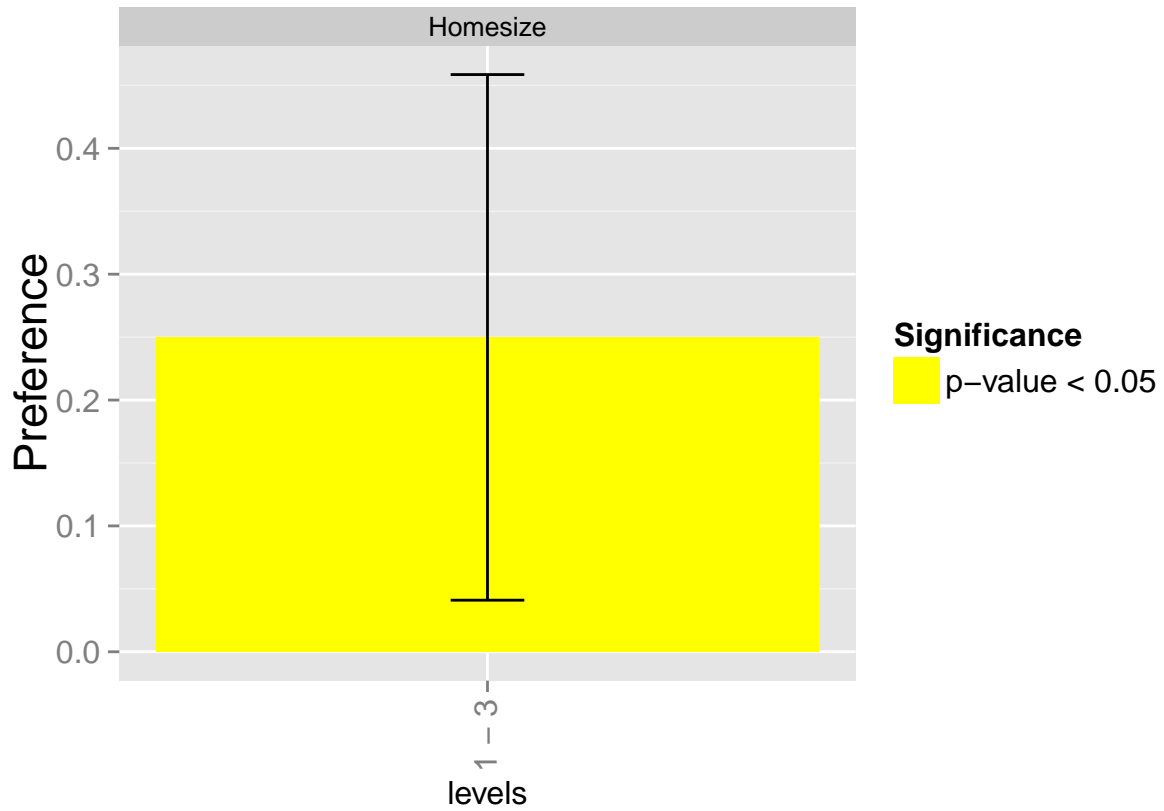
```

## Fixed effects:
##
##          Sum Sq Mean Sq NumDF   DenDF F.value elim.num
## sens2:sens1:Homesize:Age 3.5460  1.1820    3 1003.07  1.1419     1
## sens2:Homesize:Age      1.2121  0.4040    3   94.98  0.3901     2
## sens2:sens1:Homesize    0.5873  0.5873    1 1006.10  0.5671     3
## sens2:Homesize         2.0757  2.0757    1   97.99  2.0051     4
## sens1:Homesize:Age     5.6623  1.8874    3 1007.10  1.8233     5
## sens1:Homesize         0.1513  0.1513    1 1010.17  0.1458     6
## Homesize:Age          5.9012  1.9671    3   94.85  1.8972     7
## sens2:sens1:Age       7.2292  2.4097    3 1011.14  2.3242     8
## sens2:sens1           0.1181  0.1181    1    7.98  0.1134     9
## sens1:Age            1.4003  0.4668    3 1014.18  0.4484    10
## sens2:Age            2.3193  0.7731    3   99.01  0.7440    11
## Age                  0.6683  0.2228    3   97.99  0.2144    12
## sens1                 0.5423  0.5423    1    8.98  0.5219    13
## sens2                18.1647 18.1647    1   12.19 17.4799   kept
## Homesize              5.8511  5.8511    1  100.97  5.6305   kept
##
##          Pr(>F)
## sens2:sens1:Homesize:Age 0.3311
## sens2:Homesize:Age      0.7604
## sens2:sens1:Homesize    0.4516
## sens2:Homesize         0.1599
## sens1:Homesize:Age     0.1412
## sens1:Homesize         0.7027
## Homesize:Age          0.1353
## sens2:sens1:Age       0.0734
## sens2:sens1           0.7450
## sens1:Age            0.7185
## sens2:Age            0.5284
## Age                  0.8862
## sens1                 0.4884
## sens2                 0.0012
## Homesize              0.0195
##
## Least squares means:
##
##          Homesize Estimate Standard Error   DF t-value Lower CI
## Homesize 1      1.0   4.9110      0.0871 39.9 56.3500    4.73
## Homesize 3      2.0   4.6612      0.0937 48.9 49.7600    4.47
##
##          Upper CI p-value
## Homesize 1      5.09 <2e-16 ***
## Homesize 3      4.85 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Differences of LSMEANS:
##
##          Estimate Standard Error   DF t-value Lower CI Upper CI
## Homesize 1 - 3      0.2      0.105 101.0    2.37    0.041    0.459
##
##          p-value
## Homesize 1 - 3      0.02 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Final model:
## lme4::lmer(formula = Preference ~ sens2 + Homesize + (1 | product) +

```

```
## (sens2 | Consumer), data = carrots, REML = reml.lmerTest.private,  
## contrasts = 1.lmerTest.private.contrast, devFunOnly = devFunOnly.lmerTest.private)
```

```
# Look at the (default) plots  
plot(stepCarrots)
```



```
library(knitr)
```

```
purl("lmerTestTutorial.Rmd", documentation = 0)
```