

The Ladybird example from Welham et al. (2014)

Chris Brien

24 February, 2019

Introduction

Welham et al. (2014, Example 8.2) is an experiment to answer the question “Will ladybirds transfer fungus to aphids on plants?” The experiment consists of 2 runs of 36 containers, each with a plant and aphids. There are three factors that results in 12 treatments: Host plant (beans, trefoil), infected Cadavers (5, 10, 20), Ladybird (-, +). They are randomized to the containers within a run so that each is replicated 3 times within a run. The response to be analysed is the logit of the proportion of live aphids that were infected.

Initialize

```
library(asreml)

## Loading required package: Matrix
library(asremlPlus)
library(dae)

## Loading required package: ggplot2
options(width = 105, show.signif.stars = FALSE)
```

Get data available in asremlPlus

```
data("Ladybird.dat")
```

Do an ANOVA of logits

```
Ladybird.aov <- aov(logitP ~ Host*Cadavers*Ladybird + Error(Run/Plant),
                    data=Ladybird.dat)
summary(Ladybird.aov)
```

```
##
## Error: Run
##           Df  Sum Sq Mean Sq F value Pr(>F)
## Residuals  1 0.06766  0.06766
##
## Error: Run:Plant
##           Df  Sum Sq Mean Sq F value  Pr(>F)
## Host       1 13.599   13.599   59.172 1.82e-10
## Cadavers   2 17.027    8.514   37.044 3.78e-11
## Ladybird   1 11.091   11.091   48.257 3.33e-09
## Host:Cadavers 2  0.308    0.154    0.670  0.5158
## Host:Ladybird 1  0.228    0.228    0.992  0.3234
```

```
## Cadavers:Ladybird      2  1.735   0.867   3.774   0.0287
## Host:Cadavers:Ladybird 2  0.200   0.100   0.435   0.6493
## Residuals              59 13.560   0.230
```

The anova table gives the F-tests for the three-factor effects and interactions. Note the **Residuals Mean Sq** value for **Run:Plant** of 0.230. Also, it is clear that the Run component is negative, given that the **Residuals Mean Sq** value for **Run** is less than that for **Run:Plant**; it is $(0.06766 - 0.230) / 36$. From the table it is seen that the only significant interaction is **Cadavers:Ladybird** and that the **Host** main effect is significant.

Use asreml to analyse the logits

Mixed model analysis of logits

```
m <- asreml(logitP ~ Host*Cadavers*Ladybird,
  random = ~ Run,
  residual = ~ Run:Plant,
  data = Ladybird.dat)
```

```
## Model fitted using the gamma parameterization.
```

```
## ASReml 4.1.0 Sun Feb 24 21:33:29 2019
```

```
##          LogLik      Sigma2      DF      wall      cpu
## 1          3.07130      0.226239      60 21:33:29      0.0 (1 restrained)
## 2          3.64260      0.226912      60 21:33:29      0.0 (1 restrained)
## 3          3.71250      0.227106      60 21:33:29      0.0 (1 restrained)
## 4          3.71721      0.227121      60 21:33:29      0.0 (1 restrained)
## 5          3.71751      0.227122      60 21:33:29      0.0 (1 restrained)
```

```
## Warning in asreml(logitP ~ Host * Cadavers * Ladybird, random = ~Run, residual = ~Run:Plant, : Some
## components changed by more than 1% on the last iteration.
```

```
summary(m)$varcomp
```

```
##          component  std.error      z.ratio bound %ch
## Run          2.298309e-08 0.01638903 1.402346e-06    B 93.7
## Run:Plant!R 2.271216e-01 0.04156985 5.463612e+00    P  0.0
```

As expected the Run component is bound (B) at approximately zero. This results in a change in the estimate of the residual variance to 0.227. To allow for a negative estimate we will unconstrain the Run component. As Littell et al. (2006, p.150) say

if you do not set the negative variance component estimate to zero, but allow it to remain negative, you get better control over Type I error and, for cases of negative wholeplot error variance estimates, greater power. Therefore, this is the recommended procedure.

Unconstrain Reps to make the analysis equivalent to ANOVA

```
m <- setvarianceterms(m$call, terms = "Run", bounds = "U")
```

```
## Model fitted using the gamma parameterization.
```

```
## ASReml 4.1.0 Sun Feb 24 21:33:30 2019
```

```
##          LogLik      Sigma2      DF      wall      cpu
## 1          3.07130      0.226239      60 21:33:30      0.0 (1 restrained)
## 2          3.64260      0.226912      60 21:33:30      0.0 (1 restrained)
## 3          3.80283      0.227454      60 21:33:30      0.0 (1 restrained)
```

```
## 4      3.83949      0.233418      60 21:33:30      0.0
## 5      3.95568      0.230971      60 21:33:30      0.0
## 6      3.97333      0.230035      60 21:33:30      0.0
## 7      3.97405      0.229835      60 21:33:30      0.0
```

```
summary(m)$varcomp
```

```
##              component  std.error  z.ratio bound %ch
## Run          -0.004504789 0.002896281 -1.555370    U 0.1
## Run:Plant!R  0.229834648 0.042316936  5.431269    P 0.0
```

Now the Run component is negative and the Run:Plant variance estimate is now equal to that for the Residuals Mean Sq for Run:Plant from the anova table.

Set up an asrtests object

```
current.asrt <- as.asrtests(m)
```

```
## Calculating denominator DF
```

```
print(current.asrt, which = "pseudoanova")
```

```
##
##
## ##### Pseudo-anova table for fixed terms
##
##
## Wald tests for fixed effects.
## Response: logitP
##
##              Df denDF    F.inc    Pr
## (Intercept)    1     1 1550.00 0.0162
## Host           1    59   59.17 0.0000
## Cadavers       2    59   37.04 0.0000
## Ladybird       1    59   48.26 0.0000
## Host:Cadavers  2    59    0.67 0.5158
## Host:Ladybird  1    59    0.99 0.3234
## Cadavers:Ladybird 2    59    3.77 0.0287
## Host:Cadavers:Ladybird 2    59    0.44 0.6493
```

The `asrtests` object contains a `wald.tab` component which can be printed by specifying that the `pseudoanova` is printed. The F -values for the fixed terms in this table are the same as those in the anova table.

Obtain the marginality matrix for the fixed terms

The `pstructure` function from the `dae` package (Brien, 2018) produce the marginality matrix for a formula as a side effect and we take advantage of that to obtain the matrix required here.

```
Ladybird.pstr <- pstructure(~ Host*Cadavers*Ladybird, data = Ladybird.dat)
HCL.marg <- marginality(Ladybird.pstr)
print(HCL.marg)
```

```
##              Host Cadavers Host:Cadavers Ladybird Host:Ladybird Cadavers:Ladybird
## Host           1         0             1         0             1             0
## Cadavers       0         1             1         0             0             1
## Host:Cadavers  0         0             1         0             0             0
```

## Ladybird	0	0	0	1	1	1
## Host:Ladybird	0	0	0	0	1	0
## Cadavers:Ladybird	0	0	0	0	0	1
## Host:Cadavers:Ladybird	0	0	0	0	0	0
##	Host:Cadavers:Ladybird					
## Host				1		
## Cadavers				1		
## Host:Cadavers				1		
## Ladybird				1		
## Host:Ladybird				1		
## Cadavers:Ladybird				1		
## Host:Cadavers:Ladybird				1		

This marginality matrix is interpreted by taking a row term and noting that it is marginal to any column term with a one in this row.

Choose marginality-compliant model

```
sigmod <- chooseModel(current.asrt, terms.marginality = HCL.marg)

## Calculating denominator DF
## Calculating denominator DF
## Calculating denominator DF
## Calculating denominator DF
## Calculating denominator DF

print(sigmod$sig.terms)

## [[1]]
## [1] "Cadavers:Ladybird"
##
## [[2]]
## [1] "Host"

current.asrt <- sigmod$asrtests.obj
print(current.asrt, which = "test")

##
##
## Sequence of model terms whose status in the model has been investigated
##
##      terms DF denDF      p      action
## 1 Host:Cadavers:Ladybird  2    59 0.6493 Nonsignificant
## 2      Cadavers:Ladybird  2    59 0.0287      Significant
## 3      Host:Ladybird    1    59 0.3234 Nonsignificant
## 4      Host:Cadavers    2    59 0.5158 Nonsignificant
## 5              Host    1    59 0.0000      Significant
```

The `chooseModel` function produces a list with components `sigmod`, a list with the terms in the marginality-compliant model, and `asrtests.obj`, the `asrtests` object resulting from the model selection. In particular, the `asrtests` object contains a `test.summary` that details the tests performed in choosing the model. Note that `chooseModel` does not test the main effects for Cadavers or Ladybird, because these are marginal to the significant two-factor interaction Cadavers:Ladybird.

Form formula for selected model

```
mod <- paste(unlist(sigmod$sig.terms), collapse = " + ")
mod <- as.formula(paste("~", mod))
print(mod)
```

```
## ~Cadavers:Ladybird + Host
```

Obtain predictions under the chosen model and form an alldiffs object

```
diffs <- predictPlus(current.asrt$asreml.obj,
  classify = "Host:Ladybird:Cadavers",
  linear.transformation = ~Cadavers:Ladybird + Host,
  wald.tab = current.asrt$wald.tab,
  error.intervals = "halfLeast",
  meanLSD.type = "factor.combination", LSDby = "Host",
  tables = "predictions")
```

```
##
##
## ##### Predictions for logitP transform(s) from Host:Ladybird:Cadavers
##
##
## Notes:
## The original predictions, obtained as described below, have
## been linearly transformed to form estimated marginal means.
## - The predictions are obtained by averaging across the hypertable
##   calculated from model terms constructed solely from factors in
##   the averaging and classify sets.
## - Use 'average' to move ignored factors into the averaging set.
## - The ignored set: Run
##
##
##      Host Ladybird Cadavers predicted.value standard.error upper.halfLeastSignificant.limit
## 1  bean      -         5    -1.6038338      0.1417454      -1.4080222
## 2  bean      -        10    -1.1454308      0.1417454      -0.9496192
## 3  bean      -        20    -0.7448097      0.1417454      -0.5489981
## 4  bean      +         5    -1.0195475      0.1417454      -0.8237359
## 5  bean      +        10    -0.5983440      0.1417454      -0.4025323
## 6  bean      +        20     0.4786704      0.1417454       0.6744820
## 7 trefoil    -         5    -2.4730339      0.1417454      -2.2772223
## 8 trefoil    -        10    -2.0146309      0.1417454      -1.8188193
## 9 trefoil    -        20    -1.6140098      0.1417454      -1.4181982
##10 trefoil    +         5    -1.8887476      0.1417454      -1.6929360
##11 trefoil    +        10    -1.4675441      0.1417454      -1.2717325
##12 trefoil    +        20    -0.3905297      0.1417454      -0.1947181
##      lower.halfLeastSignificant.limit est.status
## 1                                -1.7996454 Estimable
## 2                                -1.3412425 Estimable
## 3                                -0.9406214 Estimable
## 4                                -1.2153592 Estimable
## 5                                -0.7941556 Estimable
## 6                                 0.2828588 Estimable
```

```
## 7          -2.6688455 Estimable
## 8          -2.2104426 Estimable
## 9          -1.8098215 Estimable
## 10         -2.0845593 Estimable
## 11         -1.6633557 Estimable
## 12         -0.5863414 Estimable
##
##
## LSD values
##
## minimum LSD =  0.3916233 0.3916233
##
## mean LSD =  0.3916233 0.3916233
##
## maximum LSD =  0.3916233 0.3916233
##
## (sed range / mean sed =  7.09e-16 5.67e-16 )
```

Setting the `terms` argument to `Host:Ladybird:Cadavers` requests predictions for all combinations of the three factors and the `linear.transformation` argument is used to obtain estimated marginal means (emm) that conform to the chosen model. The `wald.tab` is supplied so that it can be used to get the degrees of freedom for the *t*-value to be used in calculating the LSD; the degrees of freedom of the source for the `terms` argument will be used. The `error.intervals` argument has been set to `"halfLeast"`, the `meanLSD.type` argument to `"factor.combination"` and the `LSDby` argument to `"Host"` so that the average LSD will be calculated for each Host. This necessary because, under the chosen model, the LSDs differ between Hosts. It results in `lower.halfLeastSignificant.limit` and `upper.halfLeastSignificant.limit` being added to the `predictions` component of the `alldiffs` object.

Or, calculate predictions to check first and then transform to conform to chosen model

```
diffs.full <- predictPlus(current.asrt$asreml.obj,
                          classify = "Host:Ladybird:Cadavers",
                          wald.tab = current.asrt$wald.tab,
                          tables = "none", Vmatrix = TRUE)

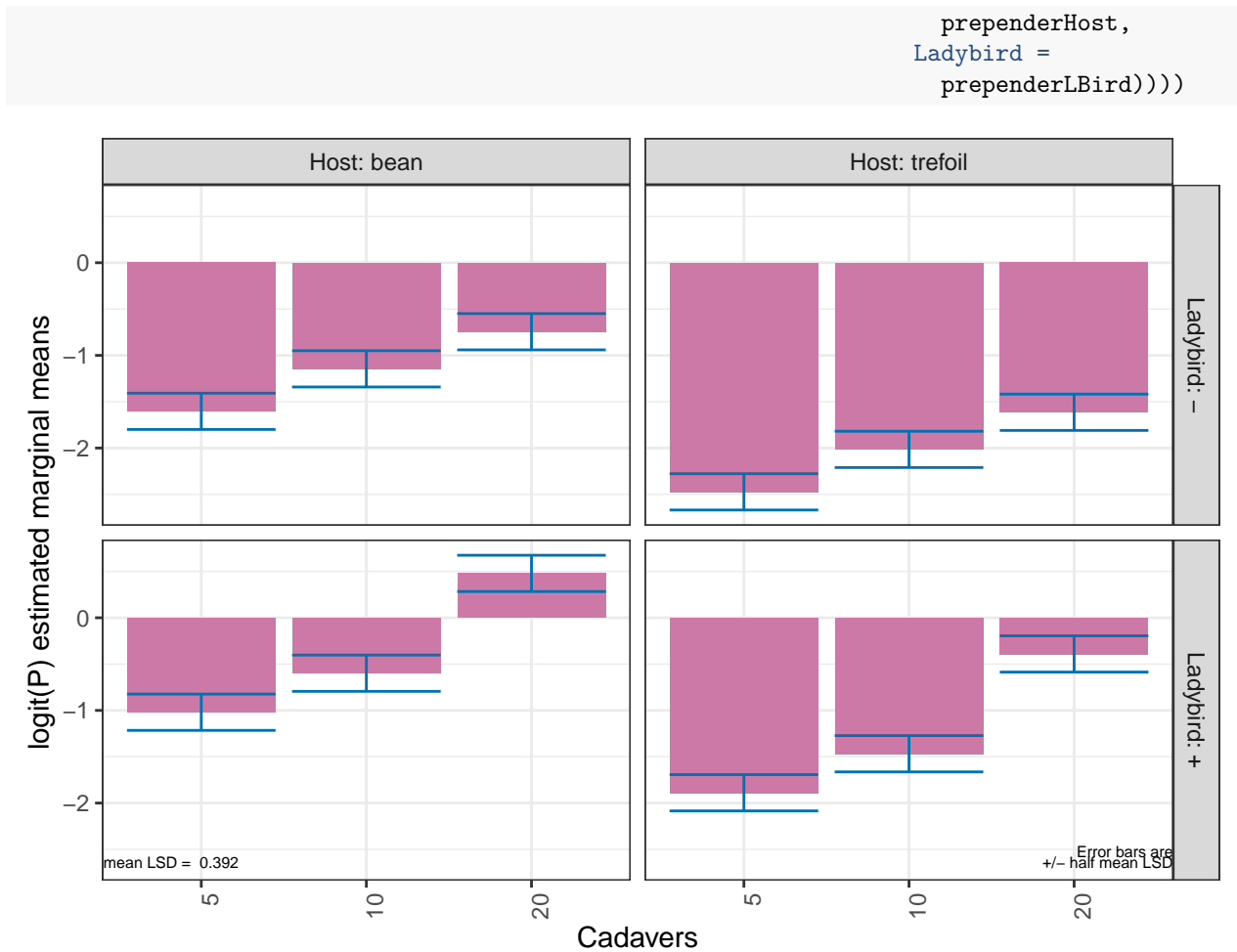
diffs <- linTransform(diffs.full, linear.transformation = ~Cadavers:Ladybird + Host,
                     wald.tab = current.asrt$wald.tab,
                     error.intervals = "halfLeast",
                     meanLSD.type = "factor.combination", LSDby = "Host",
                     tables = "predictions")
```

```
##
##
## ##### Predictions for logitP transform(s) from Host:Ladybird:Cadavers
##
##
## Notes:
## The original predictions, obtained as described below, have
## been linearly transformed to form estimated marginal means.
## - The predictions are obtained by averaging across the hypertable
##   calculated from model terms constructed solely from factors in
##   the averaging and classify sets.
## - Use 'average' to move ignored factors into the averaging set.
```

```
## - The ignored set: Run
##
##
##      Host Ladybird Cadavers predicted.value standard.error upper.halfLeastSignificant.limit
## 1    bean      -         5    -1.6038338      0.1417454      -1.4080222
## 2    bean      -        10    -1.1454308      0.1417454      -0.9496192
## 3    bean      -        20    -0.7448097      0.1417454      -0.5489981
## 4    bean      +         5    -1.0195475      0.1417454      -0.8237359
## 5    bean      +        10    -0.5983440      0.1417454      -0.4025323
## 6    bean      +        20     0.4786704      0.1417454       0.6744820
## 7  trefoil     -         5    -2.4730339      0.1417454      -2.2772223
## 8  trefoil     -        10    -2.0146309      0.1417454      -1.8188193
## 9  trefoil     -        20    -1.6140098      0.1417454      -1.4181982
## 10 trefoil     +         5    -1.8887476      0.1417454      -1.6929360
## 11 trefoil     +        10    -1.4675441      0.1417454      -1.2717325
## 12 trefoil     +        20    -0.3905297      0.1417454      -0.1947181
##      lower.halfLeastSignificant.limit est.status
## 1                                -1.7996454 Estimable
## 2                                -1.3412425 Estimable
## 3                                -0.9406214 Estimable
## 4                                -1.2153592 Estimable
## 5                                -0.7941556 Estimable
## 6                                 0.2828588 Estimable
## 7                               -2.6688455 Estimable
## 8                               -2.2104426 Estimable
## 9                               -1.8098215 Estimable
## 10                              -2.0845593 Estimable
## 11                              -1.6633557 Estimable
## 12                              -0.5863414 Estimable
##
##
## LSD values
##
## minimum LSD =  0.3916233 0.3916233
##
## mean LSD =  0.3916233 0.3916233
##
## maximum LSD =  0.3916233 0.3916233
##
## (sed range / mean sed = 7.09e-16 5.67e-16 )
```

Plot the predictions

```
prependHost <- function(string, prefix = "Host: ") paste0(prefix, string)
prependHost <- as_labeller(prependHost)
prependLBird <- function(string, prefix = "Ladybird: ") paste0(prefix, string)
prependLBird <- as_labeller(prependLBird)
plotPredictions(diffs$predictions, y = "predicted.value",
  y.title = "logit(P) estimated marginal means",
  classify = "Host:Ladybird:Cadavers",
  error.intervals = "halfLeast",
  ggplotFuncs = list(facet_grid(Ladybird ~ Host,
                                labeller = labeller(Host =
```



The function `plotPredictions` uses `ggplot` to produce the plot and the `ggplotFuncs` argument allows the addition of `ggplot` functions to modify the plot. In this case, the `facet.grid` function is respecified to include `prepending` functions that modify the labels of the facets to include the factor names. Note the error bars in the plots are of $\pm 0.5LSD$ so that pairs of prediction with nonoverlapping bars are significantly different (Snee, 1981).

Get and plot the predictions with a single function call

The `predictPresent` function combines the functionality of `predictPlus` and `plotPredictions`, as demonstrated now. Also, the use of `plotPvalues` to plot the pairwise p -values is displayed. The `predictPresent` function has the capability of producing `alldiffs` objects for multiple `terms` and these are stored in a list each of which is named for the term whose `alldiffs` object it stores. Thus, the term has to be specified in referencing components of `diffs`.

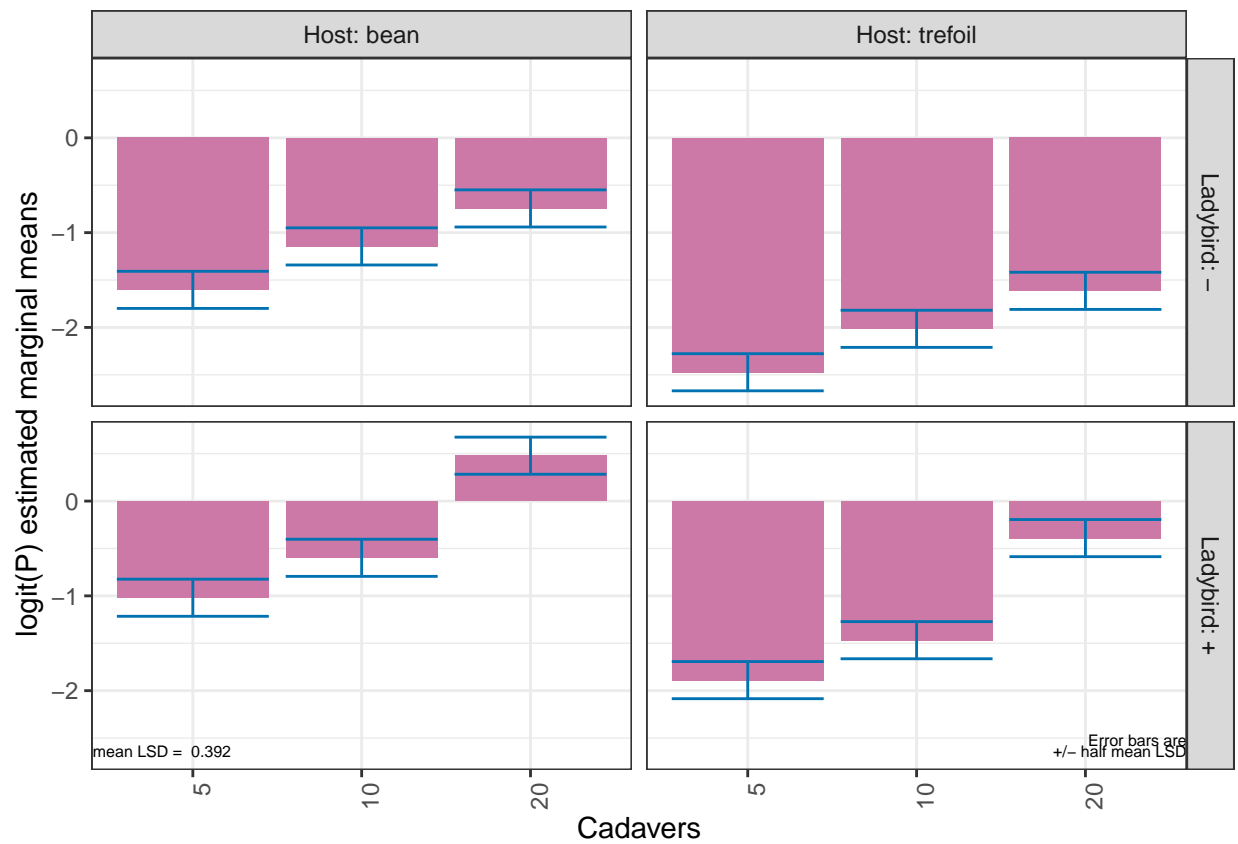
```

prependingHost <- function(string, prefix = "Host: ") paste0(prefix, string)
prependingHost <- as_labeller(prependingHost)
prependingLBird <- function(string, prefix = "Ladybird: ") paste0(prefix, string)
prependingLBird <- as_labeller(prependingLBird)
titl <- "logit(P) estimated marginal means"
names(titl) <- "logitP"
diffs <- predictPresent(current.asrt$asreml.obj,
                        terms = "Host:Ladybird:Cadavers",

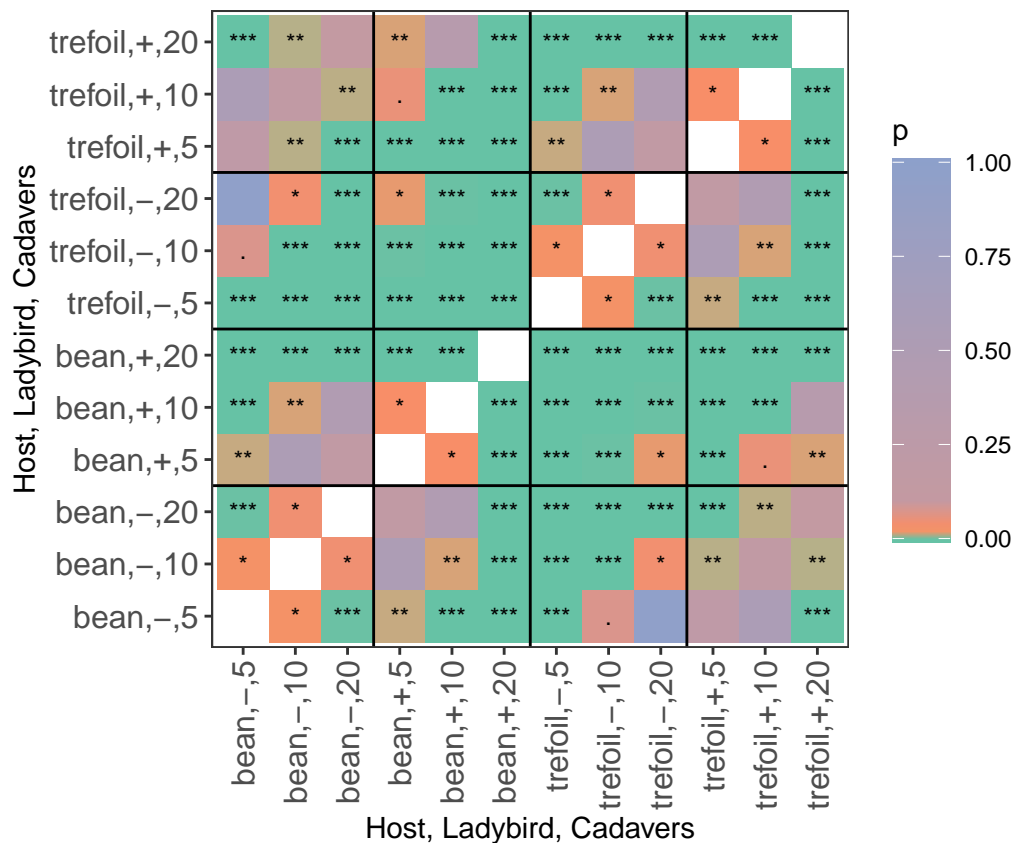
```



```
linear.transformation = ~Cadavers:Ladybird + Host,
titles = titl,
wald.tab = current.asrt$wald.tab,
error.intervals = "halfLeast",
meanLSD.type = "factor.combination", LSDby = "Host",
tables = "none",
ggplotFuncs = list(facet_grid(Ladybird ~ Host,
                             labeller = labeller(Host =
                                                     prependerHost,
                                                     Ladybird =
                                                         prependerLBird))))
```



```
plotPvalues(diffs$Host.Ladybird.Cadavers, factors.per.grid = 1, show.sig = TRUE)
```



```
options(width = 90)
```

```
diffs$Host.Ladybird.Cadavers$differences
```

```
##          bean,-5  bean,-10  bean,-20  bean,+5  bean,+10  bean,+20
## bean,-5      0.0000000 -0.4584030 -0.8590241 -0.5842863 -1.0054898 -2.0825042
## bean,-10     0.45840297 0.0000000 -0.4006211 -0.1258833 -0.5470869 -1.6241012
## bean,-20     0.85902408 0.4006211 0.0000000 0.2747378 -0.1464657 -1.2234801
## bean,+5      0.58428627 0.1258833 -0.2747378 0.0000000 -0.4212036 -1.4982179
## bean,+10     1.00548982 0.5470869 0.1464657 0.4212036 0.0000000 -1.0770144
## bean,+20     2.08250420 1.6241012 1.2234801 1.4982179 1.0770144 0.0000000
## trefoil,-5  -0.86920012 -1.3276031 -1.7282242 -1.4534864 -1.8746899 -2.9517043
## trefoil,-10 -0.41079715 -0.8692001 -1.2698212 -0.9950834 -1.4162870 -2.4933014
## trefoil,-20 -0.01017604 -0.4685790 -0.8692001 -0.5944623 -1.0156659 -2.0926802
## trefoil,+5   -0.28491385 -0.7433168 -1.1439379 -0.8692001 -1.2904037 -2.3674180
## trefoil,+10  0.13628970 -0.3221133 -0.7227344 -0.4479966 -0.8692001 -1.9462145
## trefoil,+20  1.21330408 0.7549011 0.3542800 0.6290178 0.2078143 -0.8692001
##          trefoil,-5 trefoil,-10 trefoil,-20 trefoil,+5 trefoil,+10 trefoil,+20
## bean,-5      0.8692001 0.4107972 0.01017604 0.2849139 -0.1362897 -1.2133041
## bean,-10     1.3276031 0.8692001 0.46857901 0.7433168 0.3221133 -0.7549011
## bean,-20     1.7282242 1.2698212 0.86920012 1.1439379 0.7227344 -0.3542800
## bean,+5      1.4534864 0.9950834 0.59446231 0.8692001 0.4479966 -0.6290178
## bean,+10     1.8746899 1.4162870 1.01566586 1.2904037 0.8692001 -0.2078143
## bean,+20     2.9517043 2.4933014 2.09268024 2.3674180 1.9462145 0.8692001
## trefoil,-5  0.0000000 -0.4584030 -0.85902408 -0.5842863 -1.0054898 -2.0825042
## trefoil,-10 0.4584030 0.0000000 -0.40062111 -0.1258833 -0.5470869 -1.6241012
## trefoil,-20 0.8590241 0.4006211 0.00000000 0.2747378 -0.1464657 -1.2234801
```

```
## trefoil,+,5      0.5842863      0.1258833 -0.27473781      0.0000000      -0.4212036      -1.4982179
## trefoil,+,10     1.0054898      0.5470869      0.14646574      0.4212036      0.0000000      -1.0770144
## trefoil,+,20     2.0825042      1.6241012      1.22348012      1.4982179      1.0770144      0.0000000
```

```
options(width = 90)
print(diffs$Host.Ladybird.Cadavers$sed)
```

```
##          bean,-,5 bean,-,10 bean,-,20 bean,+,5 bean,+,10 bean,+,20 trefoil,-,5
## bean,-,5          NA 0.1957142 0.1957142 0.1957142 0.1957142 0.1957142 0.1129957
## bean,-,10 0.1957142          NA 0.1957142 0.1957142 0.1957142 0.1957142 0.2259913
## bean,-,20 0.1957142 0.1957142          NA 0.1957142 0.1957142 0.1957142 0.2259913
## bean,+,5   0.1957142 0.1957142 0.1957142          NA 0.1957142 0.1957142 0.2259913
## bean,+,10 0.1957142 0.1957142 0.1957142 0.1957142          NA 0.1957142 0.2259913
## bean,+,20 0.1957142 0.1957142 0.1957142 0.1957142 0.1957142          NA 0.2259913
## trefoil,-,5 0.1129957 0.2259913 0.2259913 0.2259913 0.2259913 0.2259913          NA
## trefoil,-,10 0.2259913 0.1129957 0.2259913 0.2259913 0.2259913 0.2259913 0.1957142
## trefoil,-,20 0.2259913 0.2259913 0.1129957 0.2259913 0.2259913 0.2259913 0.1957142
## trefoil,+,5 0.2259913 0.2259913 0.2259913 0.1129957 0.2259913 0.2259913 0.1957142
## trefoil,+,10 0.2259913 0.2259913 0.2259913 0.2259913 0.1129957 0.2259913 0.1957142
## trefoil,+,20 0.2259913 0.2259913 0.2259913 0.2259913 0.2259913 0.1129957 0.1957142
##          trefoil,-,10 trefoil,-,20 trefoil,+,5 trefoil,+,10 trefoil,+,20
## bean,-,5          0.2259913 0.2259913 0.2259913 0.2259913 0.2259913
## bean,-,10 0.1129957 0.2259913 0.2259913 0.2259913 0.2259913
## bean,-,20 0.2259913 0.1129957 0.2259913 0.2259913 0.2259913
## bean,+,5   0.2259913 0.2259913 0.1129957 0.2259913 0.2259913
## bean,+,10 0.2259913 0.2259913 0.2259913 0.1129957 0.2259913
## bean,+,20 0.2259913 0.2259913 0.2259913 0.2259913 0.1129957
## trefoil,-,5 0.1957142 0.1957142 0.1957142 0.1957142 0.1957142
## trefoil,-,10          NA 0.1957142 0.1957142 0.1957142 0.1957142
## trefoil,-,20 0.1957142          NA 0.1957142 0.1957142 0.1957142
## trefoil,+,5 0.1957142 0.1957142          NA 0.1957142 0.1957142
## trefoil,+,10 0.1957142 0.1957142 0.1957142          NA 0.1957142
## trefoil,+,20 0.1957142 0.1957142 0.1957142 0.1957142          NA
```

Perform the analysis with just selected model fitted

The model with nonsignificant fixed terms dropped is obtained in order to compare it with the fit when they are retained and the estimated marginal means for the chosen model are obtained.

```
ns.terms <- current.asrt$test.summary$terms[current.asrt$test.summary$action == "Nonsignificant"]
red.asrt <- changeTerms(current.asrt, dropFixed = paste(ns.terms, collapse = "+"))
```

```
## Calculating denominator DF
## Calculating denominator DF
```

```
summary(red.asrt$asreml.obj)$varcomp
```

```
##          component      std.error    z.ratio bound %ch
## Run      -0.004327123 0.002802858 -1.543825      U 0.8
## units!R   0.223431515 0.039503521  5.655990      P 0.0
```

```
print(red.asrt, which = "pseudoanova")
```

```
##
##
## ##### Pseudo-anova table for fixed terms
```

```
##
##
## Wald tests for fixed effects.
## Response: logitP
##
##           Df denDF   F.inc    Pr
## (Intercept)    1     1 1550.00 0.0162
## Host           1    64   60.88 0.0000
## Cadavers       2    64   38.12 0.0000
## Ladybird       1    64   49.65 0.0000
## Cadavers:Ladybird 2    64    3.88 0.0256
```

```
diffs.red <- predictPlus(red.asrt$asreml.obj,
                          classify = "Host:Ladybird:Cadavers",
                          wald.tab = current.asrt$wald.tab,
                          error.intervals = "halfLeast",
                          meanLSD.type = "factor.combination", LSDby = "Host",
                          tables = "predictions")
```

```
##
##
## #### Predictions for logitP from Host:Ladybird:Cadavers
##
##
## Notes:
## - The predictions are obtained by averaging across the hypertable
##   calculated from model terms constructed solely from factors in
##   the averaging and classify sets.
## - Use 'average' to move ignored factors into the averaging set.
## - The ignored set: Run
##
##
##           Host Ladybird Cadavers predicted.value standard.error
## 1    bean      -         5    -1.6038338      0.1398332
## 2    bean      -        10    -1.1454308      0.1398332
## 3    bean      -        20    -0.7448097      0.1398332
## 4    bean      +         5    -1.0195475      0.1398332
## 5    bean      +        10    -0.5983440      0.1398332
## 6    bean      +        20     0.4786704      0.1398332
## 7  trefoil     -         5    -2.4730339      0.1398332
## 8  trefoil     -        10    -2.0146309      0.1398332
## 9  trefoil     -        20    -1.6140098      0.1398332
## 10 trefoil     +         5    -1.8887476      0.1398332
## 11 trefoil     +        10    -1.4675441      0.1398332
## 12 trefoil     +        20    -0.3905297      0.1398332
##      upper.halfLeastSignificant.limit lower.halfLeastSignificant.limit est.status
## 1                                -1.4107942                        -1.7968734 Estimable
## 2                                -0.9523913                        -1.3384704 Estimable
## 3                                -0.5517702                        -0.9378493 Estimable
## 4                                -0.8265080                        -1.2125871 Estimable
## 5                                -0.4053044                        -0.7913835 Estimable
## 6                                 0.6717100                         0.2856308 Estimable
## 7                                -2.2799944                        -2.6660735 Estimable
## 8                                -1.8215914                        -2.2076705 Estimable
## 9                                -1.4209703                        -1.8070494 Estimable
```

```
## 10                -1.6957081                -2.0817872 Estimable
## 11                -1.2745045                -1.6605837 Estimable
## 12                -0.1974902                -0.5835693 Estimable
##
##
## LSD values
##
## minimum LSD =  0.3860791 0.3860791
##
## mean LSD =  0.3860791 0.3860791
##
## maximum LSD =  0.3860791 0.3860791
##
## (sed range / mean sed =  2.44e-15 2.3e-15 )
options(width = 90)
print(diffs.red$sed)

##          bean,-,5 bean,-,10 bean,-,20  bean,+,5 bean,+,10 bean,+,20 trefoil,-,5
## bean,-,5          NA 0.1929435 0.1929435 0.1929435 0.1929435 0.1929435 0.1113960
## bean,-,10 0.1929435          NA 0.1929435 0.1929435 0.1929435 0.1929435 0.2227920
## bean,-,20 0.1929435 0.1929435          NA 0.1929435 0.1929435 0.1929435 0.2227920
## bean,+,5   0.1929435 0.1929435 0.1929435          NA 0.1929435 0.1929435 0.2227920
## bean,+,10 0.1929435 0.1929435 0.1929435 0.1929435          NA 0.1929435 0.2227920
## bean,+,20 0.1929435 0.1929435 0.1929435 0.1929435 0.1929435          NA 0.2227920
## trefoil,-,5 0.1113960 0.2227920 0.2227920 0.2227920 0.2227920 0.2227920          NA
## trefoil,-,10 0.2227920 0.1113960 0.2227920 0.2227920 0.2227920 0.2227920 0.1929435
## trefoil,-,20 0.2227920 0.2227920 0.1113960 0.2227920 0.2227920 0.2227920 0.1929435
## trefoil,+,5 0.2227920 0.2227920 0.2227920 0.1113960 0.2227920 0.2227920 0.1929435
## trefoil,+,10 0.2227920 0.2227920 0.2227920 0.2227920 0.1113960 0.2227920 0.1929435
## trefoil,+,20 0.2227920 0.2227920 0.2227920 0.2227920 0.2227920 0.1113960 0.1929435
##          trefoil,-,10 trefoil,-,20 trefoil,+,5 trefoil,+,10 trefoil,+,20
## bean,-,5          0.2227920 0.2227920 0.2227920 0.2227920 0.2227920
## bean,-,10 0.1113960 0.2227920 0.2227920 0.2227920 0.2227920
## bean,-,20 0.2227920 0.1113960 0.2227920 0.2227920 0.2227920
## bean,+,5   0.2227920 0.2227920 0.1113960 0.2227920 0.2227920
## bean,+,10 0.2227920 0.2227920 0.2227920 0.1113960 0.2227920
## bean,+,20 0.2227920 0.2227920 0.2227920 0.2227920 0.1113960
## trefoil,-,5 0.1929435 0.1929435 0.1929435 0.1929435 0.1929435
## trefoil,-,10          NA 0.1929435 0.1929435 0.1929435 0.1929435
## trefoil,-,20 0.1929435          NA 0.1929435 0.1929435 0.1929435
## trefoil,+,5 0.1929435 0.1929435          NA 0.1929435 0.1929435
## trefoil,+,10 0.1929435 0.1929435 0.1929435          NA 0.1929435
## trefoil,+,20 0.1929435 0.1929435 0.1929435 0.1929435          NA
```

Using lmerTest and emmeans to get the predictions and associated statistics.

```
library(lmerTest)
```

```
## Loading required package: lme4
```

```
## Warning: package 'lme4' was built under R version 3.5.2
```

```
##
## Attaching package: 'lmerTest'

## The following object is masked from 'package:lme4':
##
##      lmer

## The following object is masked from 'package:stats':
##
##      step

library(emmeans)

## Warning: package 'emmeans' was built under R version 3.5.2
m1.lmer <- lmerTest::lmer(logitP ~ Host*Cadavers*Ladybird + (1|Run),
                        data=Ladybird.dat)

## singular fit
HCL.emm <- emmeans::emmeans(m1.lmer, specs = ~ Host:Cadavers:Ladybird)
HCL.preds <- summary(HCL.emm)
den.df <- min(HCL.preds$df)
HCL.vcov <- vcov(HCL.emm)
HCL.sed <- NULL
```

Modify HCL.preds to be compatible with a predictions.frame

```
HCL.preds <- as.predictions.frame(HCL.preds, predictions = "emmean",
                                se = "SE", interval.type = "CI",
                                interval.names = c("lower.CL", "upper.CL"))
```

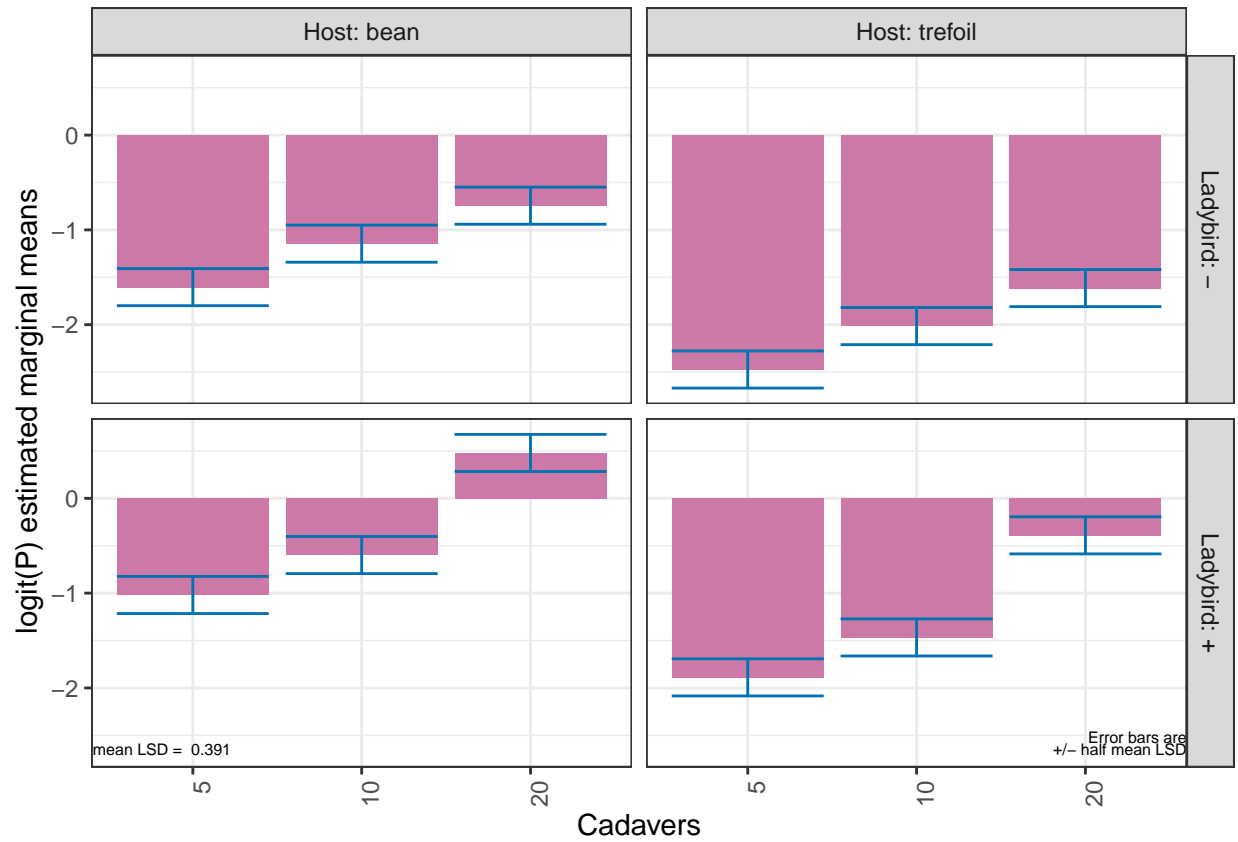
Form an alldiffs object with predictions obtained with lmerTest

```
HCL.diffs <- allDifferences(predictions = HCL.preds, classify = "Host:Ladybird:Cadavers",
                          sed = HCL.sed, vcov = HCL.vcov, tdf = den.df)
```

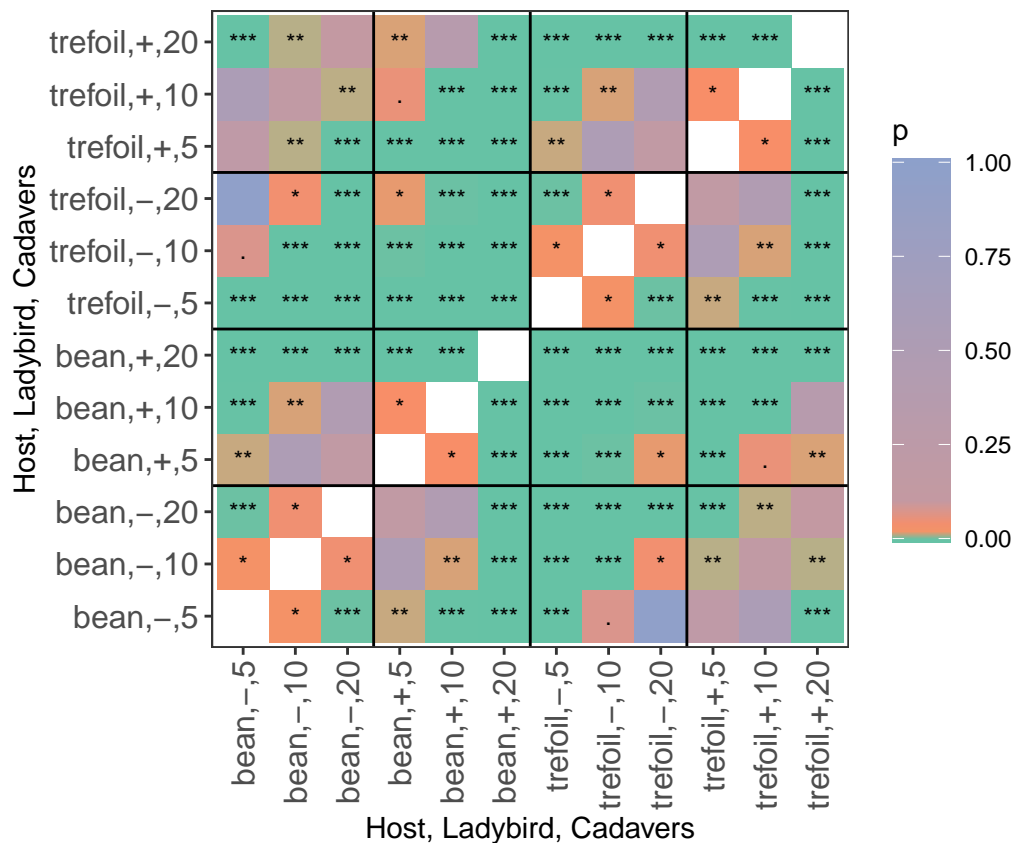
Transform and plot the predictions

```
diffs <- linTransform(HCL.diffs, linear.transformation = ~Cadavers:Ladybird + Host,
                    error.intervals = "halfLeast",
                    meanLSD.type = "factor.combination", LSDby = "Host",
                    tables = "none")
prependHost <- function(string, prefix = "Host: ") paste0(prefix, string)
prependHost <- as_labeller(prependHost)
prependLBird <- function(string, prefix = "Ladybird: ") paste0(prefix, string)
prependLBird <- as_labeller(prependLBird)
plotPredictions(diffs$predictions, y = "predicted.value",
               y.title = "logit(P) estimated marginal means",
               classify = "Host:Ladybird:Cadavers",
               error.intervals = "halfLeast",
               meanLSD.type = "factor.combination", LSDby = "Host",
```

```
ggplotFuncs = list(facet_grid(Ladybird ~ Host,
                             labeller = labeller(Host =
                                                    prependerHost,
                                                    Ladybird =
                                                    prependerLBird))))
```



```
plotPvalues(diffs, factors.per.grid = 1, show.sig = TRUE)
```



```
options(width = 90)
print(diffs$used)
```

```
##          bean,-,5 bean,-,10 bean,-,20  bean,+,5 bean,+,10 bean,+,20 trefoil,-,5
## bean,-,5          NA 0.1945600 0.1945600 0.1945600 0.1945600 0.1945600 0.1123293
## bean,-,10 0.1945600          NA 0.1945600 0.1945600 0.1945600 0.1945600 0.2246586
## bean,-,20 0.1945600 0.1945600          NA 0.1945600 0.1945600 0.1945600 0.2246586
## bean,+,5   0.1945600 0.1945600 0.1945600          NA 0.1945600 0.1945600 0.2246586
## bean,+,10 0.1945600 0.1945600 0.1945600 0.1945600          NA 0.1945600 0.2246586
## bean,+,20 0.1945600 0.1945600 0.1945600 0.1945600 0.1945600          NA 0.2246586
## trefoil,-,5 0.1123293 0.2246586 0.2246586 0.2246586 0.2246586 0.2246586          NA
## trefoil,-,10 0.2246586 0.1123293 0.2246586 0.2246586 0.2246586 0.2246586 0.1945600
## trefoil,-,20 0.2246586 0.2246586 0.1123293 0.2246586 0.2246586 0.2246586 0.1945600
## trefoil,+,5 0.2246586 0.2246586 0.2246586 0.1123293 0.2246586 0.2246586 0.1945600
## trefoil,+,10 0.2246586 0.2246586 0.2246586 0.2246586 0.1123293 0.2246586 0.1945600
## trefoil,+,20 0.2246586 0.2246586 0.2246586 0.2246586 0.2246586 0.1123293 0.1945600
##          trefoil,-,10 trefoil,-,20 trefoil,+,5 trefoil,+,10 trefoil,+,20
## bean,-,5          0.2246586 0.2246586 0.2246586 0.2246586 0.2246586
## bean,-,10 0.1123293 0.2246586 0.2246586 0.2246586 0.2246586
## bean,-,20 0.2246586 0.1123293 0.2246586 0.2246586 0.2246586
## bean,+,5   0.2246586 0.2246586 0.1123293 0.2246586 0.2246586
## bean,+,10 0.2246586 0.2246586 0.2246586 0.1123293 0.2246586
## bean,+,20 0.2246586 0.2246586 0.2246586 0.2246586 0.1123293
## trefoil,-,5 0.1945600 0.1945600 0.1945600 0.1945600 0.1945600
## trefoil,-,10          NA 0.1945600 0.1945600 0.1945600 0.1945600
## trefoil,-,20 0.1945600          NA 0.1945600 0.1945600 0.1945600
```


## trefoil,+,5	0.1945600	0.1945600	NA	0.1945600	0.1945600
## trefoil,+,10	0.1945600	0.1945600	0.1945600	NA	0.1945600
## trefoil,+,20	0.1945600	0.1945600	0.1945600	0.1945600	NA

References

- Brien, C. J. (2018) **dae**: functions useful in the design and anova of experiments. version 3.0-23. <https://CRAN.R-project.org/package=dae>.
- Littell, R. C., Milliken, G. A., Stroup, W. W., Wolfinger, R. D., & Schabenberger, O. (2006). *SAS for Mixed Models* (2nd ed.). Cary, N.C.: SAS Press.
- Snee, R. D. (1981). Graphical Display and Assessment of Means. *Biometrics*, **37**, 835-836.
- Welham, S. J., Gezan, S. A., Clark, S. J., & Mead, A. (2014). *Statistical Methods in Biology: Design and Analysis of Experiments and Regression*. Boca Raton: Chapman and Hall/CRC.