Exp35 2015 data analysis

JBH

May 23, 2017

```
library(asreml)
```

```
## Loading required package: lattice

## Licensed to: North Carolina State University
## Serial Number: 40206269 Expires: 30-nov-2017 (190 days)
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
setwd("M:/File_Server/FieldBooks/Data15/Exp35")
```

Get ear rot, fungal density, and fumonisin data

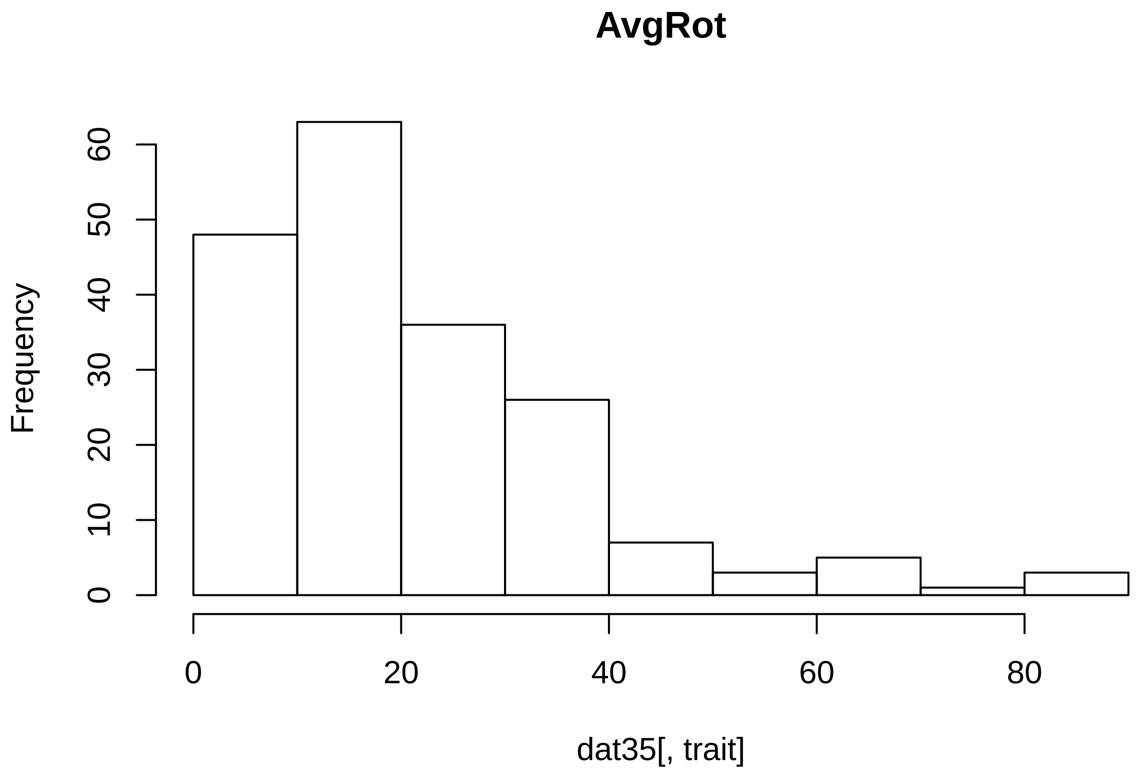
I noticed a couple of plots with no ear counts and no ear rot values, but Laura had fumonisin data, I am knocking those out.

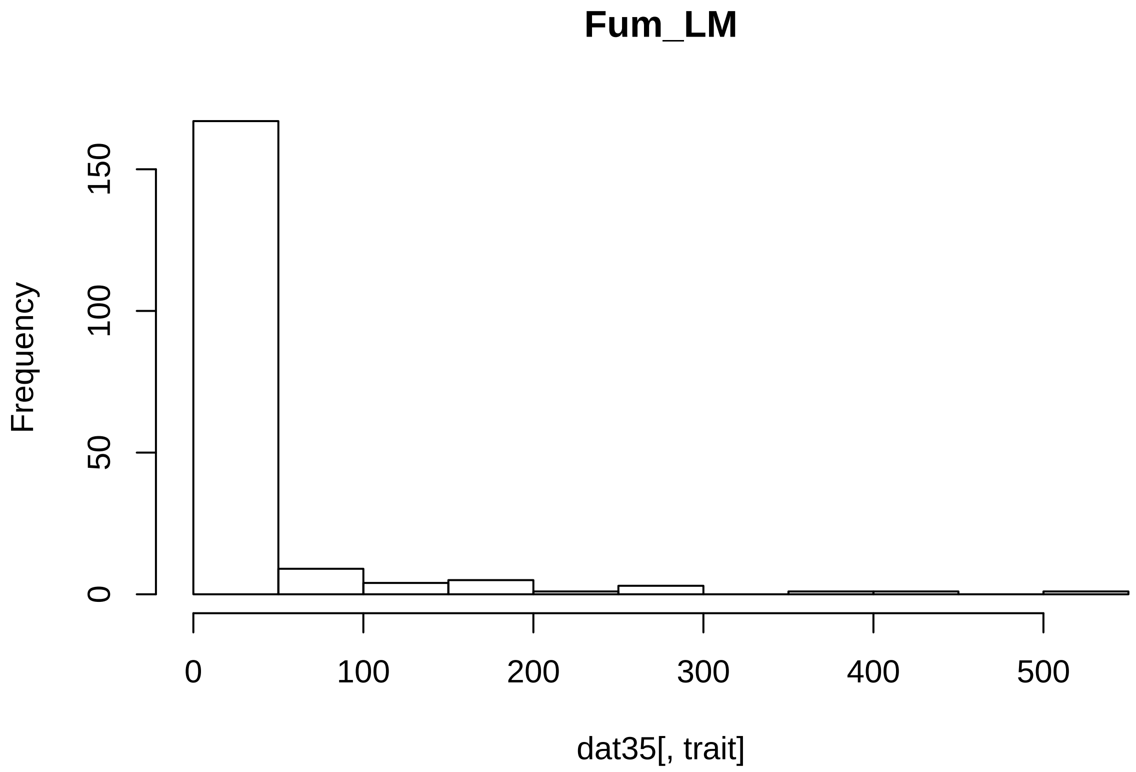
```
dat35 = read.csv("Exp35_2015_rot_density_fum.csv")
dat35 = mutate(dat35, Fum_LM = FUMppb/1000,
              Rep = as.factor(Rep), Block = as.factor(Block),
              Fum_LM = ifelse(is.na(NoEars), NA, Fum_LM)) %>% rename(Fum_TM = fum_ppm_TPM) %>% select(
tbl_df(dat35)
```

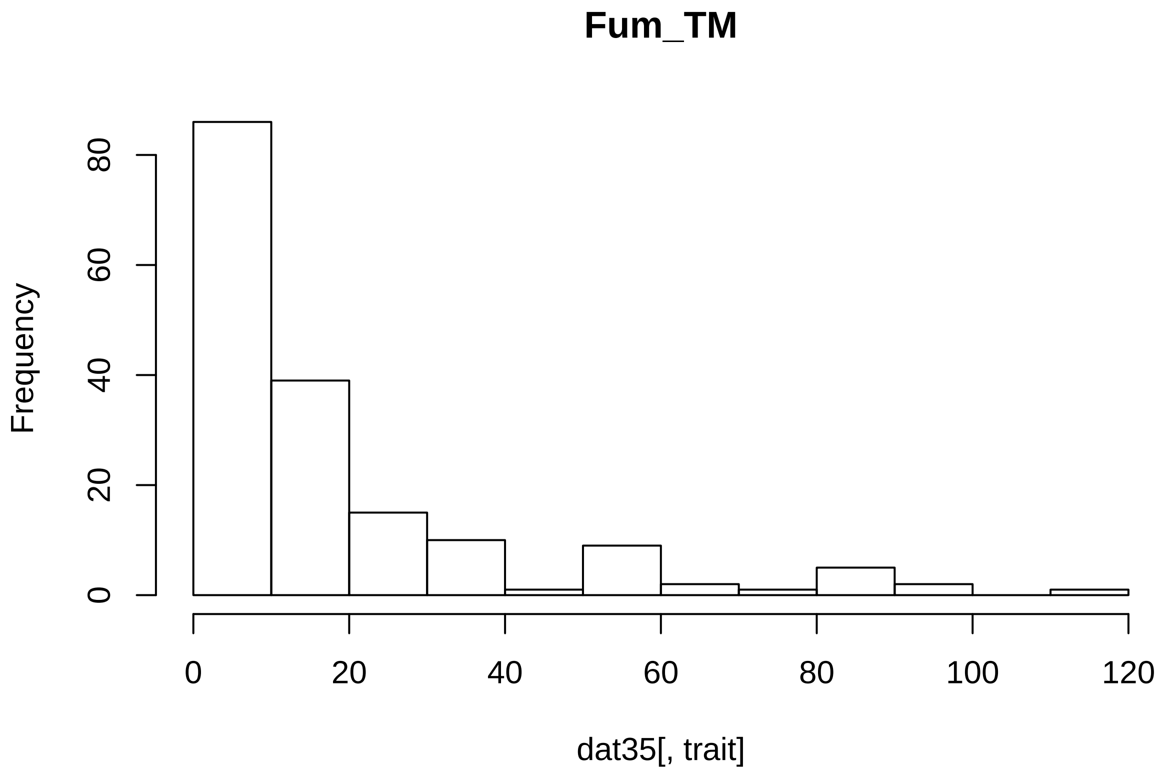
```
## # A tibble: 194 × 9
##       Rep  Block  Plot   Entry InoculationMethod NoEars    AvgRot
##    <fctr> <fctr> <int>  <fctr>            <fctr>  <int>     <dbl>
## 1       1      1     1   NC526           Syringe     12 22.083333
## 2       1      1     1   NC526         Toothpick     10  9.000000
## 3       1      1     2   NC538           Syringe      6 21.666667
## 4       1      1     2   NC538         Toothpick      4 15.000000
## 5       1      1     3 NCG1511           Syringe     11 15.909091
## 6       1      1     3 NCG1511         Toothpick      9  2.222222
## 7       1      1     4 NCG1514           Syringe     10 11.000000
## 8       1      1     4 NCG1514         Toothpick     10 10.000000
## 9       1      1     5 NCG1516           Syringe      9 24.444444
## 10      1      1     5 NCG1516         Toothpick      8 30.000000
## # ... with 184 more rows, and 2 more variables: Fum_LM <dbl>, Fum_TM <dbl>
```

Check histograms of raw data

```r
for (trait in c("AvgRot", "Fum_LM", "Fum_TM")){
  hist(dat35[,trait], main = trait)
}
```


AvgRot
Frequency
0
10
20
30
40
50
60
0
20
40
60
80
dat35[, trait]


Fum_LM
Frequency
0
50
100
150
0
100
200
300
400
500
dat35[, trait]



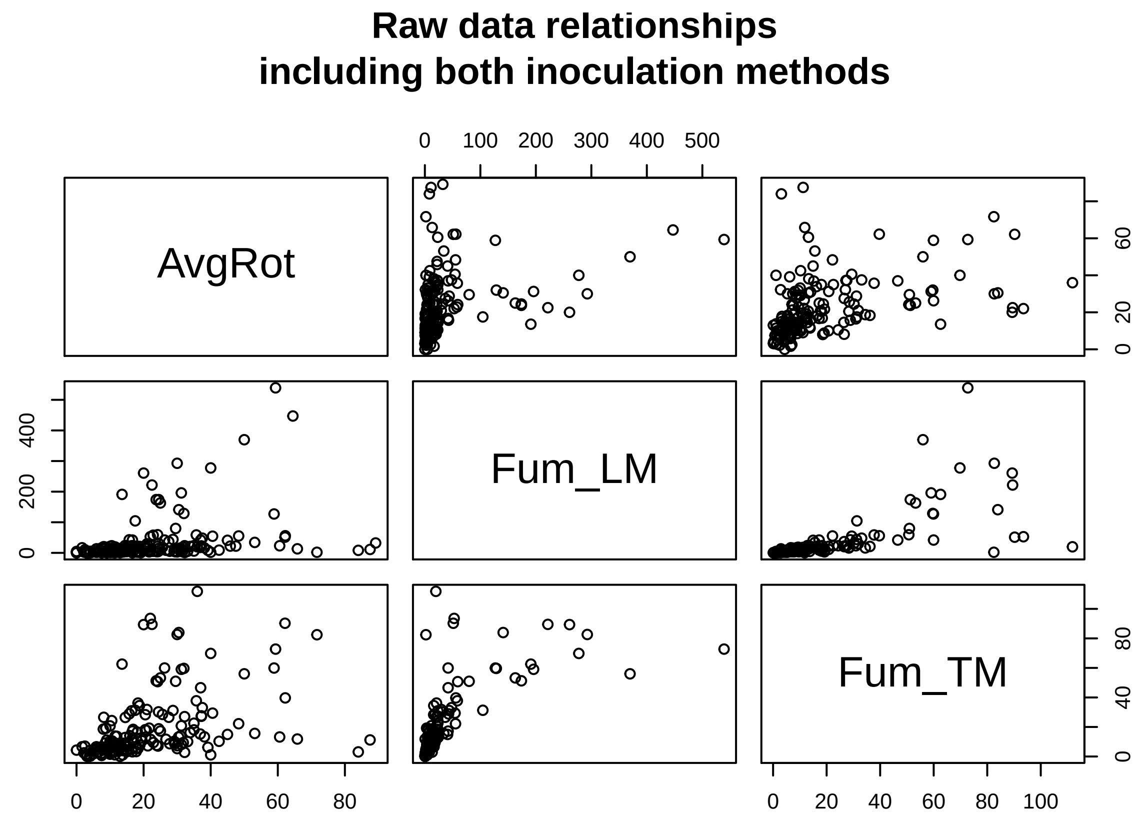
Compare Laura vs. Thiago ppm overall and each against mean ear rot

```
cor(dat35[,c("AvgRot", "Fum_LM", "Fum_TM")], use = "pairwise.complete.obs")
```

```
##           AvgRot    Fum_LM    Fum_TM
## AvgRot 1.0000000 0.3510503 0.4315838
## Fum_LM 0.3510503 1.0000000 0.6804028
## Fum_TM 0.4315838 0.6804028 1.0000000
```

Neither fumonisin value is very highly correlated with ear rot, but fum__ppm__TPM is a little better.

```
pairs(dat35[,c("AvgRot", "Fum_LM", "Fum_TM")], main = "Raw data relationships\nincluding both inoculati
```



Compare toothpick inoculations to backpack inoculations

```
dat35.syr = filter(dat35, InoculationMethod == "Syringe") %>%
  rename(No_Ears_syr = NoEars, Rot_syr = AvgRot, Fum_LM_syr = Fum_LM, Fum_TM_syr = Fum_TM) %>% select(-I

dat35.tooth = filter(dat35, InoculationMethod == "Toothpick") %>%
  rename(No_Ears_tooth = NoEars, Rot_tooth = AvgRot, Fum_LM_tooth = Fum_LM, Fum_TM_tooth = Fum_TM) %>% s

dat35.wide = merge(dat35.syr, dat35.tooth, all = T)
```

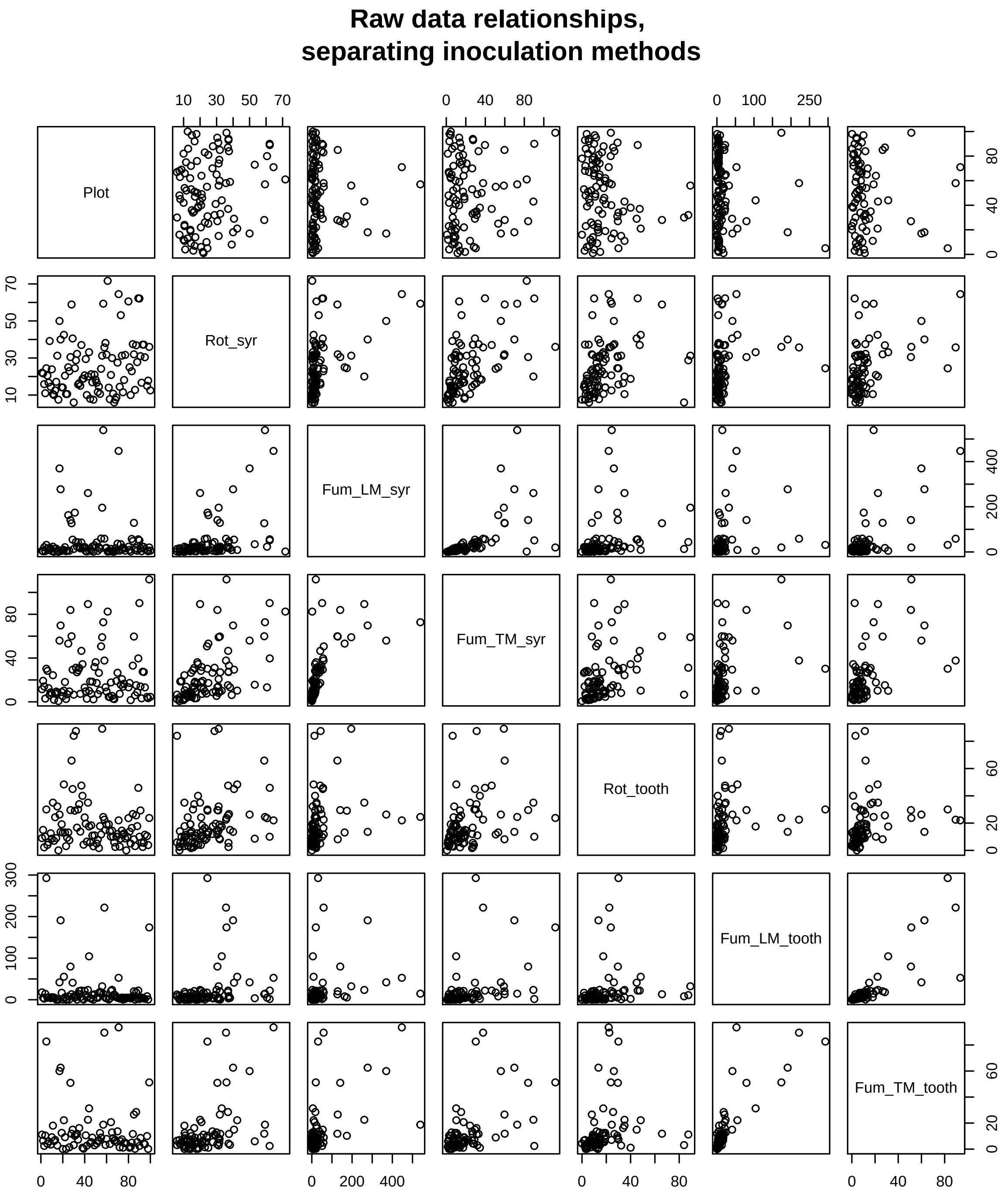
Check the correlations and plot them

```
cor(select(dat35.wide, -c(Rep, Block, Plot, Entry, No_Ears_syr, No_Ears_tooth)), use = "pairwise.comple
```

```
##                Rot_syr Fum_LM_syr Fum_TM_syr Rot_tooth Fum_LM_tooth
## Rot_syr      1.0000000  0.4679414  0.5700181 0.3158904    0.2281510
## Fum_LM_syr   0.4679414  1.0000000  0.6147534 0.2422867    0.2110726
## Fum_TM_syr   0.5700181  0.6147534  1.0000000 0.3249242    0.3994333
## Rot_tooth    0.3158904  0.2422867  0.3249242 1.0000000    0.1643693
## Fum_LM_tooth 0.2281510  0.2110726  0.3994333 0.1643693    1.0000000
## Fum_TM_tooth 0.4497670  0.5361814  0.5036511 0.2046750    0.8300721
##              Fum_TM_tooth
## Rot_syr         0.4497670
## Fum_LM_syr      0.5361814
## Fum_TM_syr      0.5036511
## Rot_tooth        0.2046750
## Fum_LM_tooth     0.8300721
## Fum_TM_tooth     1.0000000
```

The best correlations are between the two fumonisin measures with toothpick inoculation or with syringe, but not across methods!

```
pairs(select(dat35.wide, -c(Rep, Block, Entry, No_Ears_syr, No_Ears_tooth)), main = "Raw data relations
```



It really looks like outliers are driving these relationships.

Ear rot raw data analysis

Make a function to generate initial analysis results on raw data.

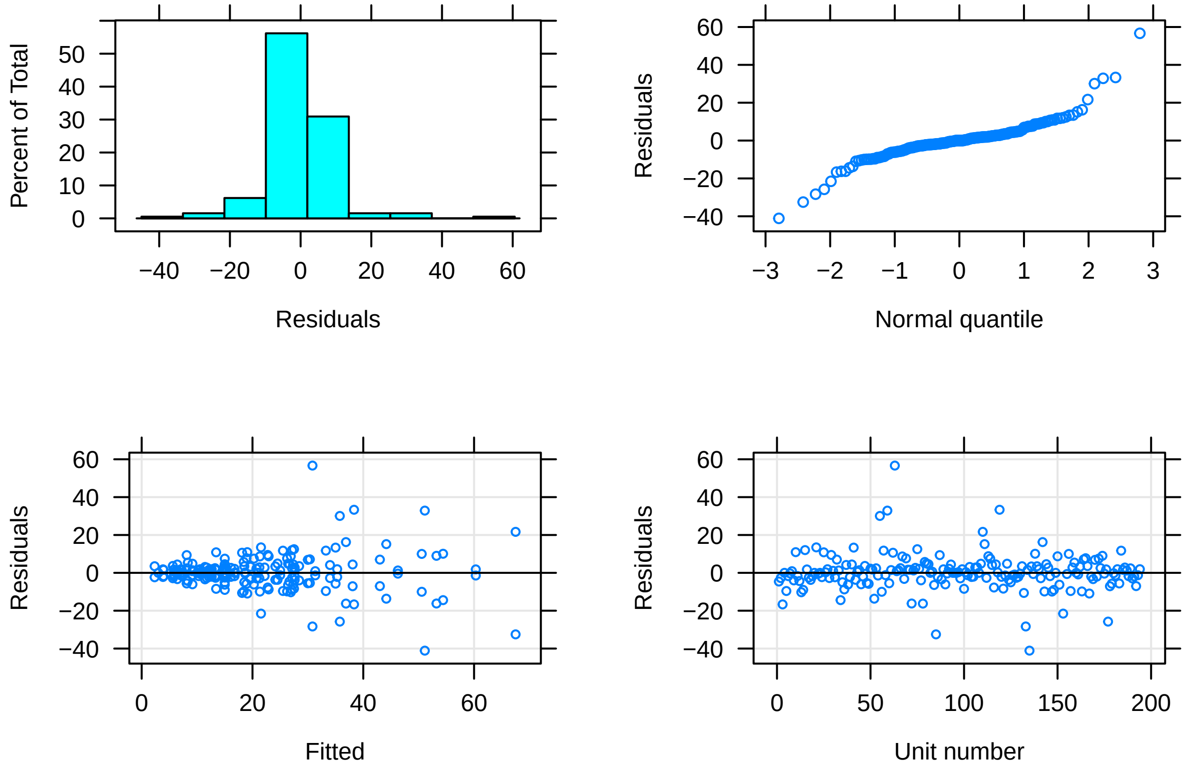
Note, I would like to use this form of wald test:
wald.asreml(mod,denDF=“default”,ssType=“conditional”)
But it returns an error, so inside the function (not outside); we will use wald.asreml(mod) instead It returns Type I SS, but good enough for now.

Also, not the use of assign() here, this is to work around a problem referencing form object later in the function call.

see this: https://www.vsni.co.uk/forum/viewtopic.php?t=1081

```
exp35_lmm = function(data, trait){
  form = as.formula(paste(trait, "~ Entry*InoculationMethod"))
  assign("form", form, envir = globalenv() )
  mod = asreml(fixed = form, random = ~ Rep + Block:Rep, weights = NoEars, data = data, na.method.X='om

  print(summary(mod))
  print(wald.asreml(mod, denDF = "default", ssType = "conditional", data = data))
  plot(mod)

  return(mod)
}

rot.raw = exp35_lmm(dat35, 'AvgRot')
```

```
## ASReml: Wed May 24 12:51:57 2017
##
##      LogLik         S2      DF      wall     cpu
##   -1046.4175    868.7356    92  12:51:57     0.0 (2 restrained)
##   -1040.3049    998.6086    92  12:51:57     0.0 (2 restrained)
##   -1039.4089   1031.6323    92  12:51:57     0.0 (1 restrained)
##   -1039.4089   1031.6323    92  12:51:57     0.0 (2 restrained)
##   -1039.2811   1034.1747    92  12:51:57     0.0 (1 restrained)
##   -1039.2811   1034.1747    92  12:51:57     0.0 (2 restrained)
##   -1039.2719   1034.3378    92  12:51:57     0.0 (1 restrained)
##   -1039.2713   1034.3482    92  12:51:57     0.0
##   -1039.2713   1034.3482    92  12:51:57     0.0
##   -1039.2713   1034.3482    92  12:51:57     0.0
##
## Finished on: Wed May 24 12:51:57 2017
##
## LogLikelihood Converged
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep, data = data,
##     weights = NoEars, na.method.X = "omit")
##
## $loglik
## [1] -1039.271
##
## $nedf
## [1] 92
##
## $sigma
## [1] 32.16128
##
## $varcomp
##                            gamma    component    std.error z.ratio
## Rep!Rep.var         1.011929e-07 1.046687e-04 1.543256e-05 6.78233
## Block:Rep!Block.var 1.011929e-07 1.046687e-04 1.543256e-05 6.78233
## R!variance          1.000000e+00 1.034348e+03 1.525063e+02 6.78233
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## ASReml: Wed May 24 12:51:57 2017
##
##      LogLik         S2      DF      wall     cpu
##    -306.3716   1034.3482    92  12:51:57     0.0
##    -306.3716   1034.3482    92  12:51:57     0.0
##    -306.3716   1034.3482    92  12:51:57     0.0
##    -306.3716   1034.3482    92  12:51:57     0.0
##  Warning: 2 negative weights may have been treated as zero
##
## Finished on: Wed May 24 12:51:57 2017
##
## LogLikelihood Converged
## $Wald
##                         Df denDF   F.inc   F.con Margin           Pr
## (Intercept)              1    92 650.300 650.300        1.714062e-43
## Entry                   49    92   3.196   3.209      A 7.058645e-07
## InoculationMethod        1    92  22.350  22.350      A 8.160245e-06
## Entry:InoculationMethod 49    92   1.114   1.114      B 3.232881e-01
##
## $stratumVariances
##           df Variance Block:Rep
## Block:Rep 92 1034.348         1
```

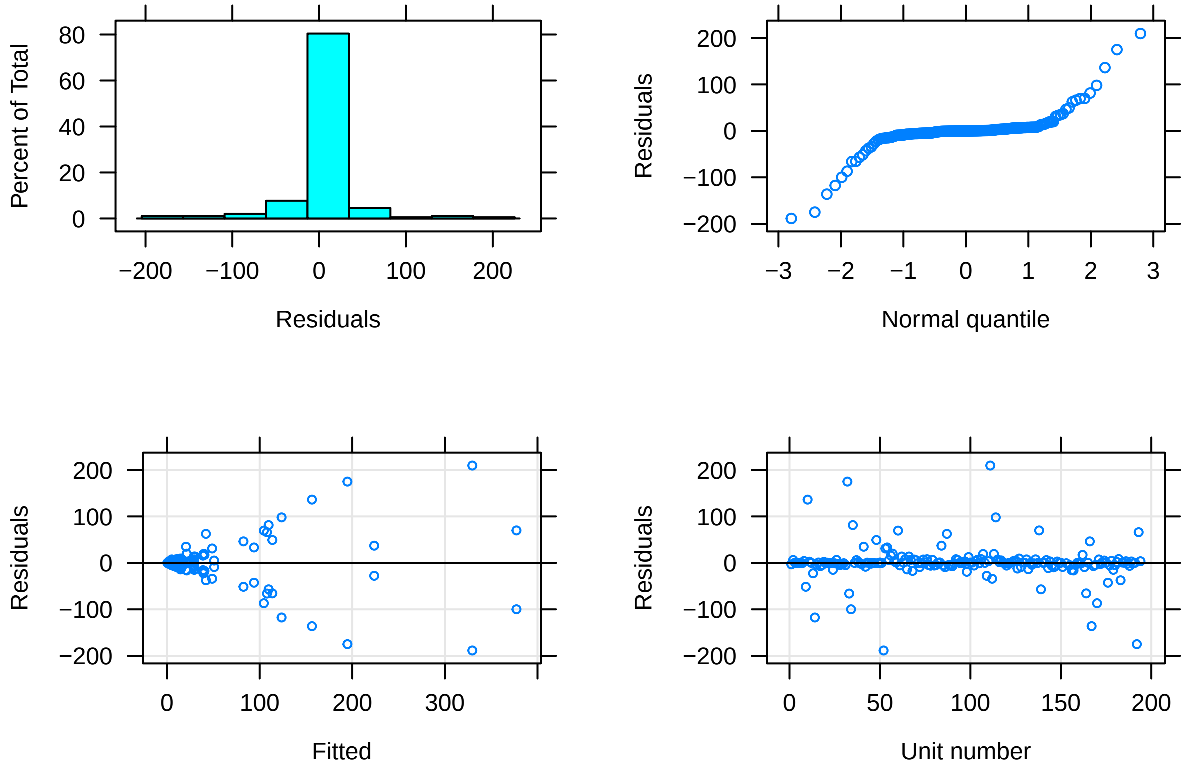


## Analysis of raw fumonisin Laura data

```
Fum_LM.raw = exp35_lmm(dat35, 'Fum_LM')
```

```
## ASReml: Wed May 24 12:51:58 2017
##
##      LogLik         S2      DF       wall      cpu
##   -1190.3567  20024.2158    92  12:51:58     0.0 (2 restrained)
##   -1181.7841  21819.1495    92  12:51:58     0.0 (2 restrained)
##   -1180.5590  22380.0032    92  12:51:58     0.0 (1 restrained)
##   -1180.5590  22380.0032    92  12:51:58     0.0 (2 restrained)
##   -1180.4162  22427.8733    92  12:51:58     0.0 (1 restrained)
##   -1180.4162  22427.8733    92  12:51:58     0.0 (2 restrained)
##   -1180.4062  22431.0167    92  12:51:58     0.0 (1 restrained)
##   -1180.4056  22431.2160    92  12:51:58     0.0
##   -1180.4056  22431.2160    92  12:51:58     0.0
##   -1180.4056  22431.2160    92  12:51:58     0.0
##
## Finished on: Wed May 24 12:51:58 2017
##
## LogLikelihood Converged
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep, data = data,
##     weights = NoEars, na.method.X = "omit")
##
## $loglik
## [1] -1180.406
```

```
##
## $nedf
## [1] 92
##
## $sigma
## [1] 149.7705
##
## $varcomp
##                            gamma    component    std.error z.ratio
## Rep!Rep.var         1.011929e-07 2.269880e-03 3.346755e-04 6.78233
## Block:Rep!Block.var 1.011929e-07 2.269880e-03 3.346755e-04 6.78233
## R!variance          1.000000e+00 2.243122e+04 3.307302e+03 6.78233
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## ASReml: Wed May 24 12:51:58 2017
##
##      LogLik         S2      DF      wall     cpu
##    -447.8990  22431.2160    92  12:51:58     0.0
##    -447.8990  22431.2160    92  12:51:58     0.0
##    -447.8990  22431.2160    92  12:51:58     0.0
##    -447.8990  22431.2160    92  12:51:58     0.0
##  Warning: 2 negative weights may have been treated as zero
##
## Finished on: Wed May 24 12:51:58 2017
##
## LogLikelihood Converged
## $Wald
##                         Df denDF  F.inc  F.con Margin           Pr
## (Intercept)              1    92 65.580 65.580        2.252106e-12
## Entry                   49    92  3.381  3.409      A 1.959904e-07
## InoculationMethod        1    92 10.910 10.910      A 1.363165e-03
## Entry:InoculationMethod 49    92  1.544  1.544      B 3.695050e-02
##
## $stratumVariances
## NULL
```

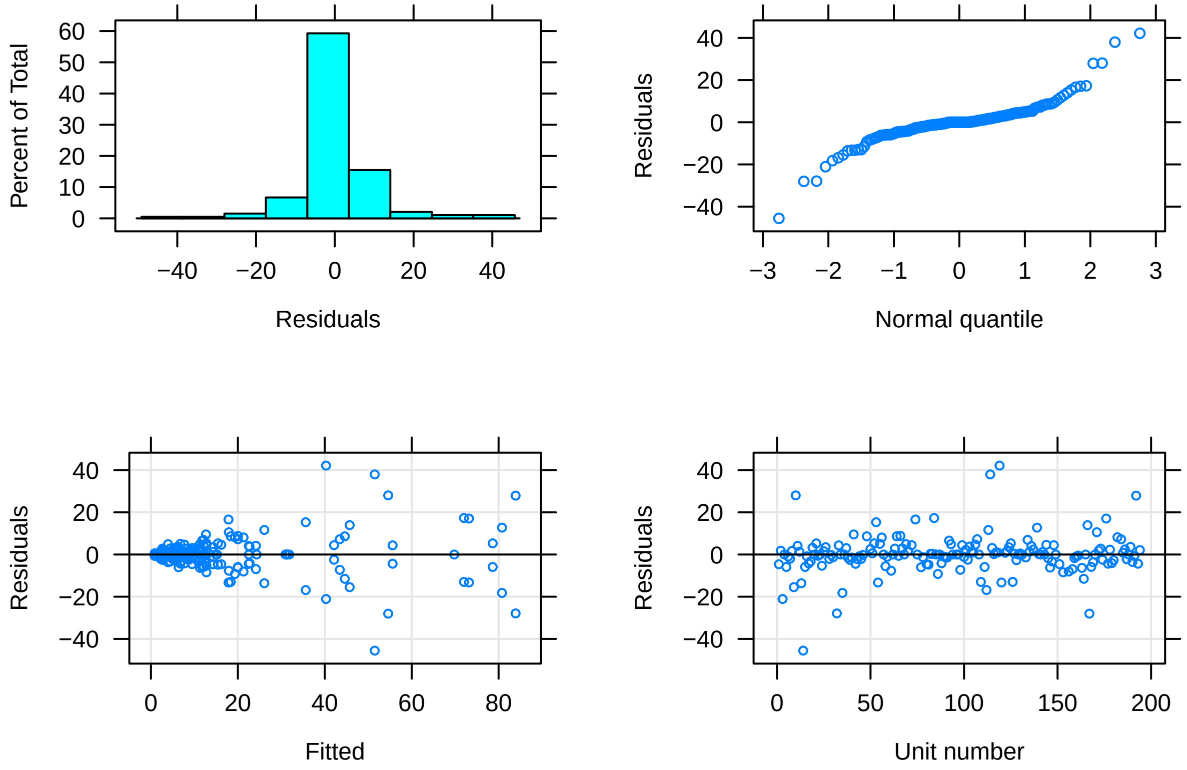


Analysis of raw fumonisin Thiago data

```
Fum_TM.raw = exp35_lmm(dat35, 'Fum_TM')
```

```
## ASReml: Wed May 24 12:51:58 2017
##
##      LogLik        S2      DF      wall     cpu
##    -990.4506   1045.6535   74  12:51:58     0.0 (2 restrained)
##    -984.1526   1198.5530   74  12:51:58     0.0 (2 restrained)
##    -983.2025   1234.8954   74  12:51:58     0.0 (1 restrained)
##    -983.1304   1239.1732   74  12:51:58     0.0 (1 restrained)
##    -983.1304   1239.1732   74  12:51:58     0.0 (2 restrained)
##    -983.1680   1237.4481   74  12:51:58     0.0 (1 restrained)
##    -983.1283   1239.3425   74  12:51:58     0.0 (1 restrained)
##    -983.1259   1239.4888   74  12:51:58     0.0 (1 restrained)
##    -983.1258   1239.4982   74  12:51:58     0.0
##    -983.1258   1239.4982   74  12:51:58     0.0
##    -983.1258   1239.4982   74  12:51:58     0.0
##
## Finished on: Wed May 24 12:51:58 2017
##
## LogLikelihood Converged
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep, data = data,
##     weights = NoEars, na.method.X = "omit")
```

```
##
## $loglik
## [1] -983.1258
##
## $nedf
## [1] 74
##
## $sigma
## [1] 35.20651
##
## $varcomp
##                            gamma    component    std.error  z.ratio
## Rep!Rep.var         1.011929e-07 1.254284e-04 2.062030e-05 6.082763
## Block:Rep!Block.var 1.011929e-07 1.254284e-04 2.062030e-05 6.082763
## R!variance          1.000000e+00 1.239498e+03 2.037723e+02 6.082763
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## ASReml: Wed May 24 12:51:58 2017
##
##      LogLik         S2      DF      wall     cpu
##    -239.8451   1256.4777    73  12:51:58     0.0
## Logliklihood decreased to    -252.40 - trying again with reduced updates
## Logliklihood decreased to    -252.40 - trying again with reduced updates
## Terminating with nfault = 3
##  Warning: 2 negative weights may have been treated as zero
##
## Finished on: Wed May 24 12:51:58 2017
##
## Convergence failed
## $Wald
##                         Df denDF   F.inc   F.con Margin           Pr
## (Intercept)              1    74 339.400 339.400        2.304981e-29
## Entry                   49    74   6.219   6.306      A 1.079280e-12
## InoculationMethod        1    74  20.780  20.780      A 1.994733e-05
## Entry:InoculationMethod 46    74   1.328   1.328      B 1.369035e-01
##
## $stratumVariances
##            df Variance R!variance
## R!variance 74 1239.498          1
```



Looks like we need a transformation of the data. Maybe a log transformation will help all of the traits

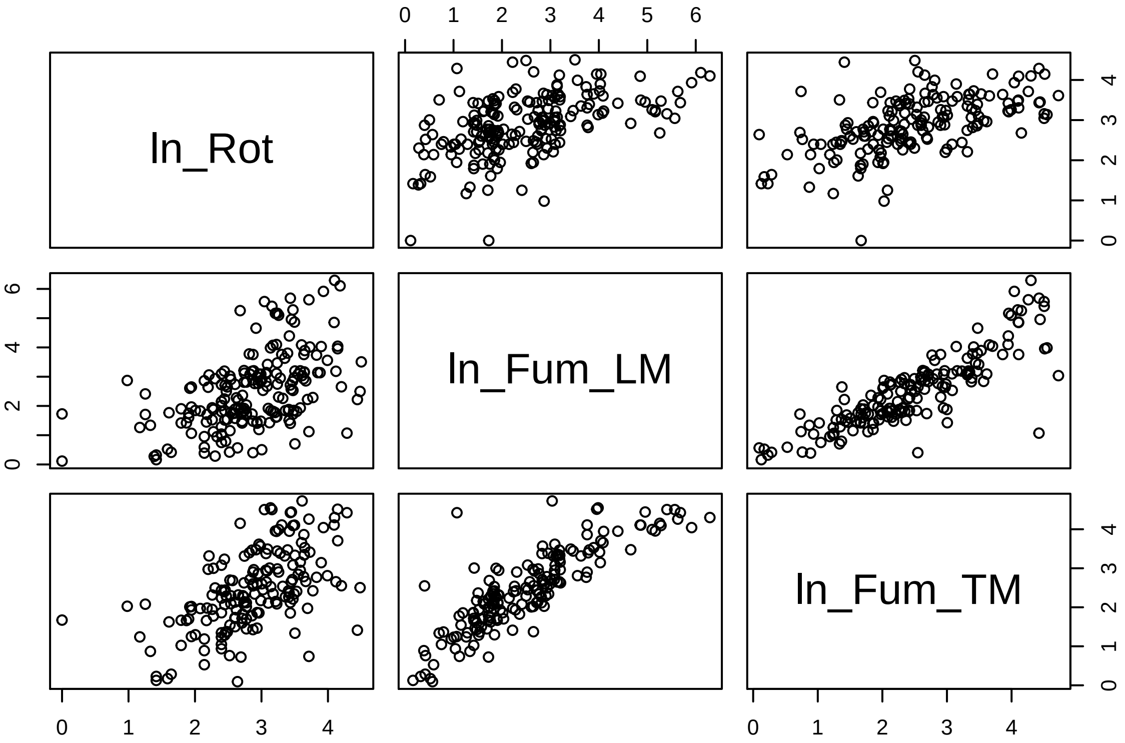
```
dat35 = mutate(dat35,
               ln_Rot = log(1 + AvgRot),
               ln_Fum_LM = log(1 + Fum_LM),
               ln_Fum_TM = log(1 + Fum_TM))
```

Check the correlations among transformed values, across inoculation methods.

```
cor(dat35[,c("ln_Rot", "ln_Fum_LM", "ln_Fum_TM")], use = "pairwise.complete.obs")
```

```
##              ln_Rot ln_Fum_LM ln_Fum_TM
## ln_Rot    1.0000000 0.5159397 0.5791966
## ln_Fum_LM 0.5159397 1.0000000 0.8550472
## ln_Fum_TM 0.5791966 0.8550472 1.0000000
```

```
pairs(dat35[,c("ln_Rot", "ln_Fum_LM", "ln_Fum_TM")])
```



OK! this looks much nicer!

Check the correlations among transformed values separately for each inoculation method.

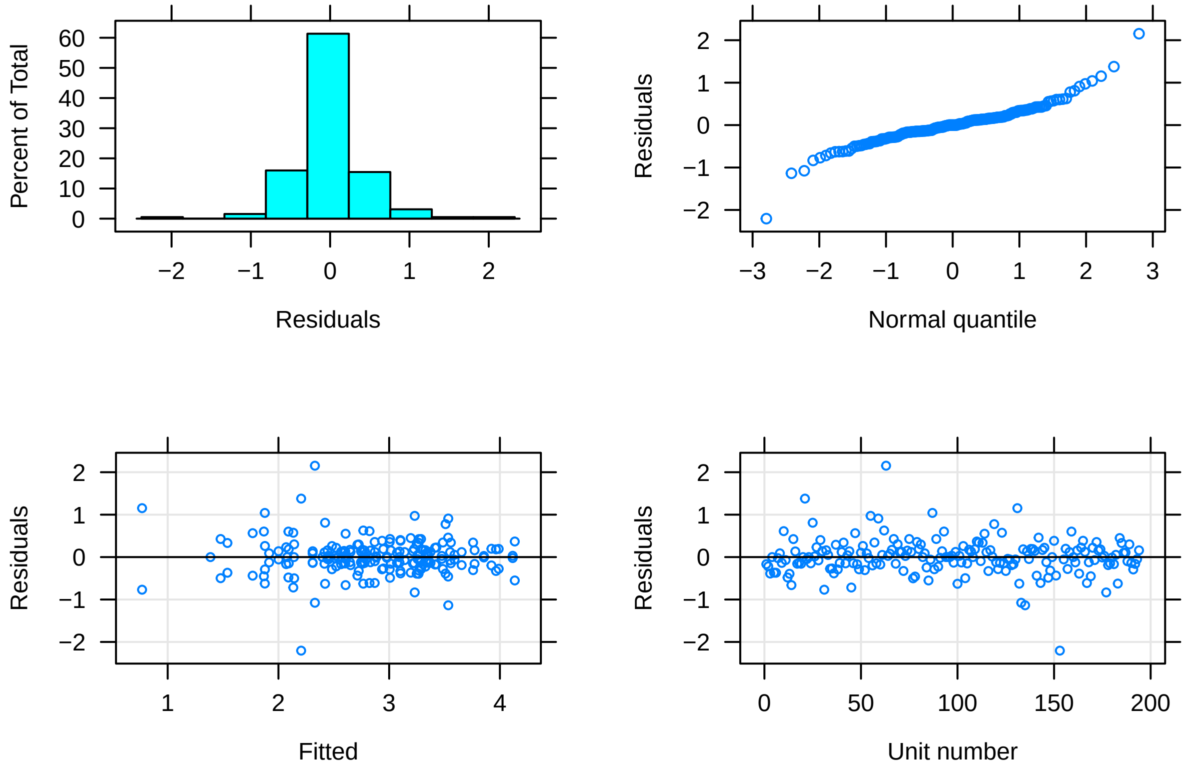
```
dat35.wide = mutate(dat35.wide,
                    ln_Rot_syr = log(1 + Rot_syr),
                    ln_Fum_LM_syr = log(1 + Fum_LM_syr),
                    ln_Fum_TM_syr = log(1 + Fum_TM_syr),
                    ln_Rot_tooth = log(1 + Rot_tooth),
                    ln_Fum_LM_tooth = log(1 + Fum_LM_tooth),
                    ln_Fum_TM_tooth = log(1 + Fum_TM_tooth)
                    )

cor(dat35.wide[,c("ln_Rot_syr", "ln_Fum_LM_syr", "ln_Fum_TM_syr", "ln_Rot_tooth","ln_Fum_LM_tooth", "ln_
```

```
##                 ln_Rot_syr ln_Fum_LM_syr ln_Fum_TM_syr ln_Rot_tooth
## ln_Rot_syr       1.0000000     0.5296012     0.6435863    0.4343257
## ln_Fum_LM_syr    0.5296012     1.0000000     0.8211500    0.4457921
## ln_Fum_TM_syr    0.6435863     0.8211500     1.0000000    0.4664575
## ln_Rot_tooth     0.4343257     0.4457921     0.4664575    1.0000000
## ln_Fum_LM_tooth  0.3663410     0.4620779     0.5185385    0.4501808
## ln_Fum_TM_tooth  0.4400353     0.5027808     0.5182262    0.4633914
##                 ln_Fum_LM_tooth ln_Fum_TM_tooth
## ln_Rot_syr            0.3663410       0.4400353
## ln_Fum_LM_syr         0.4620779       0.5027808
## ln_Fum_TM_syr         0.5185385       0.5182262
## ln_Rot_tooth          0.4501808       0.4633914
## ln_Fum_LM_tooth       1.0000000       0.8759961
## ln_Fum_TM_tooth       0.8759961       1.0000000
```

Analysis of log transformed Ear Rot

```r
Rot.ln = exp35_lmm(dat35, 'ln_Rot')
```

```
## ASReml: Wed May 24 12:51:59 2017
## 
##      LogLik         S2      DF      wall     cpu
##    -762.2771      1.8007    92  12:51:59     0.0 (1 restrained)
##    -758.6596      1.9376    92  12:51:59     0.0 (1 restrained)
##    -757.3526      2.0777    92  12:51:59     0.0 (1 restrained)
##    -757.0869      2.1617    92  12:51:59     0.0 (1 restrained)
##    -757.0412      2.1980    92  12:51:59     0.0 (1 restrained)
##    -757.0294      2.2169    92  12:51:59     0.0
##    -757.0251      2.2310    92  12:51:59     0.0 (1 restrained)
##    -757.0246      2.2339    92  12:51:59     0.0 (1 restrained)
##    -757.0246      2.2340    92  12:51:59     0.0 (1 restrained)
##    -757.0246      2.2341    92  12:51:59     0.0
##    -757.0246      2.2341    92  12:51:59     0.0
##    -757.0246      2.2341    92  12:51:59     0.0
## 
## Finished on: Wed May 24 12:51:59 2017
## 
## LogLikelihood Converged
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep, data = data, 
##     weights = NoEars, na.method.X = "omit")
## 
## $loglik
## [1] -757.0246
## 
## $nedf
## [1] 92
## 
## $sigma
## [1] 1.494678
## 
## $varcomp
##                            gamma    component    std.error z.ratio
## Rep!Rep.var         1.011929e-07 2.260711e-07 3.333237e-08 6.78233
## Block:Rep!Block.var 4.588422e-08 1.025082e-07 1.511400e-08 6.78233
## R!variance          1.000000e+00 2.234061e+00 3.293943e-01 6.78233
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## R!variance            Positive
## 
## attr(,"class")
## [1] "summary.asreml"
## ASReml: Wed May 24 12:51:59 2017
## 
##      LogLik         S2      DF      wall     cpu
##     -24.0371      2.2341    92  12:51:59     0.0
##     -24.0371      2.2341    92  12:51:59     0.0
##     -24.0371      2.2341    92  12:51:59     0.0
##     -24.0371      2.2341    92  12:51:59     0.0
##  Warning: 2 negative weights may have been treated as zero
##
## Finished on: Wed May 24 12:51:59 2017
##
## LogLikelihood Converged
## $Wald
##                         Df denDF    F.inc    F.con Margin           Pr
## (Intercept)              1    92 5490.000 5490.000        8.014155e-84
## Entry                   49    92    3.389    3.367      A 2.553024e-07
## InoculationMethod        1    92   41.800   41.800      A 4.776406e-09
## Entry:InoculationMethod 49    92    0.933    0.933      B 5.985439e-01
##
## $stratumVariances
##           df Variance Block:Rep
## Block:Rep 92 2.234061         1
```

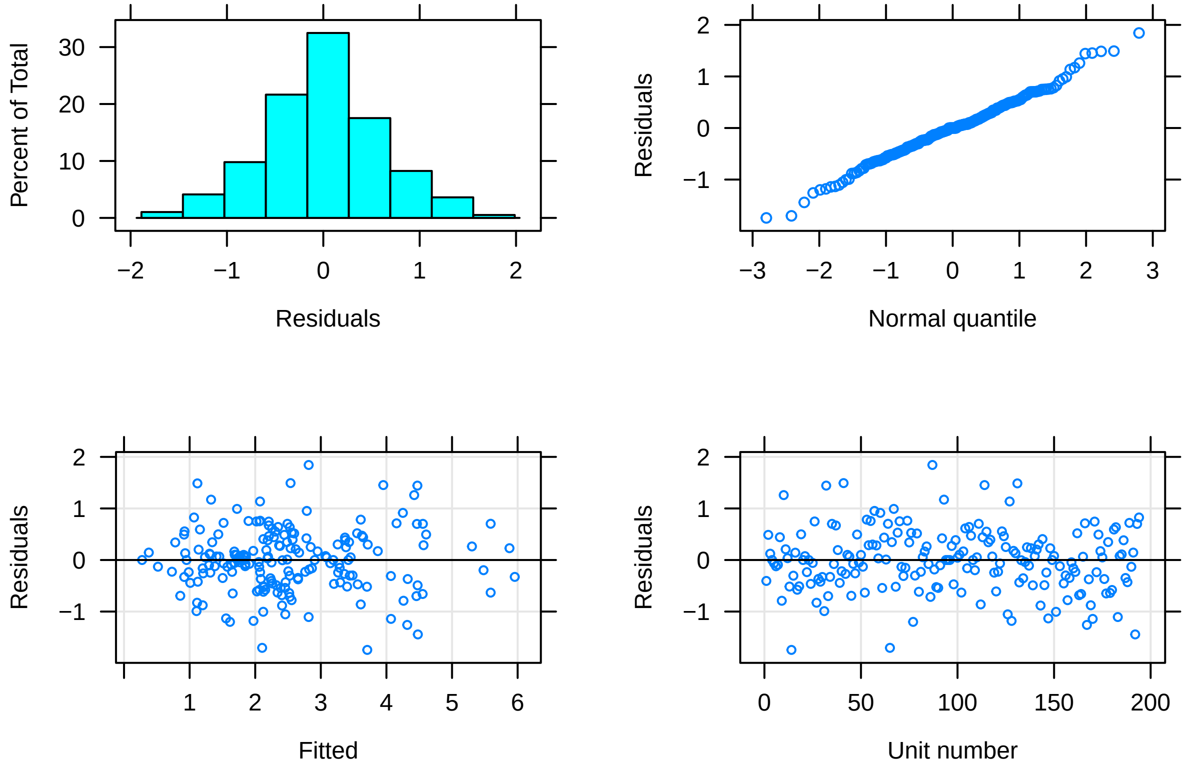


## Analysis of log transformed fumonisin (Laura)

```
Fum_LM.ln = exp35_lmm(dat35, 'ln_Fum_LM')
```

```
## ASReml: Wed May 24 12:51:59 2017
##
##      LogLik         S2      DF      wall      cpu
##    -806.6550      4.7647    92  12:51:59     0.0 (1 restrained)
##    -801.1216      5.5210    92  12:51:59     0.0 (1 restrained)
##    -800.6017      5.5165    92  12:51:59     0.0 (1 restrained)
##    -800.5362      5.5451    92  12:51:59     0.0 (1 restrained)
##    -800.5311      5.5580    92  12:51:59     0.0 (1 restrained)
##    -800.5308      5.5613    92  12:51:59     0.0
##    -800.5308      5.5623    92  12:51:59     0.0
##    -800.5308      5.5623    92  12:51:59     0.0
##
## Finished on: Wed May 24 12:51:59 2017
##
## LogLikelihood Converged
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep, data = data,
##     weights = NoEars, na.method.X = "omit")
##
## $loglik
## [1] -800.5308
##
## $nedf
## [1] 92
##
## $sigma
## [1] 2.358446
##
## $varcomp
##                            gamma    component    std.error   z.ratio
## Rep!Rep.var         1.011929e-07 5.628620e-07 9.142433e-08 6.1565882
## Block:Rep!Block.var 6.237142e-03 3.469265e-02 6.108215e-02 0.5679671
## R!variance          1.000000e+00 5.562268e+00 9.034659e-01 6.1565882
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## ASReml: Wed May 24 12:51:59 2017
##
##      LogLik         S2      DF      wall     cpu
##     -67.9266      5.5623    92  12:51:59     0.0
##     -67.9266      5.5623    92  12:51:59     0.0
##     -67.9266      5.5623    92  12:51:59     0.0
##     -67.9266      5.5623    92  12:51:59     0.0
##  Warning: 2 negative weights may have been treated as zero
##
## Finished on: Wed May 24 12:51:59 2017
##
## LogLikelihood Converged
## $Wald
##                         Df denDF     F.inc     F.con Margin           Pr
## (Intercept)              1   7.0 1122.0000 1122.0000        5.007763e-09
## Entry                   49  63.6    4.5860    4.5940      A 1.190900e-08
## InoculationMethod        1  76.6   42.2700   42.2700      A 7.331321e-09
## Entry:InoculationMethod 49  76.5    0.8859    0.8859      B 6.719532e-01
```

```
##
## $stratumVariances
##                    df Variance Block:Rep R!variance
## Block:Rep   16.19286 7.062484  43.24298          1
## R!variance 75.80714 5.562269   0.00000          1
```

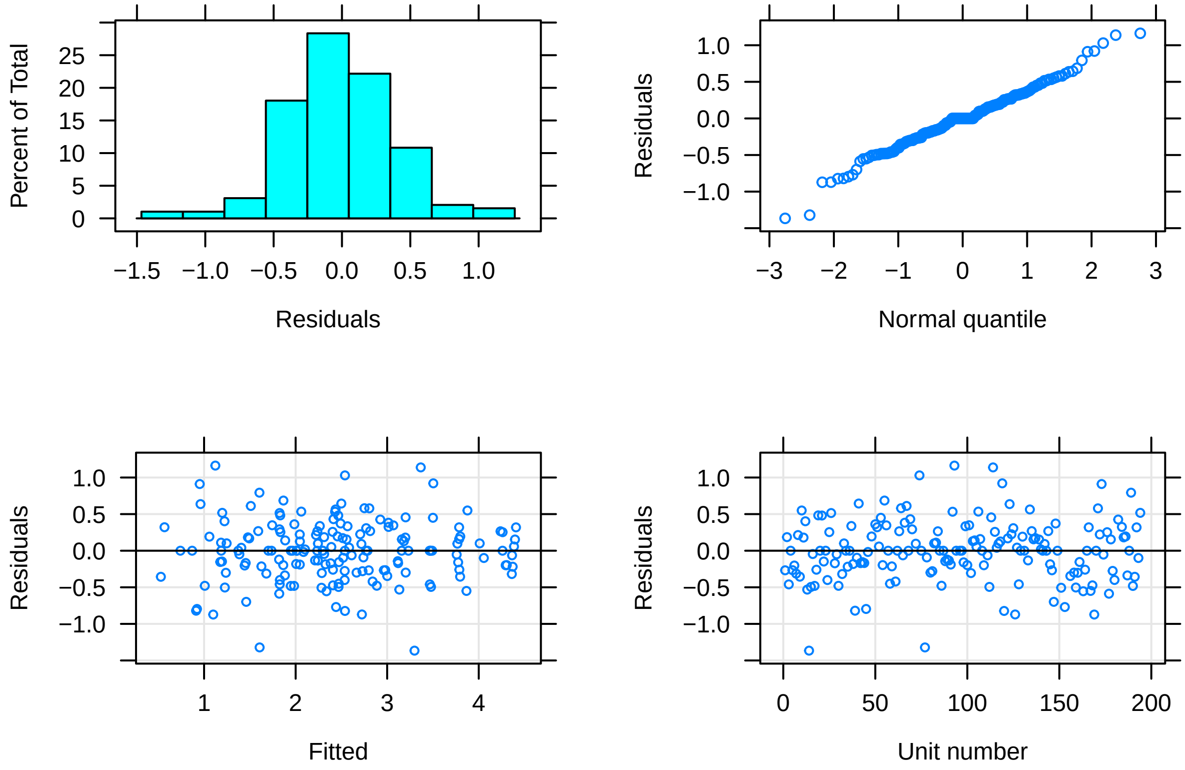


Analysis of log transformed fumonisin (Thiago)

```
Fum_TM.ln = exp35_lmm(dat35, 'ln_Fum_TM')
```

```
## ASReml: Wed May 24 12:51:59 2017
##
##       LogLik         S2      DF      wall      cpu
##    -769.5252       2.2546    74   12:52:00     0.0 (1 restrained)
##    -767.3362       2.3904    74   12:52:00     0.0 (1 restrained)
##    -765.9998       2.6112    74   12:52:00     0.0 (1 restrained)
##    -765.6726       2.7643    74   12:52:00     0.0 (1 restrained)
##    -765.6293       2.8208    74   12:52:00     0.0 (1 restrained)
##    -765.6217       2.8450    74   12:52:00     0.0
##    -765.6199       2.8595    74   12:52:00     0.0
##    -765.6198       2.8635    74   12:52:00     0.0
##    -765.6198       2.8646    74   12:52:00     0.0
##
## Finished on: Wed May 24 12:52:00 2017
##
```

```
## LogLikelihood Converged
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep, data = data,
##     weights = NoEars, na.method.X = "omit")
##
## $loglik
## [1] -765.6198
##
## $nedf
## [1] 74
##
## $sigma
## [1] 1.692523
##
## $varcomp
##                            gamma    component    std.error   z.ratio
## Rep!Rep.var         1.011929e-07 2.898807e-07 5.528174e-08 5.2436964
## Block:Rep!Block.var 3.127525e-03 8.959216e-03 3.914666e-02 0.2288629
## R!variance          1.000000e+00 2.864635e+00 5.463006e-01 5.2436964
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## ASReml: Wed May 24 12:52:00 2017
##
##      LogLik         S2      DF      wall     cpu
##     -28.6638      2.8650    74  12:52:00     0.0
##     -28.6638      2.8650    74  12:52:00     0.0
##     -28.6638      2.8650    74  12:52:00     0.0
##     -28.6638      2.8651    74  12:52:00     0.0
##  Warning: 2 negative weights may have been treated as zero
##
## Finished on: Wed May 24 12:52:00 2017
##
## LogLikelihood Converged
## $Wald
##                         Df denDF    F.inc    F.con Margin           Pr
## (Intercept)              1   4.0 2297.000 2297.000        1.165227e-06
## Entry                   49  40.7    5.241    5.323      A 1.367309e-07
## InoculationMethod        1  57.9   47.920   47.920      A 3.994451e-09
## Entry:InoculationMethod 46  57.0    1.202    1.202      B 2.534898e-01
##
## $stratumVariances
##                 df Variance R!variance <NA>
## R!variance 19.0065 3.128602   29.46962    1
## <NA>       54.9935 2.865072    0.00000    1
```



Log transformed variables have better residuals distributions, so we will use them.

Now compute heritabilities for each combination of inoculation method and trait

Cullis estimator is :

$$h^2 = 1 - \frac{(Avg.\ variance\ of\ line\ comparisons)}{2Vg}$$

```
exp35_h2 = function(data, inoc, trait){
  data.sub = data[data$InoculationMethod == inoc,]
  form = as.formula(paste(trait, "~ 1"))
  assign("form", form, envir = globalenv() )
  mod = asreml(fixed = form, random = ~ Rep + Block:Rep + Entry, weights = NoEars, data = data.sub, na.

  print(summary(mod))

  #estimate heritability with Cullis estimator
  preds = predict(mod, classify = "Entry", data = data)
  avsed = preds$predictions$avsed

  Vcomps = summary(mod)$varcomp
  Vg = Vcomps["Entry!Entry.var", "component"]
  h2 =1 -((avsed**2)/(2*Vg))
  print(paste("Heritability of line means for", trait, "and", inoc, ":", round(h2, 2)))
  return(h2)
}
```

Apply the heritability estimator over each combination of trait and inoculation method

```
h2.results = list()
for (inoc in c("Syringe", "Toothpick")){
  for (trait in c("ln_Rot", "ln_Fum_LM", "ln_Fum_TM")){
    h2.results[[paste(inoc, trait)]] = exp35_h2(dat35, inoc, trait)
  }
}
```

```
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data.sub,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -707.5683
##
## $nedf
## [1] 96
##
## $sigma
## [1] 0.7970235
##
## $varcomp
##                          gamma   component  std.error   z.ratio constraint
## Rep!Rep.var         0.01873810 0.011903313 0.02047503 0.5813575   Positive
## Block:Rep!Block.var 0.01276263 0.008107412 0.01340087 0.6049913   Positive
## Entry!Entry.var     0.31245120 0.198483502 0.05154727 3.8505142   Positive
## R!variance          1.00000000 0.635246403 0.15236642 4.1692020   Positive
##
## attr(,"class")
## [1] "summary.asreml"
## [1] "Heritability of line means for ln_Rot and Syringe : 0.68"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data.sub,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -789.3228
##
## $nedf
## [1] 96
##
## $sigma
## [1] 1.761809
##
## $varcomp
##                          gamma    component    std.error  z.ratio
## Rep!Rep.var         0.00000160 4.966353e-06 1.201332e-06 4.134039
## Block:Rep!Block.var 0.04646217 1.442172e-01 9.196611e-02 1.568156
## Entry!Entry.var     0.31437357 9.758063e-01 2.564231e-01 3.805454
## R!variance          1.00000000 3.103970e+00 7.508325e-01 4.134039
##                      constraint
## Rep!Rep.var            Boundary
## Block:Rep!Block.var    Positive
## Entry!Entry.var        Positive
## R!variance             Positive
##
## attr(,"class")
## [1] "summary.asreml"
## [1] "Heritability of line means for ln_Fum_LM and Syringe : 0.8"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data.sub,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -757.3213
##
## $nedf
## [1] 90
##
## $sigma
## [1] 1.370048
##
## $varcomp
##                             gamma    component    std.error   z.ratio
## Rep!Rep.var         1.011929e-07 1.899423e-07 5.168435e-08 3.6750460
## Block:Rep!Block.var 1.295129e-02 2.430999e-02 3.946681e-02 0.6159605
## Entry!Entry.var     3.024736e-01 5.677527e-01 1.536637e-01 3.6947742
## R!variance          1.000000e+00 1.877032e+00 5.107507e-01 3.6750460
##                      constraint
## Rep!Rep.var            Boundary
## Block:Rep!Block.var    Positive
## Entry!Entry.var        Positive
## R!variance             Positive
##
## attr(,"class")
## [1] "summary.asreml"
## [1] "Heritability of line means for ln_Fum_TM and Syringe : 0.77"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data.sub,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -29.97551
##
## $nedf
## [1] 94
##
## $sigma
## [1] 1.768993
##
## $varcomp
##                           gamma  component  std.error   z.ratio constraint
## Rep!Rep.var         0.005373026 0.01681400 0.04634347 0.3628126   Positive
## Block:Rep!Block.var 0.020406516 0.06385882 0.06406876 0.9967233   Positive
## Entry!Entry.var     0.055229935 0.17283296 0.09982527 1.7313548   Positive
## R!variance          1.000000000 3.12933487 0.73781593 4.2413490   Positive
##
## attr(,"class")
## [1] "summary.asreml"
## [1] "Heritability of line means for ln_Rot and Toothpick : 0.63"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data.sub,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -65.40039
##
## $nedf
## [1] 94
##
## $sigma
## [1] 2.688771
##
## $varcomp
##                            gamma    component    std.error  z.ratio
## Rep!Rep.var         1.011929e-07 7.315732e-07 1.478192e-07 4.949109
## Block:Rep!Block.var 6.797389e-08 4.914167e-07 9.929396e-08 4.949109
## Entry!Entry.var     5.849268e-02 4.228723e-01 2.100233e-01 2.013455
## R!variance          1.000000e+00 7.229491e+00 1.460766e+00 4.949109
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## Entry!Entry.var       Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## [1] "Heritability of line means for ln_Fum_LM and Toothpick : 0.55"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data.sub,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -38.5529
##
## $nedf
## [1] 79
##
## $sigma
## [1] 1.930437
##
## $varcomp
##                            gamma    component    std.error  z.ratio
## Rep!Rep.var         1.011929e-07 3.771039e-07 8.785718e-08 4.292238
## Block:Rep!Block.var 1.011929e-07 3.771039e-07 8.785718e-08 4.292238
## Entry!Entry.var     1.343969e-01 5.008416e-01 1.783618e-01 2.808009
## R!variance          1.000000e+00 3.726585e+00 8.682149e-01 4.292238
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## Entry!Entry.var       Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
## [1] "Heritability of line means for ln_Fum_TM and Toothpick : 0.74"
```

```
for (i in names(h2.results)) {
  print(i)
  print(paste("heritability of line means =", round(h2.results[[i]], 2)))
}
```

```
## [1] "Syringe ln_Rot"
## [1] "heritability of line means = 0.68"
## [1] "Syringe ln_Fum_LM"
## [1] "heritability of line means = 0.8"
## [1] "Syringe ln_Fum_TM"
## [1] "heritability of line means = 0.77"
## [1] "Toothpick ln_Rot"
## [1] "heritability of line means = 0.63"
## [1] "Toothpick ln_Fum_LM"
## [1] "heritability of line means = 0.55"
## [1] "Toothpick ln_Fum_TM"
## [1] "heritability of line means = 0.74"
```

We will also check results for the average values across inoculation methods and assay methods. ln_FUM_avg is the average across inoculation methods and assay methods. NoEars is used to weight the means for the two different inoculation methods and a new NoEars value is computed as the sum of number of ears for the two methods, used for weighting the mixed model analysis below.

```
dat35b = group_by(dat35, Rep, Plot, Entry) %>%
  summarise(ln_Rot_mn = weighted.mean(ln_Rot, NoEars, na.rm = T),
            ln_Fum_LM_mn = weighted.mean(ln_Fum_LM, NoEars, na.rm = T),
            ln_Fum_TM_mn = weighted.mean(ln_Fum_TM, NoEars, na.rm = T),
            NoEars = sum(NoEars)) %>%
  mutate(ln_FUM_avg = rowMeans(cbind(ln_Fum_LM_mn, ln_Fum_TM_mn), na.rm = T))
```

This works nicely to get the means:

```
tbl_df(dat35b)
```

```
## # A tibble: 99 × 8
##       Rep  Plot          Entry ln_Rot_mn ln_Fum_LM_mn ln_Fum_TM_mn NoEars
##    <fctr> <int>         <fctr>     <dbl>        <dbl>        <dbl>  <int>
## 1       1     1          NC526  2.758872     2.336256     2.515165     22
## 2       1     2          NC538  2.981573     1.554352     2.609036     10
## 3       1     3        NCG1511  2.081850     2.263423     2.050700     20
## 4       1     4        NCG1514  2.441401     2.133574     1.901716     20
## 5       1     5        NCG1516  3.329434     4.511017     3.905138     17
## 6       1     6          NC542  2.581848     2.473606     2.793798     15
## 7       1     7          NC530  2.450875     2.467013     2.297335     11
## 8       1     8 P.3737xNC320*3  3.693037     2.221853     1.973325      6
## 9       1     9     14CL1164-1  2.666428     2.161337     1.864081     19
## 10      1    10        NCG1518  2.796329     1.754324     2.220776     16
## # ... with 89 more rows, and 1 more variables: ln_FUM_avg <dbl>
```

But, we lost the incomplete block info, so need to grab that and merge it back in.

```
dat35.blocks = group_by(dat35, Rep, Plot) %>% summarize(Block = unique(Block))
tbl_df(dat35.blocks)
```

```
## # A tibble: 99 × 3
##       Rep  Plot  Block
##    <fctr> <int> <fctr>
## 1       1     1      1
## 2       1     2      1
## 3       1     3      1
## 4       1     4      1
## 5       1     5      1
## 6       1     6      2
## 7       1     7      2
## 8       1     8      2
## 9       1     9      2
## 10      1    10      2
## # ... with 89 more rows
```

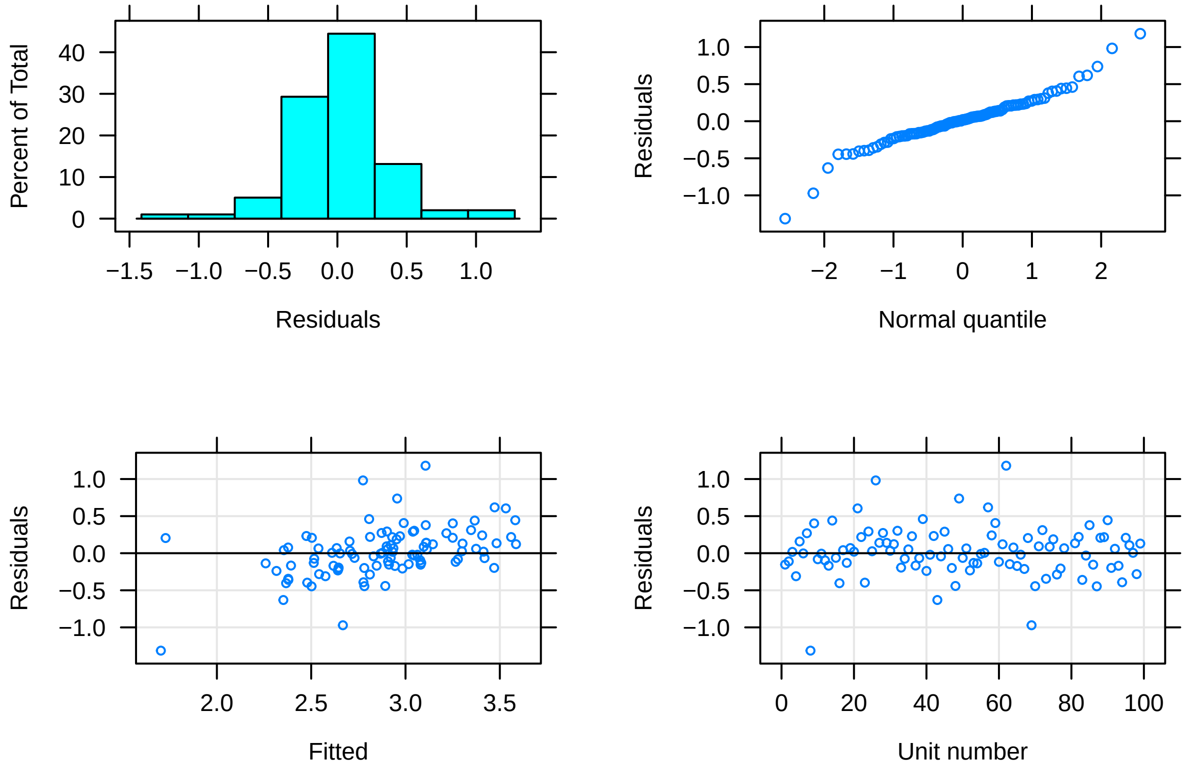
```
dat35b = merge(dat35.blocks, dat35b)
```

One more heritability function, this time to work on the average trait values in dat35b data frame

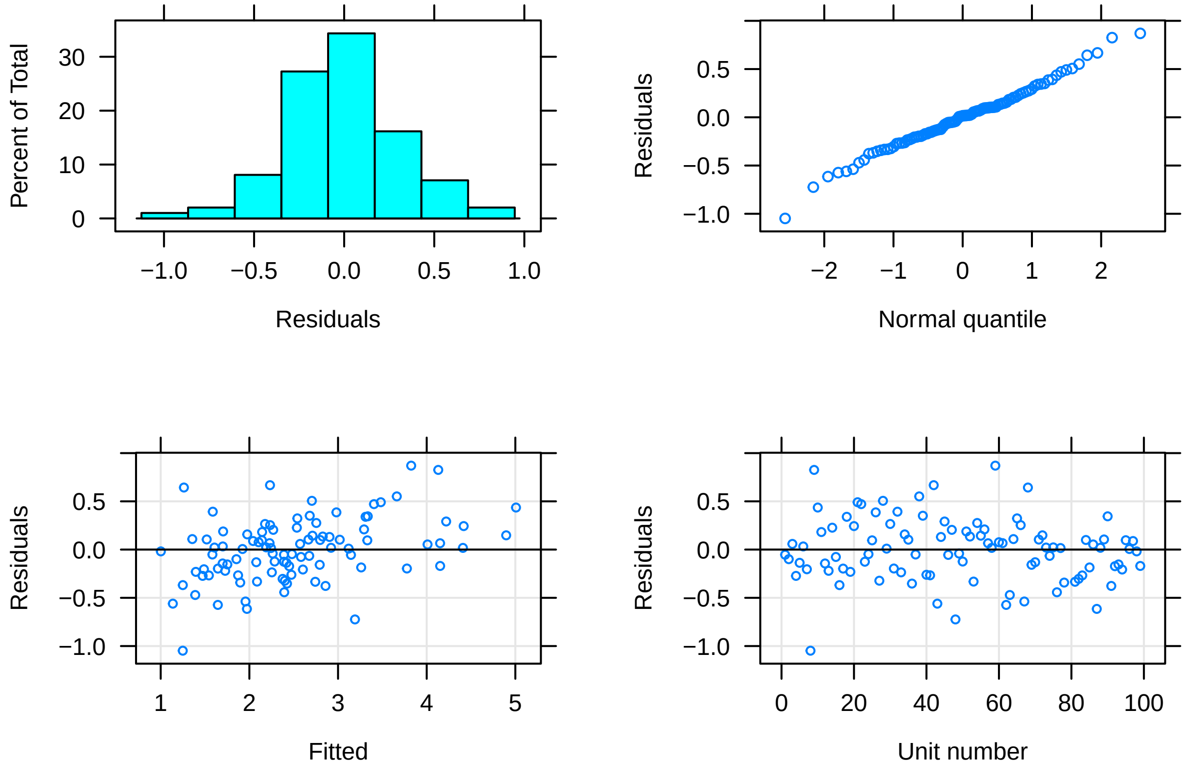
```
exp35b_h2 = function(data, trait){
  form = as.formula(paste(trait, "~ 1"))
  assign("form", form, envir = globalenv() )
  mod = asreml(fixed = form, random = ~ Rep + Block:Rep + Entry, weights = NoEars, data = data, na.metho

  print(summary(mod))
  plot(mod)

  #estimate heritability as the average reliability of line predictions
  preds = predict(mod, classify = "Entry", data = data)
  avsed = preds$predictions$avsed

  Vcomps = summary(mod)$varcomp
  Vg = Vcomps["Entry!Entry.var", "component"]
  h2 =1 -((avsed**2)/(2*Vg))
  print(paste("Heritability of line means for", trait, ":", round(h2, 2)))

  return(h2)
}
```

```
h2.results.b = list()
for (trait in c("ln_Rot_mn", "ln_Fum_LM_mn", "ln_Fum_TM_mn", "ln_FUM_avg")){
  h2.results.b[[trait]] = exp35b_h2(dat35b, trait)
}
```

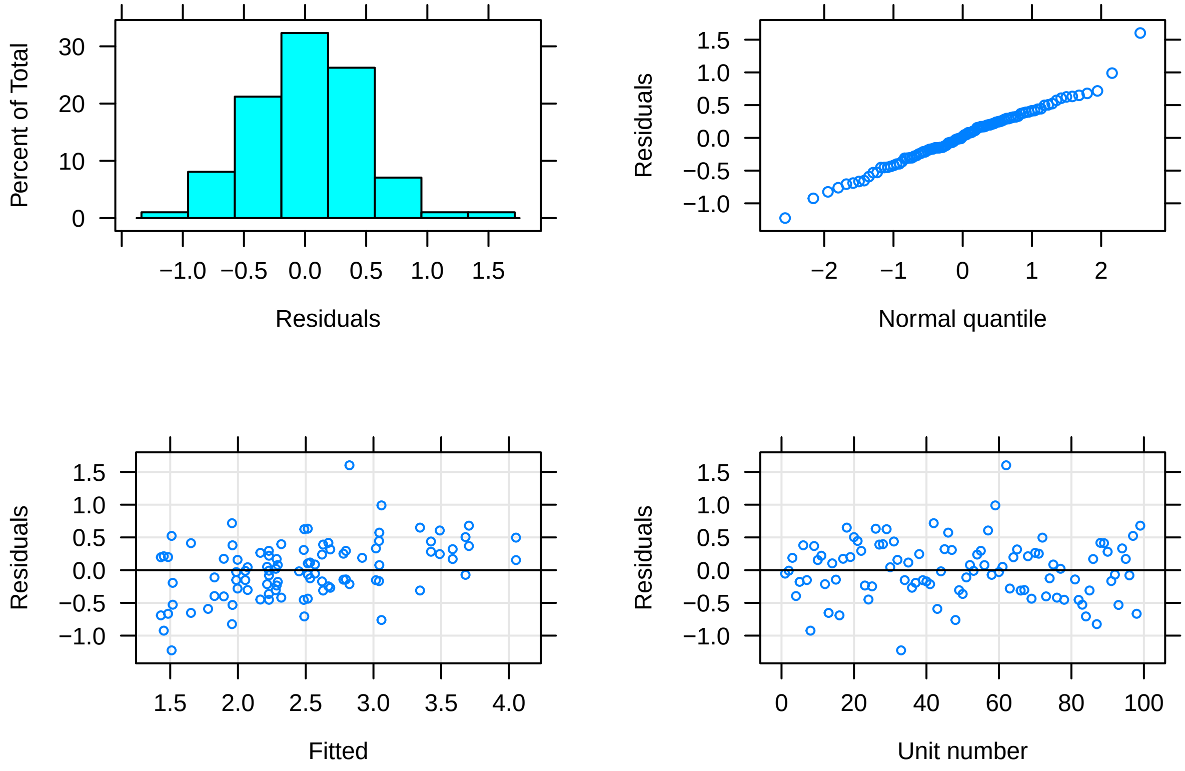
```
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -719.9986
##
## $nedf
## [1] 96
##
## $sigma
## [1] 1.424454
##
## $varcomp
##                            gamma    component    std.error   z.ratio
## Rep!Rep.var         1.011929e-07 2.053275e-07 4.821045e-08 4.2589829
## Block:Rep!Block.var 4.322631e-03 8.770922e-03 1.921128e-02 0.4565506
## Entry!Entry.var     9.105790e-02 1.847629e-01 5.665111e-02 3.2614170
## R!variance          1.000000e+00 2.029070e+00 4.764213e-01 4.2589829
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Positive
## Entry!Entry.var       Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
```



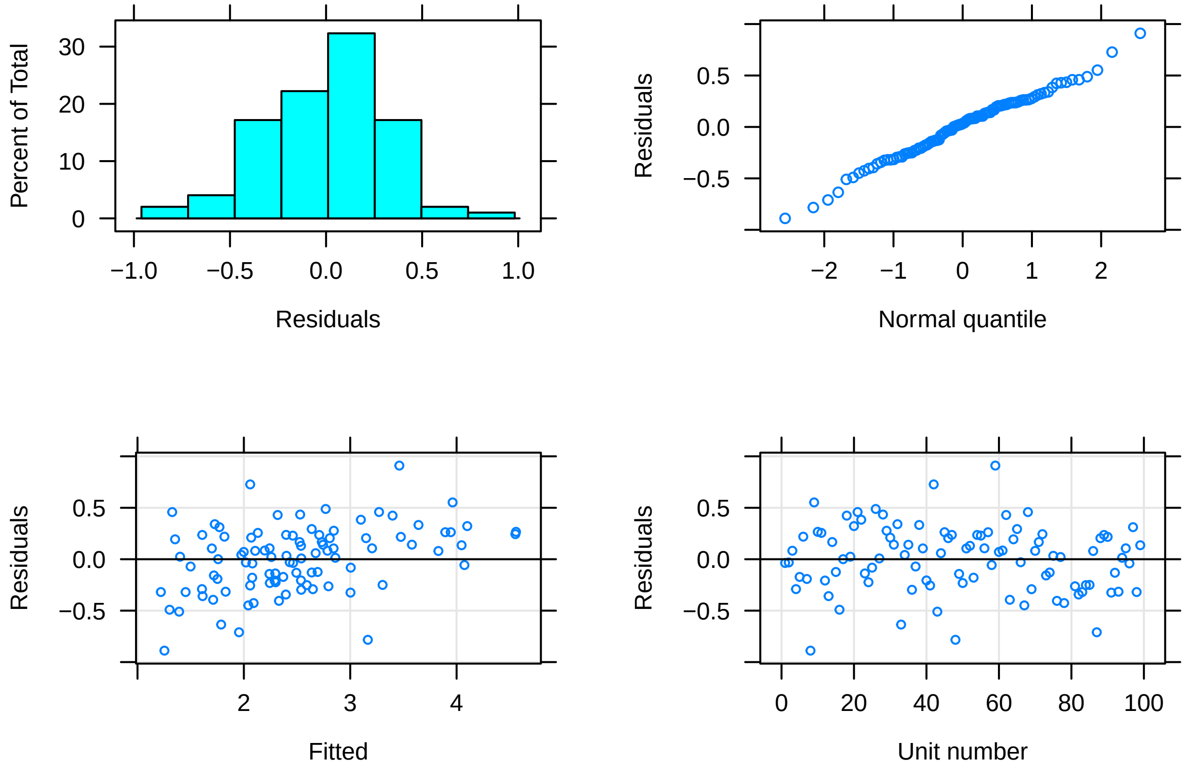
```
## [1] "Heritability of line means for ln_Rot_mn : 0.7"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -765.7682
##
## $nedf
## [1] 96
##
## $sigma
## [1] 1.732906
##
## $varcomp
##                             gamma    component    std.error  z.ratio
## Rep!Rep.var          1.011929e-07 3.038786e-07 7.891531e-08 3.850693
## Block:Rep!Block.var  2.622923e-02 7.876544e-02 5.275421e-02 1.493065
## Entry!Entry.var      2.781771e-01 8.353558e-01 1.988130e-01 4.201717
## R!variance           1.000000e+00 3.002964e+00 7.798503e-01 3.850693
##                      constraint
## Rep!Rep.var            Boundary
## Block:Rep!Block.var    Positive
## Entry!Entry.var        Positive
## R!variance             Positive
##
## attr(,"class")
## [1] "summary.asreml"
```



```
## [1] "Heritability of line means for ln_Fum_LM_mn : 0.86"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -764.9807
##
## $nedf
## [1] 96
##
## $sigma
## [1] 2.007859
##
## $varcomp
##                             gamma    component    std.error  z.ratio
## Rep!Rep.var         1.011929e-07 4.079590e-07 8.403938e-08 4.854379
## Block:Rep!Block.var 1.011929e-07 4.079590e-07 8.403938e-08 4.854379
## Entry!Entry.var     1.304676e-01 5.259800e-01 1.402408e-01 3.750550
## R!variance          1.000000e+00 4.031498e+00 8.304870e-01 4.854379
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Boundary
## Entry!Entry.var       Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
```



```
## [1] "Heritability of line means for ln_Fum_TM_mn : 0.78"
## $call
## asreml(fixed = form, random = ~Rep + Block:Rep + Entry, data = data,
##     weights = NoEars, na.method.X = "omit", maxiter = 25, trace = F)
##
## $loglik
## [1] -755.6201
##
## $nedf
## [1] 96
##
## $sigma
## [1] 1.656117
##
## $varcomp
##                            gamma    component    std.error   z.ratio
## Rep!Rep.var         1.011929e-07 2.775442e-07 7.311204e-08 3.7961493
## Block:Rep!Block.var 1.088224e-02 2.984699e-02 3.617015e-02 0.8251829
## Entry!Entry.var     2.480141e-01 6.802344e-01 1.631802e-01 4.1686088
## R!variance          1.000000e+00 2.742725e+00 7.225017e-01 3.7961493
##                     constraint
## Rep!Rep.var           Boundary
## Block:Rep!Block.var   Positive
## Entry!Entry.var       Positive
## R!variance            Positive
##
## attr(,"class")
## [1] "summary.asreml"
```



```
## [1] "Heritability of line means for ln_FUM_avg : 0.85"
```

Get the summary of heritabilities

```
for (i in names(h2.results.b)) {
  print(i)
  print(paste("heritability of line means =", round(h2.results.b[[i]], 2)))
}
```

```
## [1] "ln_Rot_mn"
## [1] "heritability of line means = 0.7"
## [1] "ln_Fum_LM_mn"
## [1] "heritability of line means = 0.86"
## [1] "ln_Fum_TM_mn"
## [1] "heritability of line means = 0.78"
## [1] "ln_FUM_avg"
## [1] "heritability of line means = 0.85"
```

Now get the line mean values, output a file of means.

```
exp35_means = function(data, trait){
  form = as.formula(paste(trait, "~ 1 + Entry"))
  assign("form", form, envir = globalenv() )
  mod = asreml(fixed = form, random = ~ Rep + Block:Rep, weights = NoEars, data = data, na.method.X='om

  preds = predict(mod, classify = "Entry", data = data)
  BLUEs = preds$predictions$pvals[c("Entry", "predicted.value")]
  #get the avg. standard error of line mean comparisons
  avgLSD = data.frame(Entry = "Avg LSD", predicted.value = 2*preds$predictions$avsed)

  BLUEs = rbind(BLUEs, avgLSD)
  return(BLUEs)
}
```

```
exp35.means = list()
for (trait in c("ln_Rot_mn", "ln_Fum_LM_mn", "ln_Fum_TM_mn", "ln_FUM_avg")){
  meanies = exp35_means(dat35b, trait)
  names(meanies)[2] = trait
  exp35.means[[trait]] = meanies
}
```

```
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik        S2      DF      wall     cpu
##    -731.4274      1.4272    47   12:52:03     0.0 (1 restrained)
##    -726.5719      1.7374    47   12:52:03     0.0 (2 restrained)
##    -725.0031      2.0197    47   12:52:03     0.0 (1 restrained)
##    -725.0031      2.0197    47   12:52:03     0.0 (2 restrained)
##    -724.8563      2.0517    47   12:52:03     0.0 (1 restrained)
##    -724.8563      2.0517    47   12:52:03     0.0 (2 restrained)
##    -724.8464      2.0538    47   12:52:03     0.0 (1 restrained)
##    -724.8458      2.0540    47   12:52:03     0.0
##    -724.8458      2.0540    47   12:52:03     0.0
##    -724.8458      2.0540    47   12:52:03     0.0
##
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik        S2      DF      wall     cpu
##    -727.8686      2.0540    47   12:52:03     0.0
##    -727.8686      2.0540    47   12:52:03     0.0
##    -727.8686      2.0540    47   12:52:03     0.0
##    -727.8686      2.0540    47   12:52:03     0.0
##
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik        S2      DF      wall     cpu
##    -742.0730      2.2446    47   12:52:03     0.0 (1 restrained)
##    -740.5877      2.4058    47   12:52:03     0.0 (1 restrained)
##    -739.9573      2.6468    47   12:52:03     0.0 (1 restrained)
##    -739.8740      2.7725    47   12:52:03     0.0 (1 restrained)
##    -739.8694      2.8006    47   12:52:03     0.0 (1 restrained)
##    -739.8691      2.8082    47   12:52:03     0.0
##    -739.8691      2.8109    47   12:52:03     0.0
##    -739.8691      2.8110    47   12:52:03     0.0
##
```

```
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik         S2      DF      wall     cpu
##    -747.5260      2.8110    47  12:52:03     0.0
##    -747.5260      2.8110    47  12:52:03     0.0
##    -747.5260      2.8110    47  12:52:03     0.0
##    -747.5260      2.8110    47  12:52:03     0.0
##
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik         S2      DF      wall     cpu
##    -744.3880      2.4774    47  12:52:03     0.0 (1 restrained)
##    -742.4259      2.7073    47  12:52:03     0.0 (2 restrained)
##    -740.6730      3.8009    47  12:52:03     0.0 (1 restrained)
##    -740.5738      3.8451    47  12:52:03     0.0 (1 restrained)
##    -740.5578      3.9094    47  12:52:03     0.0 (1 restrained)
##    -740.5532      3.9535    47  12:52:03     0.0
##    -740.5511      3.9923    47  12:52:03     0.0 (1 restrained)
##    -740.5506      4.0062    47  12:52:03     0.0 (1 restrained)
##    -740.5506      4.0071    47  12:52:03     0.0 (1 restrained)
##    -740.5506      4.0071    47  12:52:03     0.0
##    -740.5506      4.0071    47  12:52:03     0.0
##    -740.5506      4.0071    47  12:52:03     0.0
##
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik         S2      DF      wall     cpu
##    -748.2122      4.0071    47  12:52:03     0.0
##    -748.2122      4.0071    47  12:52:03     0.0
##    -748.2122      4.0071    47  12:52:03     0.0
##    -748.2122      4.0071    47  12:52:03     0.0
##
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik         S2      DF      wall     cpu
##    -739.0950      1.7926    47  12:52:03     0.0 (1 restrained)
##    -737.7104      1.9059    47  12:52:03     0.0 (1 restrained)
##    -737.0049      2.1020    47  12:52:03     0.0 (1 restrained)
##    -736.8337      2.2566    47  12:52:03     0.0 (1 restrained)
##    -736.8137      2.3131    47  12:52:03     0.0 (1 restrained)
##    -736.8106      2.3361    47  12:52:03     0.0
##    -736.8101      2.3490    47  12:52:03     0.0
##    -736.8100      2.3519    47  12:52:03     0.0
##
## Finished on: Wed May 24 12:52:03 2017
##
## LogLikelihood Converged
## ASReml: Wed May 24 12:52:03 2017
##
##      LogLik         S2      DF      wall     cpu
##    -735.1423      2.3526    47  12:52:04     0.0
##    -735.1423      2.3527    47  12:52:04     0.0
##    -735.1423      2.3527    47  12:52:04     0.0
##    -735.1423      2.3528    47  12:52:04     0.0
##
## Finished on: Wed May 24 12:52:04 2017
##
## LogLikelihood Converged
```

```
exp35.means.df = exp35.means[[1]]
for (i in 2:4) {
  exp35.means.df = merge(exp35.means.df, exp35.means[[i]])
}
tbl_df(exp35.means.df)
```

```
## # A tibble: 51 × 5
##            Entry ln_Rot_mn ln_Fum_LM_mn ln_Fum_TM_mn ln_FUM_avg
##           <fctr>     <dbl>        <dbl>        <dbl>      <dbl>
## 1     14CL1161-1 2.9495302    2.8240592     3.104910  2.9187816
## 2     14CL1162-1 2.2040377    1.6819515     1.764802  1.7615990
## 3     14CL1164-1 2.6855982    2.5540391     2.175267  2.3858013
## 4        Avg LSD 0.8050197    1.0450454     1.124407  0.9444587
## 5            B73 3.4683521    5.3508929     4.407724  4.9663378
## 6  CML373xNC320* 1.7213013    0.4406991     1.187357  0.7016125
## 7         FR1064 3.5427385    4.4825683     4.228335  4.3760784
## 8          GE440 1.1609377    1.0602682     1.096181  1.1115309
## 9      GEMS-0002 2.8942621    1.7298827     1.491752  1.6598473
## 10     GEMS-0224 2.9257302    2.1616733     2.143302  2.2304423
## # ... with 41 more rows
```

Back-transform the data

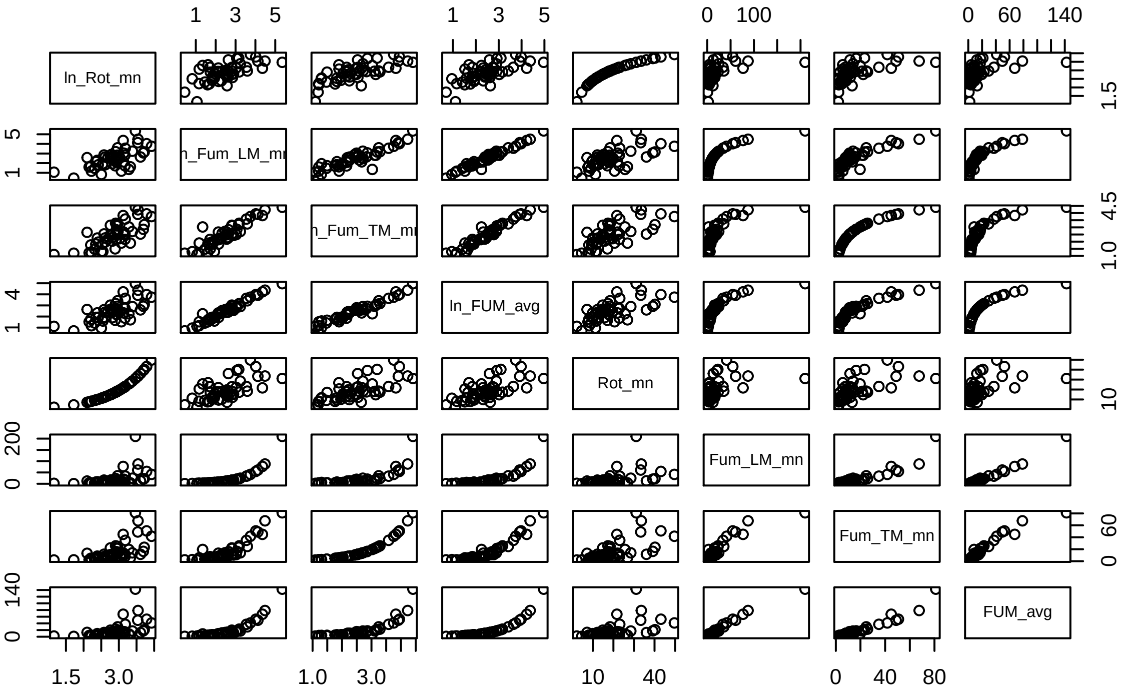
```
exp35.means.df = mutate(exp35.means.df,
                        Rot_mn = exp(ln_Rot_mn) - 1,
                        Fum_LM_mn = exp(ln_Fum_LM_mn) - 1,
                        Fum_TM_mn = exp(ln_Fum_TM_mn) - 1,
                        FUM_avg = exp(ln_FUM_avg) - 1)

#force Avg LSD to bottom of data frame
exp35.means.out = exp35.means.df
exp35.means.out$pos = ifelse(exp35.means.out$Entry == "Avg LSD", 2, 1)
exp35.means.out = exp35.means.out[order(exp35.means.out$pos, exp35.means.out$Entry),]
exp35.means.out = exp35.means.out[,c("Entry", "ln_Rot_mn", "Rot_mn", "ln_FUM_avg", "FUM_avg")]
names(exp35.means.out) = c("Entry", "ln_EarRot", "EarRot", "ln_FUM", "FUM")
write.csv(exp35.means.out, file = "Exp35_2015_means.csv", row.names = F)
```

```
cor(filter(exp35.means.df, Entry != "Avg LSD") %>% select(-Entry) )
```

```
##              ln_Rot_mn ln_Fum_LM_mn ln_Fum_TM_mn ln_FUM_avg    Rot_mn
## ln_Rot_mn    1.0000000    0.6481609    0.7290909  0.6894336 0.9364181
## ln_Fum_LM_mn 0.6481609    1.0000000    0.9072356  0.9791543 0.6312201
## ln_Fum_TM_mn 0.7290909    0.9072356    1.0000000  0.9685311 0.7138054
## ln_FUM_avg   0.6894336    0.9791543    0.9685311  1.0000000 0.6709035
## Rot_mn       0.9364181    0.6312201    0.7138054  0.6709035 1.0000000
## Fum_LM_mn    0.4294904    0.7889350    0.7129305  0.7769317 0.4574098
## Fum_TM_mn    0.5920125    0.8631612    0.8999571  0.9006248 0.6392133
## FUM_avg      0.4870725    0.8359531    0.7936381  0.8419948 0.5202446
##              Fum_LM_mn Fum_TM_mn   FUM_avg
## ln_Rot_mn    0.4294904 0.5920125 0.4870725
## ln_Fum_LM_mn 0.7889350 0.8631612 0.8359531
## ln_Fum_TM_mn 0.7129305 0.8999571 0.7936381
## ln_FUM_avg   0.7769317 0.9006248 0.8419948
## Rot_mn       0.4574098 0.6392133 0.5202446
## Fum_LM_mn    1.0000000 0.8973917 0.9856105
## Fum_TM_mn    0.8973917 1.0000000 0.9533619
## FUM_avg      0.9856105 0.9533619 1.0000000
```

```
pairs(filter(exp35.means.df, Entry != "Avg LSD") %>% select(-Entry), main = "Line mean averages\nover i
```

Line mean averages
over inoculation methods and assays


ln_Rot_mn
n_Fum_LM_mi
n_Fum_TM_mi
ln_FUM_avg
Rot_mn
Fum_LM_mn
Fum_TM_mn
FUM_avg