Using `asremlPlus`, in conjunction with `asreml`, to do a linear mixed model analysis of a wheat experiment using hypothesis tests

Chris Brien

10 June, 2025

This vignette shows how to use `asremlPlus` (Brien, 2025), in conjunction with `asreml` (Butler et al., 2020), to employ hypothesis tests to select the terms to be included in a mixed model for an experiment that involves spatial variation. It also illustrates diagnostic checking and prediction production and presentation for this experiment. Here, `asremlPlus` and `asreml` are packages for the `R` Statistical Computing environment (R Core Team, 2025).

It is divided into the following main sections:

1. Set up the maximal model for this experiment
2. Perform a series of hypothesis tests to select a linear mixed model for the data
3. Diagnostic checking using residual plots and variofaces
4. Prediction production and presentation

1. Set up the maximal model for this experiment

```
library(knitr)
opts_chunk$set("tidy" = FALSE, comment = NA)
suppressMessages(library(asreml, quietly=TRUE))
```

```
## Offline License checked out Tue Jun 10 18:03:07 2025
```

```
packageVersion("asreml")
```

```
## [1] '4.2.0.370'
```

```
suppressMessages(library(asremlPlus))
packageVersion("asremlPlus")
```

```
## [1] '4.4.49'
```

```
suppressMessages(library(qqplotr, quietly=TRUE))
options(width = 100)
```

Get data available in asremlPlus

The data are from a 1976 spring wheat experiment and are taken from Gilmour et al. (1995). An analysis is presented in the `asreml` manual by Butler et al. (2020, Section 7.6), although they suggest that it is a barley experiment.

```
data(Wheat.dat)
```

Fit the maximal model

In the following a model is fitted that has the terms that would be included for a balanced lattice. In addition, a term WithinColPairs has been included to allow for extraneous variation arising between pairs of adjacent lanes. Also, separable ar1 residual autocorrelation has been included. This model represents the maximal anticipated model,

```
current.asr <- asreml(yield ~ WithinColPairs + Variety,
                      random = ~ Rep/(Row + Column) + units,
                      residual = ~ ar1(Row):ar1(Column),
                      maxit = 30, data=Wheat.dat)
```

```
ASReml Version 4.2 10/06/2025 18:03:08
          LogLik        Sigma2     DF     wall
 1     -724.1213      23034.14    124   18:03:08
 2     -717.4149      9206.931    124   18:03:08  (  2 restrained)
 3     -694.8752      26492.99    124   18:03:08  (  2 restrained)
 4     -694.1600      33101.80    124   18:03:08  (  1 restrained)
 5     -692.0020      36912.26    124   18:03:08  (  1 restrained)
 6     -691.7892      46701.51    124   18:03:08  (  2 restrained)
 7     -691.8336      46208.51    124   18:03:08  (  1 restrained)
 8     -691.7749      47698.26    124   18:03:08
 9     -691.7711      47041.85    124   18:03:08

Warning in asreml(yield ~ WithinColPairs + Variety, random = ~Rep/(Row + : Some components changed
by more than 1% on the last iteration
```

The warning from `asreml` is probably due to a bound term.

Initialize a testing sequence by loading the current fit into an asrtests object

A label and the information criteria based on the full likelihood (Verbyla, 2019) are included in the `test.summary` stored in the `asrtests` object.

```
  current.asrt <- as.asrtests(current.asr, NULL, NULL,
                              label = "Maximal model", IClikelihood = "full")
```

```
Warning in infoCriteria.asreml(asreml.obj, IClikelihood = ic.lik, bound.exclusions = bound.exclusions):
 Rep

Warning in asreml(fixed = yield ~ WithinColPairs + Variety, random = ~Rep/(Row + : Log-likelihood
not converged
```

Check for and remove any boundary terms

```
current.asrt <- rmboundary(current.asrt, IClikelihood = "full")
```

```
Warning in infoCriteria.asreml(asreml.obj, IClikelihood = ic.lik): The following bound terms were discou
 Rep

ASReml Version 4.2 10/06/2025 18:03:08
          LogLik        Sigma2     DF     wall
 1     -691.7710      47071.42    124   18:03:08

Warning in asreml(fixed = yield ~ WithinColPairs + Variety, random = ~Rep/(Row + : Log-likelihood
not converged
```

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

```
                          component    std.error    z.ratio bound %ch
Rep:Row                4.293282e+03 3.199458e+03  1.3418779     P 0.0
Rep:Column             1.575689e+02 1.480357e+03  0.1064398     P 0.7
units                  5.742689e+03 1.652457e+03  3.4752438     P 0.0
Row:Column!R           4.706787e+04 2.515832e+04  1.8708669     P 0.0
Row:Column!Row!cor     7.920301e-01 1.014691e-01  7.8056280     U 0.0
Row:Column!Column!cor 8.799559e-01 7.370402e-02 11.9390486     U 0.0
```

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

```
####  Sequence of model investigations for yield

(If a row has NA for p but not denDF, DF and denDF relate to fixed and variance parameter numbers)

          terms DF denDF  p      AIC     BIC         action
1 Maximal model 26     6 NA 1646.129 1742.47 Starting model
2           Rep  1    NA NA 1646.129 1742.47       Boundary
```

Rep has been removed because it has been constrained to zero. Following the recommendation of Littel et al. (2006, p. 150), the bound on all variance components is set to unconstrained (U) using `setvariances.asreml` so as to avoid bias in the estimate of the residual variance. Alternatively, one could move Rep to the fixed model.

Unbind Rep, Row and Column components and reload into an asrtests object

```
current.asr <- setvarianceterms(current.asr$call,
                                terms = c("Rep", "Rep:Row", "Rep:Column"),
                                bounds = "U")
```

```
ASReml Version 4.2 10/06/2025 18:03:09
          LogLik        Sigma2     DF     wall
 1     -724.1213      23034.14    124   18:03:09
 2     -717.4149      9206.931    124   18:03:09  (  2 restrained)
 3     -694.8752      26492.99    124   18:03:09  (  2 restrained)
 4     -693.9744      33129.65    124   18:03:09  (  1 restrained)
 5     -692.8856      39662.12    124   18:03:09
 6     -691.4276      53103.83    124   18:03:09
 7     -691.2387      48092.17    124   18:03:09
 8     -691.1808      47278.94    124   18:03:09
 9     -691.1710      46850.98    124   18:03:09
10     -691.1700      46690.46    124   18:03:09

Warning in asreml(fixed = yield ~ WithinColPairs + Variety, random = ~Rep/(Row + : Some components
changed by more than 1% on the last iteration

WARN [2025-06-10 18:03:09] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ WithinColPairs + Variety, random = ~Rep/(Row + : Some components
changed by more than 1% on the last iteration

WARN [2025-06-10 18:03:09] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ WithinColPairs + Variety, random = ~Rep/(Row + : Some components
changed by more than 1% on the last iteration
```

```r
current.asrt <- as.asrtests(current.asr, wald.tab = NULL, test.summary = current.asrt$test.summary,
                            IClikelihood = "full", label = "Max model & Unbound components")
current.asrt <- rmboundary(current.asrt)
summary(current.asrt$asreml.obj)$varcomp
```

```
                         component    std.error   z.ratio bound %ch
Rep                   -2462.3785858 1.191435e+03 -2.066734     U 0.2
Rep:Row                5012.4021415 3.396848e+03  1.475604     U 0.1
Rep:Column              920.5936391 1.704008e+03  0.540252     U 1.1
units                  5964.9099377 1.608792e+03  3.707695     P 0.1
Row:Column!R          46690.4620387 2.731906e+04  1.709080     P 0.0
Row:Column!Row!cor        0.8152180 9.988929e-02  8.161216     U 0.1
Row:Column!Column!cor     0.8857252 7.487875e-02 11.828793     U 0.0
```

```r
print(current.asrt, which = "testsummary")
```

```
####  Sequence of model investigations for yield

(If a row has NA for p but not denDF, DF and denDF relate to fixed and variance parameter numbers)

                          terms DF denDF  p      AIC      BIC         action
1                 Maximal model 26     6 NA 1646.129 1742.470 Starting model
2                           Rep  1    NA NA 1646.129 1742.470       Boundary
3 Max model & Unbound components 26     7 NA 1647.200 1746.551 Starting model
```

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

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


Wald tests for fixed effects.
Response: yield

               Df denDF   F.inc     Pr
(Intercept)     1   1.7 153.400 0.0115
WithinColPairs  1  15.6   2.543 0.1308
Variety        24  76.1  10.110 0.0000
```

Now the Rep component estimate is negative.

The `test.summary` output has been extended, by supplying the previous `test.summary` to `as.asrtests`, to show that there is a new starting model. The pseudo-anova table shows that Varieties are highly significant ($p < 0.001$)

2. Perform a series of hypothesis tests to select a linear mixed model for the data

The hypothesis tests in this section are Wald tests for fixed terms, with denominator degrees of freedom calculated using the Kenward-Rogers adjustment (Kenward and Rogers (1997), and Restricted Maximum Likelihood Ratio Tests (REMLRT) for random terms.

Check the term for within Column pairs (a post hoc factor)

The information criteria based on the full likelihood (Verbyla, 2019) is also included in the `test.summary` stored in the `asrtests` object.

```
current.asrt <- testranfix(current.asrt, term = "WithinColPairs",
                           drop.fix.ns=TRUE, IClikelihood = "full")
```

```
WARN [2025-06-10 18:03:10] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep/(Row + Column) + : Some components changed
by more than 1% on the last iteration

WARN [2025-06-10 18:03:10] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep/(Row + Column) + : Some components changed
by more than 1% on the last iteration
```

```
print(current.asrt)
```

```
####  Summary of the fitted variance parameters for yield

                          component    std.error    z.ratio bound %ch
Rep                   -2392.1616314 1.199592e+03 -1.9941460     U 0.4
Rep:Row                5033.2850607 3.408523e+03  1.4766764     U 0.2
Rep:Column              760.1498938 1.617038e+03  0.4700879     U 2.5
units                  5929.0518909 1.609478e+03  3.6838361     P 0.0
Row:Column!R          45940.6913910 2.634982e+04  1.7434920     P 0.0
Row:Column!Row!cor        0.8101561 9.995026e-02  8.1055925     U 0.1
Row:Column!Column!cor     0.8846454 7.504265e-02 11.7885681     U 0.0


####  Pseudo-anova table for fixed terms


Wald tests for fixed effects.
Response: yield

            Df denDF  F.inc     Pr
(Intercept)  1   1.7 159.00 0.0112
Variety     24  76.8  10.27 0.0000


####  Sequence of model investigations for yield

(If a row has NA for p but not denDF, DF and denDF relate to fixed and variance parameter numbers)

                           terms DF denDF      p      AIC      BIC         action
1                  Maximal model 26   6.0     NA 1646.129 1742.470 Starting model
2                            Rep  1    NA     NA 1646.129 1742.470       Boundary
3 Max model & Unbound components 26   7.0     NA 1647.200 1746.551 Starting model
4                 WithinColPairs  1  15.6 0.1308 1645.326 1741.666        Dropped
```

It is clear in the call to `testranfix` that the model is being changed by dropping the `withinColPairs` term, which could also be achieved using `update.asreml`. However, an `asremlPlus` model-changing function operates on an `asrtests` object, that includes an `asreml` object, and, except for `changeTerms.asrtests`, results in an `asrtests` object that may contain the changed model or the supplied model depending on the results of hypothesis tests or comparisons of information criteria. In addition, the result of the test or comparison will be added to a `test.summary` data.frame stored in the new `asrtests` object and, if the model was changed, the `wald.tab` in the new `asrtests` object will have been updated for the new model.

In this case, as can be seen from the summary of `current.asrt` after the call, the p-value for the `withinColPairs` was greater than 0.05 and so now the model stored in `current.asrt` does not include `withinColPAirs`. The `wald.tab` has been updated for the new model.

Test the nugget term

The nugget term represents non-spatial variance, such as random plot and measurement error. It is fitted using the `asreml` reserved word `units`.

```
current.asrt <- testranfix(current.asrt, "units", positive=TRUE, IClikelihood = "full")
```

```
WARN [2025-06-10 18:03:10] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep + Rep:Row + Rep:Column, : Some components
changed by more than 1% on the last iteration

WARN [2025-06-10 18:03:10] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep + Rep:Row + Rep:Column, : Some components
changed by more than 1% on the last iteration
```

Test Row autocorrelation

We begin testing the autocorrelation by dropping the Row autocorrelation. Because of messages about the instability of the fit, `iterate.asrtests` is used to execute extra iterations of the fitting process.

```
current.asrt <- testresidual(current.asrt, "~ Row:ar1(Column)",
                             label="Row autocorrelation",
                             simpler=TRUE, IClikelihood = "full")
```

```
WARN [2025-06-10 18:03:11] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep/(Row + Column) + : Some components changed
by more than 1% on the last iteration

WARN [2025-06-10 18:03:11] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep/(Row + Column) + : Some components changed
by more than 1% on the last iteration
```

```
current.asrt <- iterate(current.asrt)
```

```
WARN [2025-06-10 18:03:12] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep/(Row + Column) + : Some components changed
by more than 1% on the last iteration

WARN [2025-06-10 18:03:12] Some components changed by more than 1% on the last iteration

Warning in asreml(fixed = yield ~ Variety, random = ~Rep/(Row + Column) + : Some components changed
by more than 1% on the last iteration
```

Test Column autocorrelation (depends on whether Row autocorrelation retained)

The function `getTestPvalue` is used to get the p-value for the Row autocorrelation test. If it is significant then the Column autocorrelation is tested by by dropping the Column autocorrelation, while retaining the Row autocorrelation. Otherwise the model with just Row autocorrelation, whose fit is returned via `current.asrt` after the test, is compared to one with no autocorrelation.

```
(p <- getTestPvalue(current.asrt, label = "Row autocorrelation"))
```

```
[1] 4.675632e-06
```

```
{ if (p <= 0.05)
  current.asrt <- testresidual(current.asrt, "~ ar1(Row):Column",
                               label="Col autocorrelation",
                               simpler=TRUE, IClikelihood = "full")
  else
    current.asrt <- testresidual(current.asrt, "~ Row:Column",
                                 label="Col autocorrelation",
                                 simpler=TRUE, IClikelihood = "full")
}
```

```
Warning in DFdiff(bound.h1, bound.h0, DF = DF, bound.exclusions = bound.exclusions): There were a total
  The following bound terms occur in only one of the models compared and so were discounted:
   Row:Column!Row!cor
```

Output the results

```
print(current.asrt)
```

```
####  Summary of the fitted variance parameters for yield

                         component    std.error    z.ratio bound %ch
Rep                  -2392.1616314 1.199592e+03 -1.9941460     U 0.4
Rep:Row               5033.2850607 3.408523e+03  1.4766764     U 0.2
Rep:Column             760.1498938 1.617038e+03  0.4700879     U 2.5
units                 5929.0518909 1.609478e+03  3.6838361     P 0.0
Row:Column!R         45940.6913910 2.634982e+04  1.7434920     P 0.0
Row:Column!Row!cor       0.8101561 9.995026e-02  8.1055925     U 0.1
Row:Column!Column!cor    0.8846454 7.504265e-02 11.7885681     U 0.0


####  Pseudo-anova table for fixed terms


Wald tests for fixed effects.
Response: yield

            Df denDF  F.inc     Pr
(Intercept)  1   1.7 159.00 0.0112
Variety     24  76.8  10.27 0.0000


####  Sequence of model investigations for yield

(If a row has NA for p but not denDF, DF and denDF relate to fixed and variance parameter numbers)
```

```
                           terms DF denDF      p      AIC      BIC         action
1                  Maximal model 26   6.0     NA 1646.129 1742.470 Starting model
2                            Rep  1    NA     NA 1646.129 1742.470       Boundary
3 Max model & Unbound components 26   7.0     NA 1647.200 1746.551 Starting model
4                  WithinColPairs  1  15.6 0.1308 1645.326 1741.666        Dropped
5                           units  1    NA 0.0006 1645.326 1741.666       Retained
6             Row autocorrelation  1    NA 0.0000 1645.326 1741.666      Unswapped
7             Col autocorrelation  2    NA 0.0000 1645.326 1741.666      Unswapped
```

```
printFormulae(current.asrt$asreml.obj)
```

```
#### Formulae from asreml object

fixed: yield ~ Variety
random: ~ Rep/(Row + Column) + units
residual: ~ ar1(Row):ar1(Column)
```

```
print(R2adj(current.asrt$asreml.obj, include.which.random = ~ .))
```

```
ASReml Version 4.2 10/06/2025 18:03:13
          LogLik        Sigma2     DF     wall
 1     -694.6149      45873.83    125   18:03:13
 2     -694.6149      45868.25    125   18:03:13
[1] 44.62078
attr(,"fixed")
~.
<environment: 0x000002156a995418>
attr(,"random")
~.
```

The `test.summary` shows is that the model with Row and without Column autocorrelation failed to converge. The `asreml.obj` in `current.asrt` contains the model selected by the selection process, which has been printed using `printFormulae.asrtests`. It is clear that no changes were made to the variance terms. The adjusted R^2 value shows that the fixed and random terms in the fitted model account for 45% of the total variation in the yield.

3. Diagnostic checking using residual plots and variofaces

Get current fitted asreml object and update to include standardized residuals

```
current.asr <- current.asrt$asreml.obj
current.asr <- update(current.asr, aom=TRUE)
```

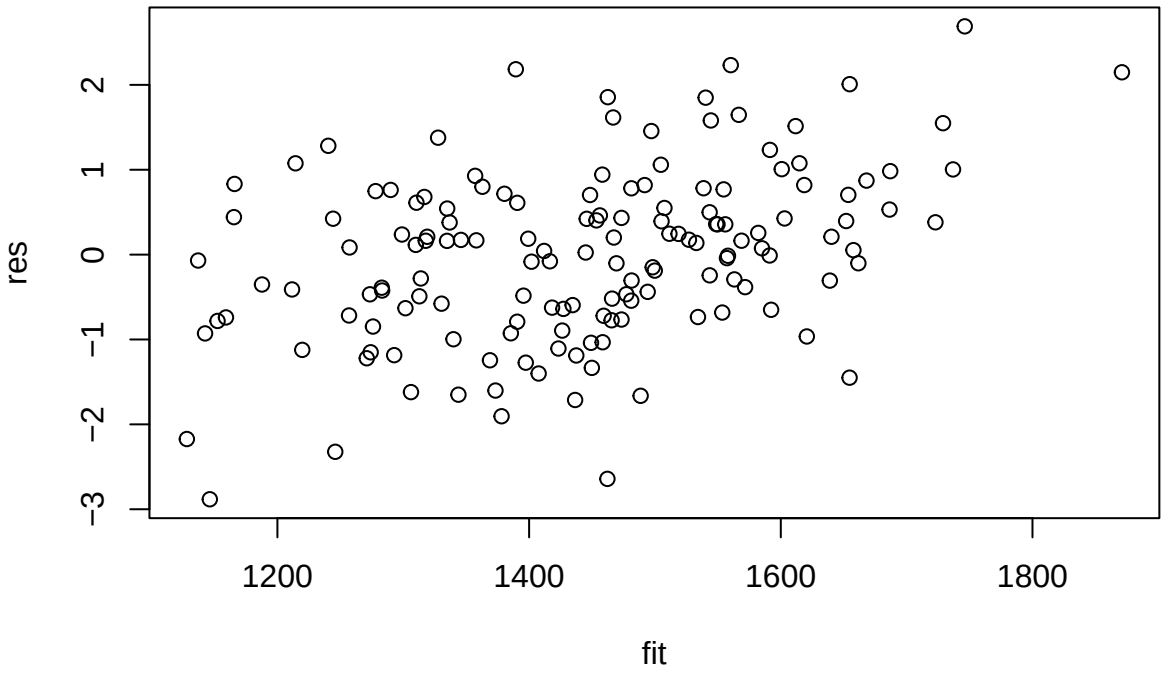
```
ASReml Version 4.2 10/06/2025 18:03:13
          LogLik        Sigma2     DF     wall
 1     -694.6149      45873.83    125   18:03:13
 2     -694.6149      45868.25    125   18:03:13
 3     -694.6149      45854.78    125   18:03:13
```

```
Wheat.dat$res <- residuals(current.asr, type = "stdCond")
Wheat.dat$fit <- fitted(current.asr)
```

Do diagnostic checking

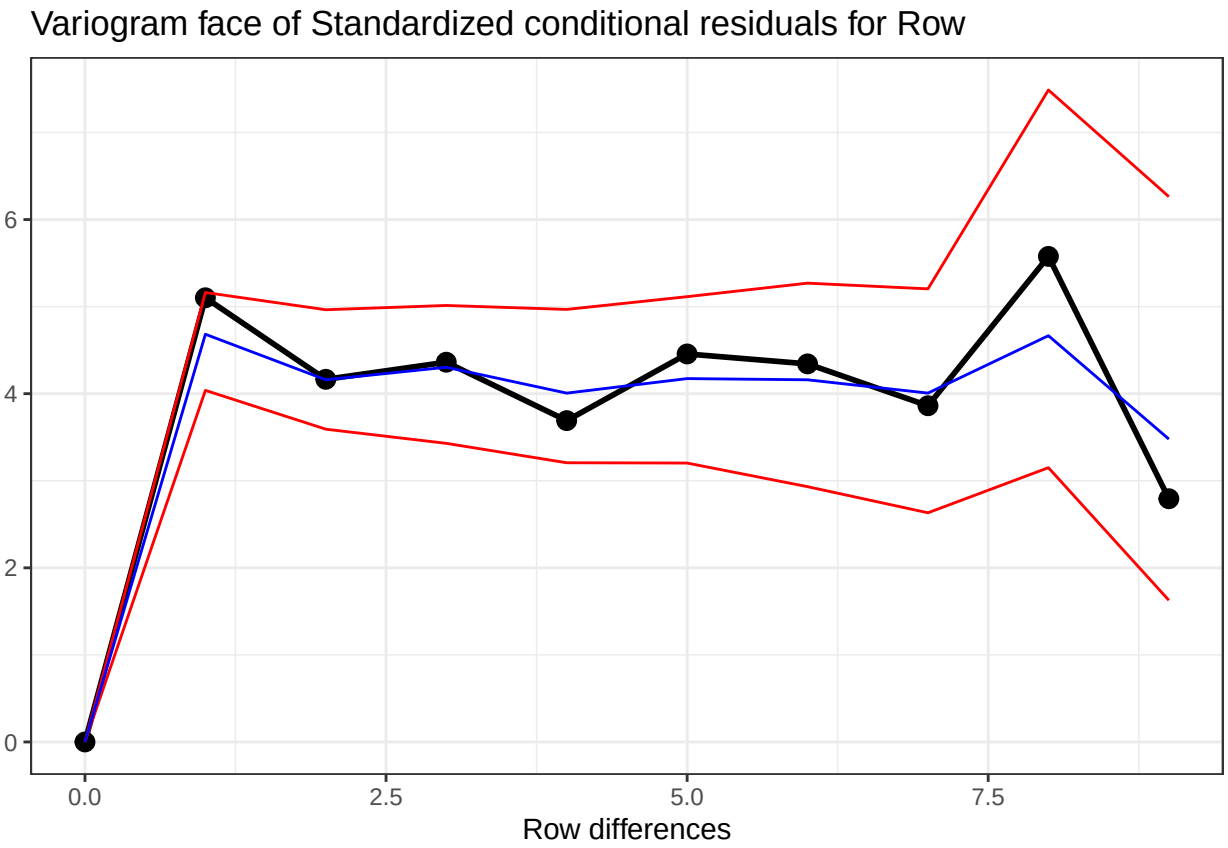
Do residuals-versus-fitted values plot

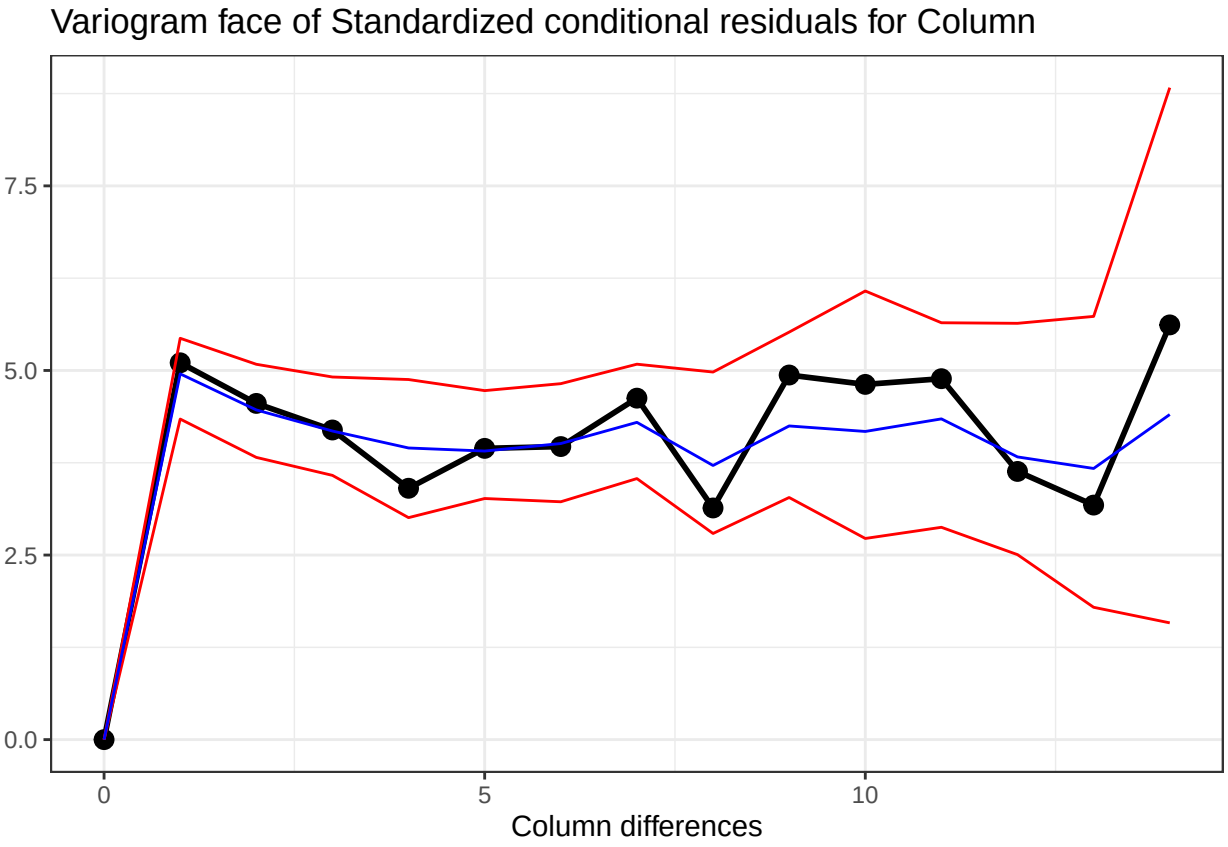
```
with(Wheat.dat, plot(fit, res))
```



Plot variofaces

```
variofaces(current.asr, V=NULL, units="addtores",
           maxiter=50, update = FALSE,
           ncores = parallel::detectCores())
```





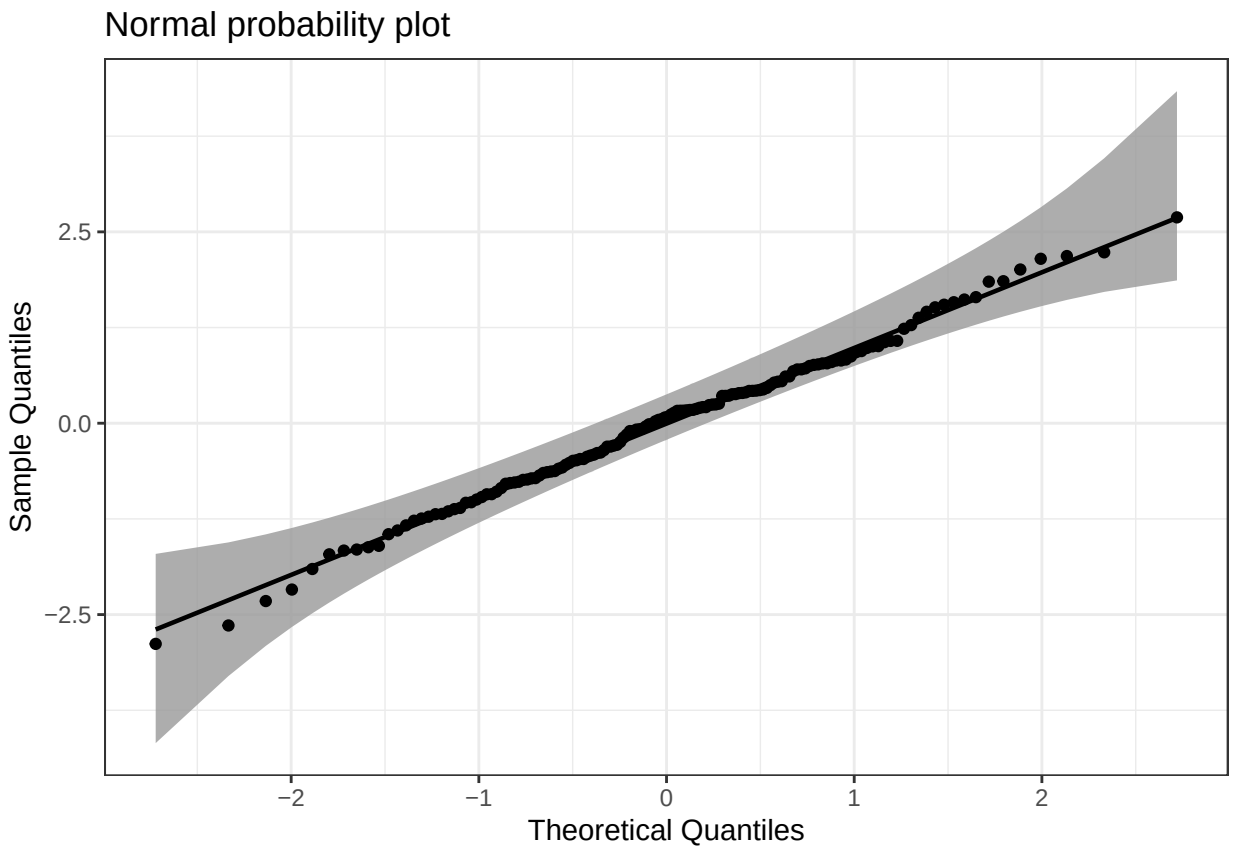
The variofaces are the lag 1 plots of the sample semivariogram with simulated confidence envelopes (Stefanova

et al., 2009).

Plot normal quantile plot

The plot is obtained using the `ggplot` function with extensions available from the `qqplotr` package (Almeida, A., Loy, A. and Hofmann, H., 2023).

```
suppressWarnings(
  ggplot(data = Wheat.dat, mapping = aes(sample = res)) +
    stat_qq_band(bandType = "ts") + stat_qq_line() + stat_qq_point() +
    labs(x = "Theoretical Quantiles", y = "Sample Quantiles",
         title = "Normal probability plot") +
    theme(plot.title = element_text(size = 12, face = "bold")) + theme_bw())
```



4. Prediction production and presentation

Get Variety predictions and all pairwise prediction differences and p-values

```
Var.diffs <- predictPlus(classify = "Variety",
                         asreml.obj=current.asr,
                         error.intervals="halfLeast",
                         wald.tab=current.asrt$wald.tab,
```

```
                          sortFactor = "Variety",
                          tables = "predictions")
```

```
#### Predictions for yield from Variety


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: Rep,Row,Column,units
- Variety is included in this prediction
- (Intercept) is included in this prediction
- units is ignored in this prediction

   Variety predicted.value standard.error upper.halfLeastSignificant.limit
1       10        1168.989       120.4776                          1228.315
2        1        1242.750       119.8112                          1302.076
3        9        1257.137       119.9716                          1316.463
4       16        1285.719       119.9407                          1345.045
5       14        1293.527       119.9234                          1352.853
6       23        1313.653       120.2937                          1372.979
7       11        1322.159       120.1971                          1381.485
8        7        1374.446       120.2415                          1433.773
9        3        1394.070       120.4040                          1453.396
10       4        1410.980       120.1063                          1470.306
11      12        1444.557       120.6042                          1503.883
12       8        1453.396       120.5948                          1512.722
13      15        1458.383       120.4354                          1517.709
14       5        1473.782       120.4462                          1533.108
15      17        1487.828       120.2904                          1547.154
16       6        1498.294       120.1196                          1557.621
17      21        1517.121       120.2270                          1576.447
18       2        1520.466       119.6330                          1579.792
19      24        1533.768       120.3003                          1593.095
20      18        1541.148       120.3671                          1600.474
21      25        1575.795       120.5149                          1635.121
22      22        1610.482       120.3289                          1669.808
23      13        1610.761       120.4582                          1670.088
24      20        1627.971       120.2336                          1687.297
25      19        1652.992       120.3443                          1712.318
   lower.halfLeastSignificant.limit est.status
1                          1109.662  Estimable
2                          1183.424  Estimable
3                          1197.811  Estimable
4                          1226.392  Estimable
5                          1234.200  Estimable
6                          1254.327  Estimable
7                          1262.832  Estimable
8                          1315.120  Estimable
```

```
9                          1334.743  Estimable
10                         1351.654  Estimable
11                         1385.231  Estimable
12                         1394.070  Estimable
13                         1399.057  Estimable
14                         1414.456  Estimable
15                         1428.501  Estimable
16                         1438.968  Estimable
17                         1457.795  Estimable
18                         1461.140  Estimable
19                         1474.442  Estimable
20                         1481.822  Estimable
21                         1516.468  Estimable
22                         1551.156  Estimable
23                         1551.435  Estimable
24                         1568.645  Estimable
25                         1593.666  Estimable


LSD values

minimum LSD =  114.0129

mean LSD =  118.6523

maximum LSD =  123.3577

(sed range / mean sed =  0.0788 )
```

We have set `error.intervals` to `halfLeast` so that the limits for so that the limits for each prediction $\pm$ (0.5 LSD) are calculated. When these are plotted overlapping error bars indicate predictions that are not significant, while those that do not overlap are significantly different (Snee, 1981).

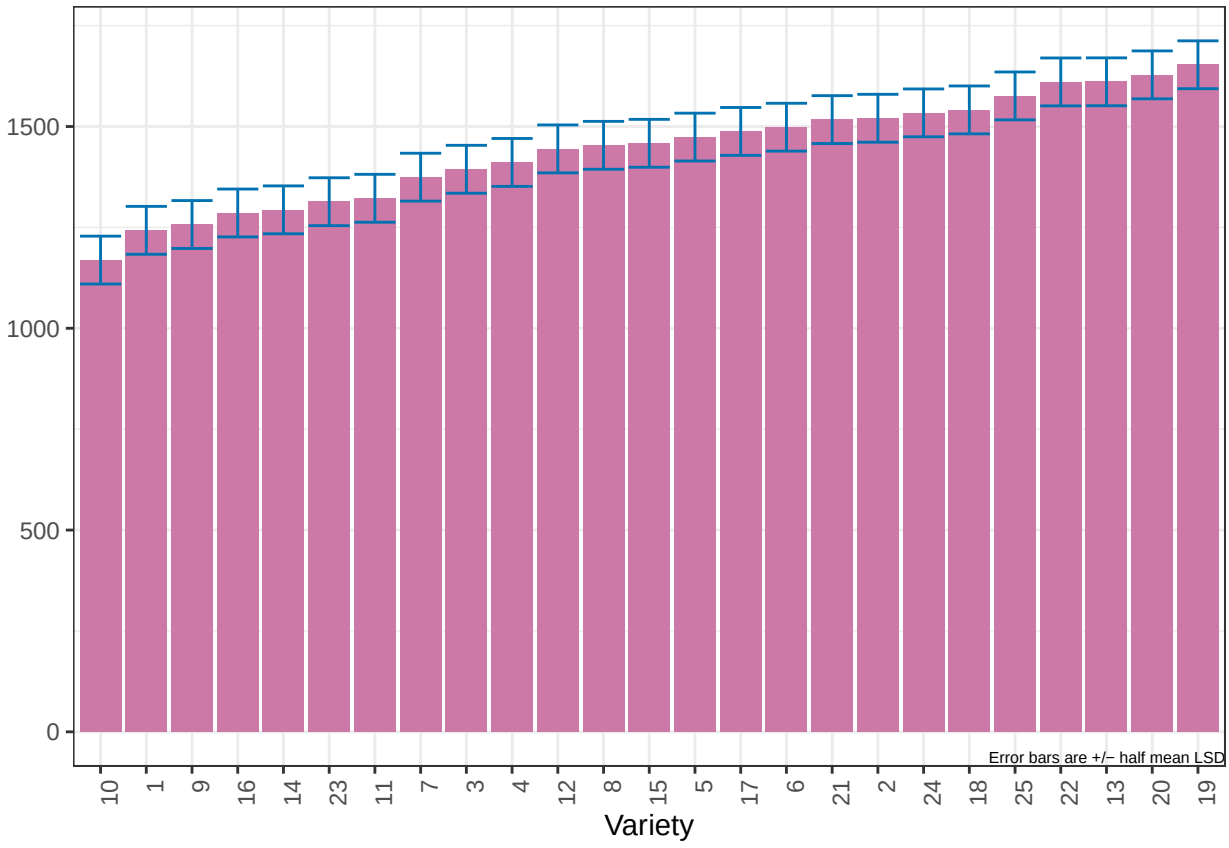
Also set was `sortFactor`, so that the results would be ordered for the values of the predictions for Variety.

The function `predictPlus` returns an `alldiffs` object, a list consisting of the following components:

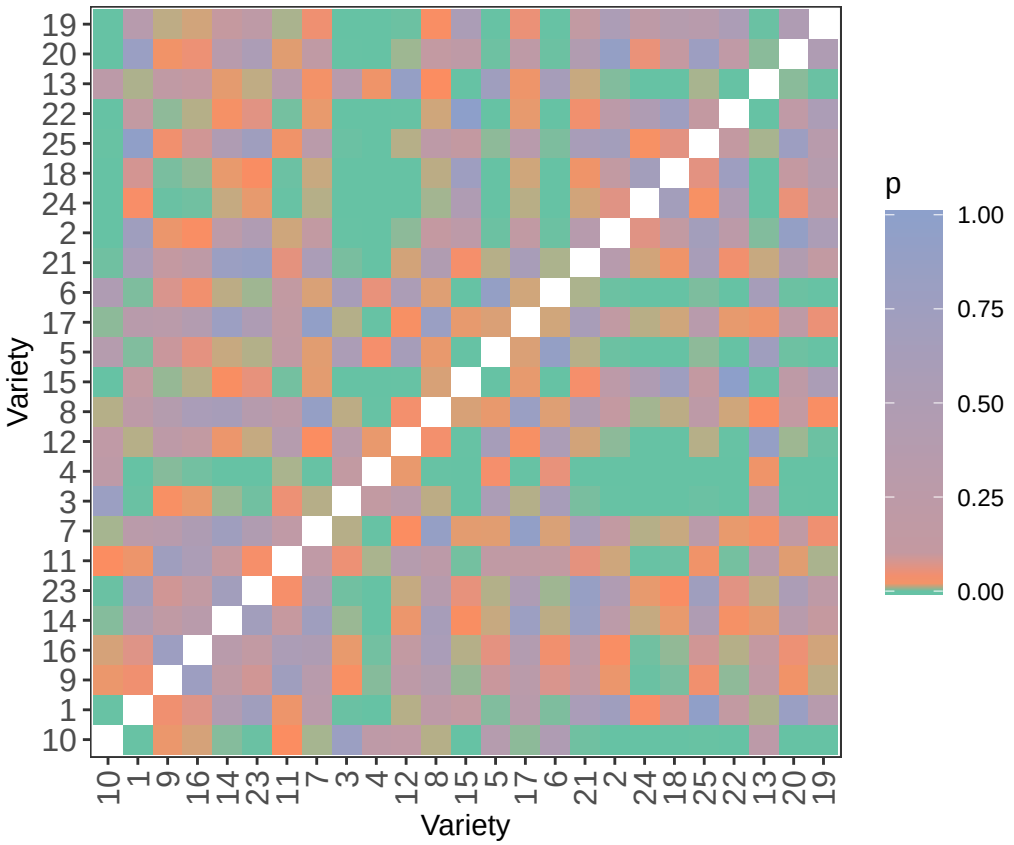
- `predictions`: the predictions, their standard errors and error intervals;
- `vcov`: the variance matrix of the predictions;
- `differences`: all pairwise differences between the predictions,
- `p.differences`: p-values for all pairwise differences between the predictions;
- `sed`: the standard errors of all pairwise differences between the predictions;
- `LSD`: the mean, minimum and maximum LSDs.

Plot the Variety predictions, with halfLSD intervals, and the p-values

```
plotPredictions(Var.diffs$predictions,
                classify = "Variety", y = "predicted.value",
                error.intervals = "half")
```




```
plotPvalues(Var.diffs)
```



References

Almeida, A., Loy, A. and Hofmann, H. (2023) `qqplotr`: *Quantile-Quantile plot extensions for 'ggplot2'*, Version 0.0.6. https://cran.r-project.org/package=qqplotr/ or https://github.com/aloy/qqplotr/.

Brien, C. J. (2025) `asremlPlus`: *Augments ASReml-R in fitting mixed models and packages generally in exploring prediction differences*. Version 4.4.49. https://cran.r-project.org/package=asremlPlus/ or http://chris.brien.name/rpackages/.

Butler, D. G., Cullis, B. R., Gilmour, A. R., Gogel, B. J. and Thompson, R. (2023). *ASReml-R Reference Manual Version 4.2.* VSN International Ltd, https://asreml.kb.vsni.co.uk/.

Gilmour, A. R., Thompson, R., & Cullis, B. R. (1995). Average Information REML: An Efficient Algorithm for Variance Parameter Estimation in Linear Mixed Models. *Biometrics*, **51**, 1440–1450.

Kenward, M. G., & Roger, J. H. (1997). Small sample inference for fixed effects from restricted maximum likelihood. *Biometrics*, **53**, 983-997.

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.

R Core Team (2025) `R`: *A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.r-project.org/.

Snee, R. D. (1981). Graphical Display and Assessment of Means. *Biometrics*, **37**, 835–836.

Stefanova, K. T., Smith, A. B. & Cullis, B. R. (2009) Enhanced diagnostics for the spatial analysis of field trials. *Journal of Agricultural, Biological, and Environmental Statistics*, **14**, 392–410.

Verbyla, A. P. (2019). A note on model selection using information criteria for general linear models estimated using REML. *Australian & New Zealand Journal of Statistics*, **61**, 39-50.https://doi.org/10.1111/anzs.12254/.