Package 'agriTutorial'

July 22, 2025

Type Package

Title Tutorial Analysis of Some Agricultural Experiments

Version 0.1.5

Date 2019-06-01

Maintainer Rodney Edmondson <rodney.edmondson@gmail.com>

Depends R (>= 3.1.0)

Description Example software for the analysis of data from designed
experiments, especially agricultural crop experiments. The basics
of the analysis of designed experiments are discussed
using real examples from agricultural field trials.
A range of statistical methods using a range
of R statistical packages are exemplified . The experimental
data is made available as separate data sets for each example
and the R analysis code is made available as example code.
The example code can be readily extended, as required.

License GPL (>= 2)

Imports lmerTest, emmeans, pbkrtest, lattice, nlme, ggplot2

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Suggests R.rsp

VignetteBuilder R.rsp

NeedsCompilation no

Author Rodney Edmondson [aut, cre],
Hans-Peter Piepho [aut, ctb],
Muhammad Yaseen [aut, ctb]

Repository CRAN

Date/Publication 2019-06-01 15:30:03 UTC

Contents

---

agriTutorial   *Tutorial Analysis of Agricultural Experiments*

Description

The `agriTutorial` package provides R software for the analysis of five agricultural example data sets as discussed in the paper: 'A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels' by Piepho and Edmondson (2018). See: View

Details

Code

The example code produces statistical analysis for the five agricultural data sets in Piepho & Edmondson (2018) and also produces additional graphical analysis. The data for each example is provided as a data frame which loads automatically whenever the package is loaded and the code for each analysis is provided as a set of examples.

Printed output defaults to the device terminal window but can be diverted to a suitable text file by using a sink file command: see `help(sink)`, if required. Similarly, graphical output defaults to the device graphics window but can be diverted to a suitable graphics device if required: see `vignette('agriTutorial')` for further details,

The example code demonstrates some basic methodology for the analysis of data from designed experiments but there are other methods available in R and it is straightforward to extend the example code by adding functionality from other packages. One source of package information is the set of 'task views' available at: Task Views.

Polynomials

The polynomials used in this tutorial are either raw polynomials or orthogonal polynomials.

A raw polynomial is a weighted sum of the powers and cross-products of a set of treatment or nuisance effect vectors.

An orthogonal polynomial is a weighted sum of orthogonal combinations of the powers and cross-products of a set of treatment or nuisance effect vectors.

Raw polynomials have a direct interpretation as a fitted polynomial model but can be numerically unstable whereas orthogonal polynomials are numerically stable but give coefficients which are linear combinations of the required polynomial model coefficients and are difficult to interpret.

Raw polynomials are the polynomials of choice for most analyses but sometimes orthogonal polynomials can be useful when, for example, fitting higher-degree polynomials in a long series of repeated measures (see example 4).

Functional marginality

Polynomial expansions are based on a Taylor series expansion and normally must include all polynomial terms up to and including the maximum degree of the expansion. This is the property of functional marginality and applies to any polynomial or response surface model including models with polynomial interaction effects (Nelder, 2000). In this tutorial, all polynomial and response surface models will be assumed to conform with the requirements of functional marginality.

Packages

The example code depends on a number of R packages each of which must be installed on the user machine before the example code can be properly executed. The required packages are lmerTest, emmeans, pbkrtest, lattice, nlme and ggplot2, all of which should install automatically. If, for any reason, packages need to be installed by hand, this can be done by using `install.packages("package name")`.

NB. It is important to keep packages updated using the update.packages() command.

Examples:

1. example1 : split-plot design with one quantitative and one qualitative treatment factor

2. example2 : block design with one qualitative treatment factor

3. example3 : response surface design with two quantitative treatment factors

4. example4 : repeated measures design with one quantitative treatment factor

5. example5 : block design with transformed quantitative treatment levels

References

Piepho, H. P, and Edmondson. R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science. DOI: 10.1111/jac.12267. View

Taylor series. From Wikipedia, the free encyclopedia https://en.wikipedia.org/wiki/Taylor_series

Nelder, J. A. (2000). Functional marginality and response-surface fitting. Journal of Applied Statistics, 26, 109-122.

---

beet   *Beet data for Example 2*

Description

Petersen (1994, p. 125) describes an experiment conducted to assess the effects of five different quantities of N-fertiliser (0, 35, 70, 105 and 140 kg N/ha) on root dry matter yield of sugar beets (t/ha) with three complete replications laid out in three randomized complete blocks. One objective of this experiment is to determine the amount of fertilizer maximizing yield. Petersen, R.G. (1994). Agricultural field experiments. Design and analysis. New York: Marcel Dekker.

Usage

```
data(beet)
```

Format

An object of class data.frame with 15 rows and 3 columns.

---

example1   *Example 1: Split-plot design with one qualitative and one quantitative level factor*

Description

Gomez & Gomez (1984, p. 143) report a rice experiment with three management practices (minimum, optimum, intensive), five different amounts of nitrogen (N) fertilizer (0, 50, 80, 110, 140 kg/ha), and three varieties (V1, V2, V3). The experiment involved variety and management as qualitative treatment factors and nitrogen fertilizer as a quantitative treatment factor. Overall, there were 45 treatments with three replicates in complete replicate blocks. The fertilizer treatments were applied to main plots, the management practices to split-plots and the varieties to split-split-plots.

Details

Section 1

Section 1 examines treatment effects by fitting qualitative factorial models and the first analysis calculates a full analysis of variance (Table 1) for main plots (nitrogen), split-plots (management) and split-split-plots (variety). Each type of experimental unit (or "stratum") requires a separate error term in the fitted analysis.

The second analysis (Table 2) uses a REML mixed model analysis to find treatment means and SE's for each marginal treatment classification averaged over all the other treatment factors, together with estimates of pairwise contrasts of treatment means and the SE's of the pairwise treatment comparisons. This analysis fits the full set of nitrogen-by-variety interaction effects assuming additive management effects and the fit of the model is tested by a graphical plot of the model residuals. Residual plots provide an important check on model assumptions but many more options for model

example1 5

testing are available and further methods for diagnostic testing are examined in the subsequent examples.

The third analysis (Table 3) shows a mixed model analysis of the full factorial model fitted by REML using the lmer function of the lme4 package. Generally with mixed models, determination of the denominator degrees of freedom for Wald-type F- and t-statistics becomes an issue, and here we use the method proposed by Kenward & Roger (1997).

Section 2

Section 2 examines treatment effects by fitting polynomial models and the first step calculates a full set of four raw polynomials for the 5-levels of N using the poly() function. The N rates are re-scaled by division by 100 to improve numerical stability.

The second step fits a mixed model polynomial analysis of nitrogen effects assuming additive management effects (Table 7). In this analysis, most of the nitrogen treatment effect can be explained by linear and quadratic trend effects. but it is important to note that there is a non-negligible Variety x Cubic N interaction effect. This suggests that not all the varieties responded in a similar way to the N treatments and that some further analysis of the data may be required (see also the N plots of individual varieties and replicates in Fig 1).

The third step fits the required model for the actual fitted model coefficients (Table 8). When estimating model effects, only effects that are significant for the fitted model or that are marginal to those effects (functional marginality) should be included in the model therefore only linear and quadratic nitrogen effects are included in this model. The fitted model for the nitrogen effects fits the actual actual nitrogen levels used in the experiment therefore this model provides the required coefficients for the actual applied nitrogen levels.

Section 3

Section 3 provides checks on some of the assumptions underlying the blocks-by-treatments model.

The first analysis in this section shows a complete partition of the blocks-by-treatments interaction effects into factorial mean square terms where all the terms that contain a replicate:variety interaction effect are estimates of the split-split-plot error variance. If the blocks-by-treatments assumptions are valid, all the estimates of the split-split-plot error variance are expected to have the same error mean square. However, the Replicate:variety effect has a mean square of 1.54 on 4 degrees of freedom whereas the Replicate:management:variety:nitrogen effect has a mean square of 0.26 on 32 degrees of freedom. The ratio of these mean squares is 5.92 with an F-probability of 0.00110 on 4 and 32 degrees of freedom, which means that the Replicate:variety interaction effect is significantly inflated relative to the Replicate:management:variety:nitrogen effect. This shows that the assumptions underlying the blocks-by-treatments analysis of the model are invalid with a high level of probability.

The 4 degrees of freedom in the Replicate:variety interaction effect are the differences between the three varieties differenced between the three replicate blocks. Fig S1 shows graphical plots of variety effects in each replicate block averaged over management effects, and there is clear evidence that the effects of Variety 1 in blocks 1 and 2 were different from the effects of Variety 1 in block 3.

The second analysis in Section 3 shows a complete partition of the blocks-by-treatments interaction effects into factorial mean square terms ignoring Variety 1. This analysis shows a reasonably good fit to the assumed additive block which supports the hypothesis that the non-additivity of the block-and-treatment effects in the full unrestricted analysis is mainly due to Variety 1.

The final analysis in Section 3 shows an analysis of variance of the treatment effects ignoring Variety 1. In this analysis, the management:variety interaction effect becomes significant at the

0.00992 probability level compared with a non-significant management:variety interaction effect in the analysis of the full data set.

Such anomalies are not uncommon in the analysis of real data sets and it is the task of the statistician to identify anomalies as and when they occur. Factorial designs can be very powerful for practical research but, as demonstrated with this data set, the analysis of such designs is complex and anomalies can be easily missed. Unless an anomaly is due to an easily identified cause such as an incorrectly recorded data point, it is likely that the anomaly will need to be investigated by further discussion with the research workers. It is a mistake to suppose that data from a designed experiment can be analysed statistically in isolation from the research workers who conducted the experiment.

agriTutorial: return to home page if you want to select a different example

References

Gomez, K.A., & Gomez, A.A. (1984). Statistical procedures for agricultural research, 2nd edn. New York: Wiley.

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

Piepho, H. P, and Edmondson. R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science. DOI: 10.1111/jac.12267. View

Examples

```
## ********************************************************************************
##                          How to run the code
## ********************************************************************************

## Either type example("example1") to run ALL the examples succesively
## or copy and paste examples sucessively, as required

## ********************************************************************************
##                      Options and required packages
## ********************************************************************************

## Packages lmerTest, emmeans and pbkrtest MUST be installed
require(lmerTest)
require(emmeans)
require(pbkrtest)
options(contrasts = c('contr.treatment', 'contr.poly'))

## ********************************************************************************
##              Section 1: Qualitative analysis of factorial treatment effects
## ********************************************************************************

## Table 1 Full analysis of rice data assuming qualitative nitrogen effects
rice.aov1 = aov(yield ~ Replicate + management * variety * nitrogen +
Error(Replicate/Main/Sub), rice)
summary(rice.aov1, ddf = "Kenward-Roger", type = 1)
```

example1 7

```
## Table 2 REML means and se's for additive management and qualitative nitrogen effects
rice.means = lmer(yield ~ Replicate + management + nitrogen * variety +
 (1|Replicate:Main) + (1|Replicate:Main:Sub), data = rice)
anova(rice.means, ddf = "Kenward-Roger", type = 1)
plot(rice.means, sub.caption = NA, ylab = "Residuals", xlab = "Fitted",
 main = "Full analysis with full nitrogen effects")
emmeans::emmeans(rice.means, ~ nitrogen)
emmeans::emmeans(rice.means, ~ variety)
emmeans::emmeans(rice.means, ~ nitrogen * variety)

## REML contrasts and sed's for additive management and qualitative nitrogen effects
n.v = emmeans::emmeans(rice.means, ~ nitrogen|variety)
emmeans::contrast(n.v, alpha = 0.05, method = "pairwise")
v.n = emmeans::emmeans(rice.means, ~ variety|nitrogen)
emmeans::contrast(v.n, alpha = 0.05, method = "pairwise")

## Table 3 Mixed model effects for rice data with significance tests
rice.lmer = lmer(yield ~ Replicate + nitrogen * management * variety + (1|Replicate:Main) +
 (1|Replicate:Main:Sub), data = rice)
anova(rice.lmer, ddf = "Kenward-Roger", type = 1)


## *********************************************************************************
##              Section 2: Quantitative analysis of factorial treatment effects
## *********************************************************************************

## adds raw N polynomials to data frame: note that the nrate is re-scaled
N = poly((rice$nrate/100), 4, raw = TRUE)
colnames(N) = c("Linear_N", "Quadratic_N", "Cubic_N", "Quartic_N")
rice = cbind(rice, N)

## Table 7: Mixed model fitting raw polynomials for nitrogen effects
rice.fullN = lmer(yield ~ Replicate + management + variety * (Linear_N + Quadratic_N +
 Cubic_N + Quartic_N) + (1|Replicate:Main) + (1|Replicate:Main:Sub), data = rice)
anova(rice.fullN, ddf = "Kenward-Roger", type = 1)

## Table 8 Coefficients for separate linear and common quadratic N with additive management
rice.quadN = lmer(yield ~ Replicate + management + variety * Linear_N + Quadratic_N +
 (1|Replicate:Main) + (1|Replicate:Main:Sub), data = rice)
summary(rice.quadN, ddf = "Kenward-Roger")


## *********************************************************************************
##                         Section 3: Model assumptions
## *********************************************************************************

## Full analysis of variance of block and treatment effects showing large mean square error
## due to variety-by-replicates interaction effects
rice.fullaov = aov(yield ~ Replicate*management * variety * nitrogen, rice)
summary(rice.fullaov, ddf = "Kenward-Roger", type = 1)

## Fig S1 Nitrogen response per variety per plot showing anomalous behaviour of Variety 1
```

```
## in Blocks 1 and 2 compared with Block 3
Rice = aggregate(rice$yield, by = list(rice$Replicate, rice$nitrogen, rice$variety),
 FUN = mean, na.rm = TRUE)
colnames(Rice) = c("Reps", "Nlev", "Vars", "Yield")
wideRice = reshape(Rice, timevar = "Nlev", idvar = c("Vars", "Reps"), direction = "wide")
wideRice = wideRice[,-c(1, 2)]
N = c(0, 50, 80, 110, 140)
par(mfrow = c(3, 3), oma = c(0, 0, 2, 0))
for (i in 1:3) {
for (j in 1:3) {
plot(N, wideRice[(i - 1) * 3 + j, ], type = "l", ylab = "yield",
main = paste("Variety",i,"Block",j), ylim = c(0, max(wideRice)))
}
}
title(main = "Fig S1. Variety response to nitrogen for individual replicate blocks", outer = TRUE)

## Subset of data excluding variety 1
riceV2V3=droplevels(rice[rice$variety != "V1",])

## Restricted analysis of variance of block and treatment effects excluding variety 1
## compare variety-by-replicates interaction effects of full and restricted analysis
rice.fullaov = aov(yield ~ Replicate*management * variety * nitrogen, riceV2V3)
summary(rice.fullaov, ddf = "Kenward-Roger", type = 1)

## Restricted analysis assuming qualitative nitrogen effects excluding variety 1
rice.aov1 = aov(yield ~ Replicate + management * variety * nitrogen +
Error(Replicate/Main/Sub), riceV2V3)
summary(rice.aov1, ddf = "Kenward-Roger", type = 1)
```

---

example2	*Example 2: Lack-of-fit and marginality for a single quantitative treatment factor*

Description

Petersen (1994, p. 125) describes an experiment conducted to assess the effects of five different quantities of N-fertiliser (0, 35, 70, 105 and 140 kg N/ha) on root dry matter yield of sugar beet (t/ha) with three complete replications laid out in three randomized complete blocks. One objective of this experiment was to determine the amount of fertilizer for maximizing yield.

Details

The first stage of the analysis is the calculation of raw polynomial powers of N using the poly() function. The N rates are re-scaled by division by 100 to improve numerical stability.

The second stage fits a full polynomial analysis of variance based on polynomial contrasts which are fitted in sequence from the lowest to the highest. This is equivalent to the analysis shown in Tables 4 and 5 of Piepho and Edmondson (2018) except that a complete partition into single degree

example2 9

of freedom polynomial contrasts is shown here compared with the pooled 'lack of fit' term shown in Tables 4 and 5.

The third stage fits a quadratic regression model with linear and quadratic terms only. This model provides the model coefficients, standard errors and the confidence intervals shown in Table 6 of Piepho and Edmondson (2018). A set of diagnostic plots are fitted for the fitted quadratic regression model to check the validity of the model assumptions.

Finally, a smoothed quadratic graph of the yield versus the N rate is plotted to show the goodness of fit of the quadratic regression model. This plot corresponds to plot Fig 3 in Piepho and Edmondson (2018).

agriTutorial: return to home page if you want to select a different example

References

Petersen, R.G. (1994). Agricultural field experiments. Design and analysis. New York: Marcel Dekker.

Piepho, H. P, and Edmondson. R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science. DOI: 10.1111/jac.12267. View

Examples

```
## ******************************************************************************
##                         How to run the code
## ******************************************************************************

## Either type example("example2") to run ALL the examples succesively
## or copy and paste examples sucessively, as required


## ******************************************************************************
##                      Options and required packages
## ******************************************************************************

options(contrasts = c('contr.treatment', 'contr.poly'))
## ggplot2 MUST be installed
require(ggplot2)


## ******************************************************************************
##          Polynomial analysis and graphical plots of factorial treatment effects
## ******************************************************************************

N = poly((beet$nrate/100), degree = 4, raw = TRUE)
colnames(N) = c("Linear_N", "Quadratic_N", "Cubic_N", "Quartic_N")
beet = cbind(beet, N)

## Tables 4 and 5: Full polynomial analysis of variance based on raw polynomials
anova(lm(yield ~ Replicate + Linear_N + Quadratic_N + Cubic_N + Quartic_N, data = beet))

## Table 6: showing quadratic model coefficients with standard errors and confidence intervals
quadratic = lm(yield ~ Replicate + Linear_N + Quadratic_N, data = beet)
```

```
summary(quadratic)
confint(quadratic, level = 0.95)

par(mfrow = c(2, 2), oma = c(0, 0, 2, 0))
plot(quadratic, sub.caption = NA)
title(main = "Diagnostic plots for quadratic nitrogen effects model", outer = TRUE)

ggplot(beet, aes(x = nrate, y = yield)) +
 ggtitle("Fig 3 Yield versus N for sugar beet with 95% confidence band") +
 geom_point(shape = 1) + stat_summary(fun.y = mean, geom = "point") +
 geom_smooth(method = lm, formula = y ~ poly(x, 2)) + theme_bw()
```

---

example3   *Example 3: Polynomial regression model with two quantitative level treatment factors*

Description

(Gomez & Gomez, 1984, p. 401) report a two-factor nitrogen uptake greenhouse experiment on rice involving duration of water stress (W) and level of nitrogen application (N) with four complete replicates of each treatment. The experiment had four water-stress levels (0, 10, 20 and 40 days) applied as main-plot treatments and four nitrogen rates (0, 90, 180 and 270 kg/ha) applied as sub-plot treatments. The four sub-plot treatments were randomized within main plots and the four main plot treatments were randomized within complete replicate blocks.

Details

The first stage of the analysis is the calculation of polynomial powers of N and W using the poly() function. The N rates are re-scaled by division by 100 while the W rates are re-scaled by division by 10.

The second stage shows a Pearson residual plot of the untransformed N uptake data versus a Pearson residual plot of the log transformed N uptake data. Comparison of the two plots shows that the untransformed residuals increase as the fitted values increase whereas the log transformed N uptake residuals are approximately constant over the full range of the fitted values. This shows that a log transformation of the N uptake data gives a dependent variate with constant variance over the full range of fitted values which shows that a simple unweighted analysis of variance is valid for the effects of the treatment factors.

Sometimes the original scale of measurement is the proper scale of measurement for an analysis, e.g. an analysis of actual measured crop yields, and then it might be appropriate to fit a weighted analysis of variance in the original scale of measurement of the dependent variable (see Faraway 2002 Chapter 5). However, the log transformation model assumes a proportional rather than an additive model for treatment effects and, in this example, a proportional model for nitrogen uptake may well be a more natural physiological model than a simple additive model.

The next stage compares the fit of a first-order linear model (Table 9) versus a second-order quadratic model (Table 10). The first-order model shows significant lack-of-fit and is not adequate for the data. The second-order model is also not fully adequate for the data as there is a significant N lack of fit

example3 11

term indicating a significant cubic effect. However, the magnitude of the cubic effect is relatively small and it will be assumed here that a quadratic model is adequate for the data.

The final stage fits regression coefficients for the quadratic response surface model on the re-scaled water stress and re-scaled nitrogen rate treatments. The fitted coefficients are then used to plot the fitted quadratic log uptake curves versus the nitrogen rate treatments and the water stress treatments, as shown in Fig 4.

Note that in this analysis all the polynomial models are built by adding individual polynomial effects in accordance with the requirements of functional marginality.

agriTutorial: return to home page if you want to select a different example

References

Faraway J (2002) Practical Regression and Anova using R. https://cran.r-project.org/doc/contrib/Faraway-PRA.pdf

Gomez, K.A., & Gomez, A.A. (1984). Statistical procedures for agricultural research, 2nd edn. New York: Wiley.

Piepho, H. P, and Edmondson. R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science. DOI: 10.1111/jac.12267. View

Examples

```
## ********************************************************************************
##                          How to run the code
## ********************************************************************************


## Either type example("example3") to run ALL the examples succesively
## or copy and paste examples sucessively, as required


## ********************************************************************************
##                  Options and required packages
## ********************************************************************************

require(lmerTest)
require(lattice)
require(pbkrtest)
options(contrasts = c('contr.treatment', 'contr.poly'))


## ********************************************************************************
##              Section 1: Polynomial powers of N and W
## ********************************************************************************

greenrice$loguptake = log(greenrice$uptake)
greenrice$Nitrogen = factor(greenrice$N)
greenrice$Water = factor(greenrice$W)
PolW = poly((greenrice$W/10), degree = 2, raw = TRUE)
colnames(PolW) = c("Linear_W", "Quadratic_W")
PolN = poly((greenrice$N/100), degree = 2, raw = TRUE)
colnames(PolN) = c("Linear_N", "Quadratic_N")
```

```
greenrice = cbind(greenrice, PolW, PolN)

## residual plot of untransformed N uptake data
greenrice.uptake = lmer(uptake ~ Replicate + factor(N) * factor(W) +
 (1|Replicate:Main), data = greenrice)
plot(greenrice.uptake, main = "Pearson residual plot for untransformed N uptake",
 ylab = "Residuals N uptake")

## residual plot of log transformed N uptake data
greenrice.loguptake = lmer(loguptake ~ Replicate + factor(N) * factor(W) +
 (1|Replicate:Main), data = greenrice)
plot(greenrice.loguptake, main = "Pearson residual plot for log transformed N uptake",
 ylab = "Residuals log N uptake")

## Table 9: first-order model of log uptake with Wald tests
greenrice.lmer1 = lmer(loguptake ~ Linear_N + Linear_W + Nitrogen * Water +
 (1|Replicate) + (1|Replicate:Main), data = greenrice)
anova(greenrice.lmer1, ddf = "Kenward-Roger", type = 1)

## Table 10: second-order model of log uptake with Wald tests
greenrice.lmer2 = lmer(loguptake ~ Linear_N * Linear_W + Quadratic_N + Quadratic_W +
 Nitrogen * Water + (1|Replicate) + (1|Replicate:Main), data = greenrice)
anova(greenrice.lmer2, ddf = "Kenward-Roger", type = 1)

## *************************************************************************************
##          Section 2 : Fitted regression models and quadratic log uptake curves
## *************************************************************************************

## Regression coefficients of quadratic response model of W and N
greenrice.lmer0 = lmer(loguptake ~ Linear_N * Linear_W + Quadratic_N +
 Quadratic_W + (1|Replicate) + (1|Replicate:Main), data = greenrice)
summary(greenrice.lmer0, ddf = "Kenward-Roger", type = 1)


## Fig 4a fitted quadratic loguptake curve versus water stress treatments
panel.plot = function(x, y) {
panel.xyplot(x, y) # shows observed points
Nitrogen = c(0, .90, 1.80, 2.70)[panel.number()]
panel.curve(-1.16 + 0.17603 * x - 0.11599 * x * x + 0.68 * Nitrogen -
 0.0938 * Nitrogen * Nitrogen - 0.09072 * x * Nitrogen,
from = 0, to = 4.0, type = "l", lwd = 2)
}
xyplot(loguptake ~ Linear_W|factor(Linear_N), data = greenrice,
 scales = list(x = list(at = c(0, 1, 2, 4), labels = c(0, 10, 20, 40))),
 main = "Fig 4a: logN uptake versus water stress",
 xlab = " Water stress (days)", ylab = "Log nitrogen uptake (g/pot)",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("0", "90", "180", "270")),
panel = panel.plot)

## Fig 4b fitted quadratic quadratic loguptake curve versus nitrogen rate treatments
panel.plot = function(x, y) {
panel.xyplot(x, y) # shows observed points
```

example3 13

```
Water = c(0, 1.0, 2.0, 4.0)[panel.number()]
panel.curve( -1.16 + 0.17603 * Water - 0.11599 * Water * Water +
 0.68 * x - 0.0938 * x * x - 0.09072 * Water * x ,
from = 0, to = 2.70, type = "l", lwd = 2)
}
xyplot(loguptake ~ Linear_N|factor(Linear_W), data = greenrice,
 scales = list(x = list(at = c(0, .9, 1.8, 2.7), labels = c(0, 90, 180, 270))),
 main = "Fig 4b: logN uptake versus nitrogen rate",
 xlab = "Nitrogen (kg/ha)", ylab = "Log nitrogen uptake (g/pot)",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("0", "10", "20", "40")),
panel = panel.plot)




## Fig 4a backtransformed quadratic loguptake curve versus water stress treatments
panel.plot = function(x, y) {
panel.xyplot(x, y) # shows observed points
Nitrogen = c(0, .90, 1.80, 2.70)[panel.number()]
panel.curve( exp(-1.16 + 0.17603 * x - 0.11599 * x * x + 0.68 * Nitrogen -
 0.0938 * Nitrogen * Nitrogen - 0.09072 * x * Nitrogen),
from = 0, to = 4.0, type = "l", lwd = 2)
}

xyplot(uptake ~ Linear_W|factor(Linear_N), data = greenrice,
 scales = list(x = list(at = c(0, 1, 2, 4), labels = c(0, 10, 20, 40))),
 main = "Fig 4a: Back transformed N uptake versus water stress",
 xlab = " Water stress (days)", ylab = "Nitrogen uptake (g/pot)",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("0", "90", "180", "270")),
panel = panel.plot)

## Fig 4b back transformed quadratic loguptake curve versus nitrogen rate treatments
panel.plot = function(x, y) {
panel.xyplot(x, y) # shows observed points
Water = c(0, 1.0, 2.0, 4.0)[panel.number()]
panel.curve(exp(-1.16 + 0.17603 * Water - 0.11599 * Water * Water +
 0.68 * x - 0.0938 * x * x - 0.09072 * Water * x),
from = 0, to = 2.70, type = "l", lwd = 2)
}
xyplot(uptake ~ Linear_N|factor(Linear_W), data = greenrice,
 scales = list(x = list(at = c(0, .9, 1.8, 2.7), labels = c(0, 90, 180, 270))),
 main = "Fig 4b: Back transformed N uptake versus nitrogen rate",
 xlab = "Nitrogen (kg/ha)", ylab = "Nitrogen uptake (g/pot)",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("0", "10", "20", "40")),
panel = panel.plot)
```

---

example4   *Example 4: One qualitative treatment factor with repeated measure-*
                                      ments over time.

Description

Milliken & Johnson (1992, p. 429) discuss data which they describe as repeated leaf index mea-
surements on sorghum. Their data set comprises five replicate blocks of four sorghum varieties and
they assume equally spaced repeated measurements on each plot in each block on five consecutive
occasions starting two weeks after emergence. No further information is given but it appears that the
data is simulated or made-up rather than real. Although real data is more authentic, it can sometimes
be useful to discuss the analysis of an example data set from the literature, even when the data is
simulated. Milliken & Johnson discuss multivariate analysis of variance of the data but this method
take no account of the ordered relationship between repeated observations or the likely correlation
structure of the data and we will discuss alternative correlation models that are specifically intended
to account for the underlying structure of repeated measures data. The interested reader can, if
desired, refer to Milliken & Johnson (1992) Chapter 31 for comparison of the different approaches.

Details

Section 1 calculates polynomials for weeks and blocks using the `poly()` function. Two sets of
polynomials for weeks, raw and orthogonal, are calculated and saved as `sorghum$rawWeeks` and
`sorghum$polWeeks` respectively. Orthogonal polynomials for blocks are calculated and saved as
`sorghum$polBlocks`. It is important to note that the `poly()` function calculates all polynomial
contrasts up to the required degree but does NOT include the zero-degree polynomial. Additionally,
the block variable `varblock` is saved as a factor `factblock`.

Section 2 compares five different correlation structures for the repeated measures analysis using
the gls() function of the nlme package. Each analysis fits a full factorial model for the variety-
by-weeks and blocks-by-weeks effects assuming block and treatment additivity. The goodness of
fit of the five models is compared by AIC statistics where the smaller the AIC the better the fit.
Here, the AR(1)+nugget model fitted by the `corExp()` function gave the best fitting model. See
`help(corExp)` for further information about the corExp() function. Note that `corSymm` represents a
general correlation structure and will, presumably, give an analysis similar to a multivariate anal-
ysis of variance. Although this structure appears to give the best fit according to the negative log
likelihood statistic, this criterion takes no account of the number of estimated variance parameters
p in the variance model which, in the case of the `corSymm` model, is p = 15, compared to only p = 2
for the AR(1) model. When assessed by the AIC statistic, the `corSymm` model gave the least good fit
of any of the non-null correlation structures which is strong evidence that the multivariate analysis
of variance method discussed by Milliken & Johnson (1992) will lack power.

Section 3 fits a full regression model over the five weeks of repeated measures and tests for pos-
sible variety and variety-by-weeks interactions effects. The weeks factor is decomposed into indi-
vidual polynomial contrasts (see Table A2 and Table 14) to test the significance of each individual
variety-by-weeks polynomial effect. The analysis of polynomial contrasts shows that the variety-
by-weeks interaction is due mainly to the degree-1 = `variety:rawWeeks[,1]` and the degree-2 =
`variety:rawWeeks[,2]` effects, although there is also some evidence of higher-degree variety-by-
weeks interaction effects. The analysis also shows the corExp() range and nugget statistics for

example4 15

the full fitted model and these are used to calculate the correlation coefficient usingthe formula `rho=(1-nugget)*exp(-1/range)`. Note that this formula is different from the the formula used in Tables A1 and A2 and will give a different value of `rho`: see `help(corExp)`.

Section 4 fits a quadratic regression model for weeks assuming the degree-3 and degree-4 polynomial week effects are zero. The average effects of blocks are fitted by `polBlocks` and the interactions between the blocks and the weeks are fitted by `polBlocks:(rawWeeks[,1] + rawWeeks[,2] + polWeeks[,3]+ polWeeks[,4])`. The `gls()` algorithm requires the same polynomial weeks contrasts in both the blocks and the varieties models which is why raw degree-1 and degree-2 weeks contrasts have been used for the blocks-by-weeks interaction model. However, orthogonal polynomials have better numerical stability than raw polynomials so orthogonal polynomial contrasts have been used for the degree-3 and degree-4 weeks contrasts. The summary analysis shows all variety effects as differences from the intercept which, in this analysis, is variety 1 therefore all model effects in Table 15 can be derived by adding appropriate effects to the intercept. If SED's are required, these must be calculated from the variance/covariance matrix which can be extracted by the code `vcov()`. Using this matrix, the SED for variety differences was calculated to be 0.172, the SED for the variety-by-linear weeks slope parameters was calculated to be 0.117 and the SED for the variety-by-quadratic weeks slope parameters was calculated to be 0.0192. These estimates are approximately 2-3 percent larger than those shown in Table 15 but it is not clear if the discrepancies are due to the model specification or to a difference between the R and the SAS software. Possibly the implementation of the Kenward-Roger method of adjusting the denominator d.f. and the estimated variance-covariance matrix of the estimated fixed effects might be different for the two algorithms. The range, nugget and correlation coefficient are extracted and displayed and a graphical plot of the studentized residuals from the quadratic regression model is also shown.

Section 5 fits a quadratic regression model for variety-by-week interaction effects assuming a full degree-4 polynomial model for weeks and blocks-by-weeks effects. The quadratic regression model in Section 4 corresponds to the regression model used for Tables 14 and 15 of Piepho and Edmondson (2018) but the range = 3397131013 and nugget = 0.4605535 of this model are very different from the range = 10.35774 and nugget = 0.1720444 of the full factorial model. As there is evidence from Table A2 that the degree-3 and degree-4 polynomial weeks effects are non-negligible, the quadratic model for weeks effects in Section 4 may be inadequate for the data and the model may be underfitted. In this section, the assumption that the degree-3 and degree-4 polynomial weeks effects are zero is relaxed and a full degree-4 model for weeks and block-by-weeks interaction effects is fitted. The fitted model for treatment effects needs to be as parsimonious as possible to ensure that estimates of treatment effects are robust against model assumptions and a degree-2 regression model for variety-by-weeks effects appears to be the most appropriate treatment model for this data. With this model, the values of the auto-correlation parameters are: range = 42.75763, nugget = 0.3586337 and correlation = 0.6265403 which are much closer to the autocorrelation parameters from the full factorial model than are those from Section 4. As the model fits the full polynomial weeks model, it is not necessary to use polynomial blocks contrasts which gives a substantial simplification in coding.

Comment The model fitted in Section 5 appears to be the best model available based on the generalized least squares method but it is clear from the graphical plots of studentized residuals that the fitted data contains outliers that are not well accommodated by the fitted model. If the data was from a real experiment, further information about the data might be available but as the data seems to be artificial this option is not available. In this situation, various robust methods of model fitting or regression analysis that can accommodate non-standard distributions or model outliers are available. However, these methods are beyond the scope of this tutorial and will not be discussed further here.

[agriTutorial](): return to home page if you want to select a different example

References

Milliken, G.A., & Johnson, D.E. (1992). Analysis of messy data. Volume I: Designed experiments. Boca Raton: CRC Press.

Piepho, H. P, and Edmondson. R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science. DOI: 10.1111/jac.12267. View

Examples

```
## ********************************************************************************
##                          How to run the code
## ********************************************************************************

## Either type example("example4") to run ALL the examples succesively
## or copy and paste examples sucessively, as required

## ********************************************************************************
##                        Options and required packages
## ********************************************************************************

options(contrasts = c('contr.treatment','contr.poly'))
require(nlme)

## ********************************************************************************
##            Section 1:  Polynomials for weeks and blocks contrasts
## ********************************************************************************

sorghum$rawWeeks = poly(sorghum$varweek, degree = 4, raw = TRUE)
sorghum$polWeeks = poly(sorghum$varweek, degree = 4, raw = FALSE)
sorghum$polBlocks = poly(sorghum$varblock, degree = 4, raw = FALSE)
sorghum$factblock = factor(sorghum$varblock)

## ********************************************************************************
## Section 2: Various correlation models assuming full factorial blocks and weeks model
## ********************************************************************************

AIC = NULL
logLik = NULL
Model = c("ID", "CS", "AR(1)", "AR(1) + nugget", "UN")

## independent uncorrelated random plots
full_indy = gls(y ~ factweek * (Replicate + variety), sorghum)
anova(full_indy)
AIC = c(AIC, AIC(full_indy))
logLik = c(logLik, logLik(full_indy))

## corCompSymm compound symmetry
corCompSymm = gls(y ~ factweek * (Replicate + variety),
```

example4 17

```
corr = corCompSymm(form = ~ varweek|factplot), sorghum)
anova(corCompSymm)
AIC = c(AIC, AIC(corCompSymm))
logLik = c(logLik, logLik(corCompSymm))
Variogram(corCompSymm)

## corExp without nugget
corExp = gls(y ~ factweek * (Replicate + variety),
 corr = corExp(form = ~ varweek|factplot), sorghum)
anova(corExp)
AIC = c(AIC, AIC(corExp))
logLik = c(logLik, logLik(corExp))
Variogram(corExp)

##  corExp with nugget
corExp_nugget = gls(y ~ factweek * (Replicate + variety),
 corr = corExp(form = ~ varweek|factplot, nugget = TRUE), sorghum)
anova(corExp_nugget)
AIC = c(AIC, AIC(corExp_nugget))
logLik = c(logLik, logLik(corExp_nugget))
Variogram(corExp)

##  corSymm unstructured
corSymm = gls(y ~ factweek * (Replicate + variety), corr = corSymm(form = ~ 1|factplot),
 weights = varIdent(form = ~ 1|varweek), sorghum)
anova(corSymm)
AIC = c(AIC, AIC(corSymm))
logLik = c(logLik, logLik(corSymm))
Variogram(corSymm)

## Table 11 Comparison of log Likelihood and AIC statistics for different correlation structures
dAIC = AIC - AIC[4]
logLik = -2 * logLik
dlogLik = logLik - logLik[4]
AICtable = data.frame(Model, round(logLik, 2), round(dlogLik, 2), round(AIC, 2), round(dAIC, 2))
colnames(AICtable) = c("Covar_Model", "-2logLr", "-diff2logLr", "AIC", "diffAIC")
AICtable

## ********************************************************************************************
##  Section 3: Factorial block and variety effects assuming full polynomial week effects
## ********************************************************************************************

## Table A2 (cf Table 14) Sequential Wald tests for full model sorghum data
pol_Wald =
gls(y ~ (factblock+variety) * (rawWeeks[,1] + rawWeeks[,2] + polWeeks[,3] + polWeeks[,4]),
corr = corExp(form = ~ varweek | factplot, nugget = TRUE), sorghum)
anova(pol_Wald)
range=coef(pol_Wald$modelStruct$corStruct,unconstrained=FALSE)[1]
nugget=coef(pol_Wald$modelStruct$corStruct,unconstrained=FALSE)[2]
rho=(1-nugget)*exp(-1/range)
cat("Range =", range, "\n")
cat("Nugget =", nugget, "\n")
cat("Correlation =", rho, "\n")
```

```
ACF(pol_Wald)
plot(pol_Wald,sub.caption = NA, main = "Residuals from full polynomial weeks model")

## ***************************************************************************************
## Section 4: Degree-2 model for weeks-by-variety and weeks-by-blocks effects assuming
## degree-3 and degree-4 week effects are zero
## ***************************************************************************************

 ## Table 15 coefficients assuming a quadratic weeks model for both block and treatment effects
quad_Wald = gls(y ~  polBlocks + variety + rawWeeks[,1] + rawWeeks[,2] +
 polBlocks:(rawWeeks[,1] + rawWeeks[,2]+ polWeeks[,3] + polWeeks[,4]) +
 variety:(rawWeeks[,1] + rawWeeks[,2]),
 corr = corExp(form = ~ varweek | factplot, nugget=TRUE), sorghum)
anova(quad_Wald)
summary(quad_Wald)$tTable
vcov(quad_Wald)
range=coef(quad_Wald$modelStruct$corStruct,unconstrained=FALSE)[1]
nugget=coef(quad_Wald$modelStruct$corStruct,unconstrained=FALSE)[2]
rho=(1-nugget)*exp(-1/range)
cat("Range =", range, "\n")
cat("Nugget =", nugget, "\n")
cat("Correlation =", rho, "\n")
plot(quad_Wald,sub.caption = NA, main = "Residuals from quadratic regression model")

## ***************************************************************************************
## Section 5: Quadratic model for weeks-by-variety effects assuming full degree-4 model
## for weeks and weeks-by-blocks effects
## ***************************************************************************************

##  Model assuming a quadratic variety-by-weeks model and quartic blocks-by-weeks model
quad_Wald = gls(y ~ Replicate * (rawWeeks[,1] + rawWeeks[,2] + polWeeks[,3] + polWeeks[,4]) +
 variety * (rawWeeks[,1] + rawWeeks[,2]),
 corr = corExp(form = ~ varweek | factplot, nugget = TRUE), sorghum)
anova(quad_Wald)
summary(quad_Wald)$tTable
range=coef(quad_Wald$modelStruct$corStruct,unconstrained=FALSE)[1]
nugget=coef(quad_Wald$modelStruct$corStruct,unconstrained=FALSE)[2]
rho=(1-nugget)*exp(-1/range)
cat("Range =", range, "\n")
cat("Nugget =", nugget, "\n")
cat("Correlation =", rho, "\n")
plot(quad_Wald,sub.caption = NA, main = "Quadratic treatment-by-weeks model with full
blocks-by-weeks model")
```

---

example5   *Example 5: Transformation of treatment levels to improve model fit*

---

example5 19

Description

Mead (1988, p. 323) describes an experiment on spacing effects with turnips, which was laid out in three complete blocks. Five different seed rates (0.5, 2, 8, 20, 32 lb/acre) were tested in combination with four different row widths (4, 8, 16, 32 inches), giving rise to a total of 20 treatments.

Details

Transformation of the dependent variable will often stabilize the variance of the observations whereas transformation of the regressor variables will often simplify the fitted model. In this example, the fit of a regression model based on the original seed rate and row width variables is compared with the fit of a regression model based on the log transformed seed rates and log transformed row widths. In each case, the model lack-of-fit is examined by assessing the extra variability explained when the Density and Spacing treatment factors and their interactions are added to the quadratic regression models. All yields are logarithmically transformed to stabilize the variance.

The first analysis fits a quadratic regression model of log yields on the untransformed seed rates and row widths (Table 16) while the second analysis fits a quadratic regression model of log yields on the log transformed seed rates and log transformed row widths (Table 17). The analysis of variance of the first model shows that significant extra variability is explained by the Density and Spacing factors and this shows that a quadratic regression model is inadequate for the untransformed regressor variables. The analysis of variance of the second model, however, shows no significant extra variability explained by the Density and Spacing factors and this shows that the quadratic regression model with the log transformed regressor variables gives a good fit to the data and therefore is the preferred model for the observed data.

The superiority of the model with the log transformed regressor variables is confirmed by comparing the fit of the quadratic regression model for the untransformed regressor variables (Figs 8 and 9) versus the fit of the quadratic regression model for the log transformed regressor variables (Figs 10 and 11).

Fig 12a shows diagnostic plots for the fit of a quadratic model with untransformed regressor variables while Fig 12b shows corresponding diagnostic plots for the fit of a quadratic model with loge transformed regressor variables. Each of the four types of diagnostic plots in the two figures shows an improvement in fit for the transformed versus the untransformed regressor variables.

agriTutorial: return to home page if you want to select a different example

References

Mead, R. (1988). The design of experiments. Statistical principles for practical application. Cambridge: Cambridge University Press.

Piepho, H. P, and Edmondson. R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science. DOI: 10.1111/jac.12267. View

Examples

```
## ********************************************************************************
##                        How to run the code
## ********************************************************************************
```

```
## Either type example("example5") to run ALL the examples succesively
## or copy and paste examples sucessively, as required

## *********************************************************************************
##                         Options and required packages
## *********************************************************************************

options(contrasts = c('contr.treatment', 'contr.poly'))
require(lattice)

## *********************************************************************************
##   Quadratic regression models with and without transformation of regressor variables
## *********************************************************************************

RowSpacing = poly(turnip$rowspacing, 3, raw = TRUE)
colnames(RowSpacing) = c("linSpacing", "quadSpacing", "cubSpacing")
Density = poly(turnip$density, 4, raw = TRUE)
colnames(Density) = c("linDensity", "quadDensity", "cubDensity", "quartDensity")
turnip = cbind(turnip, Density, RowSpacing)

## Log transformed row spacing and density polynomials
logRowSpacing = poly(log(turnip$rowspacing), 3, raw = TRUE)
colnames(logRowSpacing) = c("linlogSpacing", "quadlogSpacing", "cublogSpacing")
logDensity = poly(log(turnip$density), 4, raw = TRUE)
colnames(logDensity) = c("linlogDensity", "quadlogDensity", "cublogDensity", "quartlogDensity")
turnip = cbind(turnip, logDensity, logRowSpacing)

## Table 16 Quadratic response surface for untransformed planting density by row spacing model
quad.mod = lm(log_yield ~ Replicate + linDensity * linSpacing + quadDensity + quadSpacing +
 Density * Spacing, turnip)
anova(quad.mod)

## Table 17 Quadratic response surface for transformed log planting density by log row spacing
log.quad.mod = lm(log_yield ~ Replicate + linlogDensity * linlogSpacing +
 quadlogDensity + quadlogSpacing + Density * Spacing, turnip)
anova(log.quad.mod)

## *********************************************************************************
##   Quadratic regression model plots with and without transformations
##   Averaged over replicate blocks to give mean of block effects
## *********************************************************************************

## Quadratic response surface for untransformed planting density by row spacing model
quad.mod = lm(log_yield ~ linDensity * linSpacing + quadDensity + quadSpacing , turnip)
quad.mod$coefficients

##  Fig 8 Plot of loge yield (lb/plot) versus row width
panel.plot = function(x, y) {
panel.xyplot(x, y) # lattice plot shows observed points
SeedDensity = c(0.5,2,8,20,32)[panel.number()]
panel.curve(1.1146900855 + 0.0284788787 * x -0.0007748656 * x * x + 0.1564753713 *SeedDensity -
  0.0033192569 * SeedDensity* SeedDensity  -0.0006749985  * x * SeedDensity,
```

example5 21

```
from = 4, to = 32.0, type = "l", lwd = 2)
}
Seed_Rate=factor(turnip$linDensity)
xyplot(log_yield  ~ linSpacing|Seed_Rate, data = turnip,
 scales = list(x = list(at = c(10,20,30), labels = c(10,20,30))),
 main = "Fig 8: loge yield versus row width",
 xlab = " Row Width ", ylab = "Loge yield ",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("0.5", "2", "8", "20", "32")),
panel = panel.plot)

##  Fig 9 Plot of loge yield (lb/plot) versus seed rate
panel.plot = function(x, y) {
panel.xyplot(x, y) # lattice plot shows observed points
RowWidth = c(4, 8, 16, 32)[panel.number()]
panel.curve(1.1146900855 + 0.1564753713 * x - 0.0033192569 * x * x + 0.0284788787 * RowWidth -
 0.0007748656* RowWidth * RowWidth  -0.0006749985  * x * RowWidth,
from = 0.5, to = 32.0, type = "l", lwd = 2)
}
Row_Width=factor(turnip$linSpacing)
xyplot(log_yield  ~ linDensity|Row_Width, data = turnip,
 scales = list(x = list(at = c(0,10,20,30), labels = c(0,10,20,30))),
 main = "Fig 9: loge yield versus seed rate",
 xlab = " Seed Rate", ylab = "Loge yield ",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("4", "8", "16", "32")),
panel = panel.plot)

## Quadratic response surface for log transformed planting density by log row spacing model
log.quad.mod = lm(log_yield ~ linlogDensity * linlogSpacing + quadlogDensity + quadlogSpacing,
turnip)
log.quad.mod$coefficients
##  Fig 10 Plot of loge yield (lb/plot) versus log row width
panel.plot = function(x, y) {
panel.xyplot(x, y) # lattice plot shows observed points
LogSeedDensity = c(-0.6931472,0.6931472,2.0794415,2.9957323,3.4657359)[panel.number()]
panel.curve( 0.18414803 + 1.09137389 * x - 0.20987137 * x * x +  0.94207543 *LogSeedDensity -
  0.10875560 * LogSeedDensity* LogSeedDensity  -0.09440938   * x * LogSeedDensity,
from = 1.35, to =3.50, type = "l", lwd = 2)
}
xyplot(log_yield  ~ linlogSpacing|Seed_Rate, data = turnip,
 scales = list(x = list(at = c(1.5,2.0,2.5,3.0,3.5), labels = c(1.5,2.0,2.5,3.0,3.5))),
 main = "Fig 10: loge yield versus loge row width",
 xlab = " Loge Row Width ", ylab = "Loge yield ",
 strip = strip.custom(strip.names = TRUE,
factor.levels = c("0.5", "2", "8", "20", "32")),
panel = panel.plot)

##  Fig 11 Plot of loge yield (lb/plot) versus log seed rate
panel.plot = function(x, y) {
panel.xyplot(x, y) # lattice plot shows observed points
LogRowWidth = c(1.386294, 2.079442, 2.772589,3.465736)[panel.number()]
panel.curve(0.18414803  + 0.94207543 * x -0.10875560 * x * x + 1.09137389* LogRowWidth -
```

```
  0.20987137* LogRowWidth * LogRowWidth  -0.09440938 * x * LogRowWidth,
from = -0.7 , to = 3.5, type = "l", lwd = 2)
}
xyplot(log_yield  ~ linlogDensity|Row_Width, data = turnip,
 scales = list(x = list(at = c(0,1,2,3),labels = c(0,1,2,3))),
 main = "Fig 11: loge yield versus loge seed rate",
 xlab = " Loge Seed Rate", ylab = "Loge yield ",
 strip = strip.custom(strip.names = TRUE,
 factor.levels = c("4", "8", "16", "32")),
panel = panel.plot)

## ******************************************************************************************
##   Quadratic regression model diagnostic plots with and without transformations
## ******************************************************************************************

## graphical plots of untransformed data
par(mfrow = c(2, 2), oma = c(0, 0, 2, 0))
fit.quad.mod = lm(log_yield ~ linDensity * linSpacing + quadDensity + quadSpacing,
 turnip)
plot(fit.quad.mod, sub.caption = NA)
title(main = "Fig 12a Diagnostics for untransformed sowing density and row spacing", outer = TRUE)

## graphical plots of log transformed data
par(mfrow = c(2, 2), oma = c(0, 0, 2, 0))
fit.log.quad.mod = lm(log_yield ~ linlogDensity * linlogSpacing + quadlogDensity +
 quadlogSpacing, turnip)
plot(fit.log.quad.mod, sub.caption = NA)
title(main = "Fig 12b Diagnostics for log transformed sowing density and row spacing", outer = TRUE)
```

---

greenrice   *Rice data for Example 3*

Description

(Gomez & Gomez, 1984, p. 401): Nitrogen uptake (g/pot) of rice was studied in a two-factor greenhouse experiment involving duration of water stress (W) and level of nitrogen application (N). The experiment had four water-stress levels (0, 10, 20 and 40 days) as main-plot treatments and four nitrogen rates (0, 90, 180 and 270 kg/ha) as sub-plot treatments. The main plots were randomized in four complete blocks. Gomez, K.A., & Gomez, A.A. (1984). Statistical procedures for agricultural research, 2nd edn. New York: Wiley.

Usage

```
data(greenrice)
```

Format

An object of class data.frame with 64 rows and 5 columns.

---

rice *Rice data for Example 1*

Description

Gomez & Gomez (1984, p. 143) report a rice experiment with three management practices (minimum, optimum, intensive), five different amounts of nitrogen (N) fertilizer (0, 50, 80, 110, 140 kg/ha), and three varieties (V1, V2, V3). The experiment involved variety and management as qualitative treatment factors and nitrogen fertilizer as a quantitative treatment factor. Overall, there were 45 treatment #'Gomez, K.A., & Gomez, A.A. (1984). Statistical procedures for agricultural research, 2nd edn. New York: Wiley.

Usage

```
data(rice)
```

Format

An object of class data.frame with 135 rows and 8 columns.

---

sorghum *Sorghum data for Example 4*

Description

Milliken & Johnson (1992, p. 429) describe an experiment with four sorghum varieties, in which the leaf area index was assessed in five consecutive weeks starting two weeks after emergence. The experiment was laid out in five randomized complete blocks. The observed data are plotted against week in Figure 5. Milliken, G.A., & Johnson, D.E. (1992). Analysis of messy data. Volume I: Designed experiments. Boca Raton: CRC Press.

Usage

```
data(sorghum)
```

Format

An object of class data.frame with 100 rows and 7 columns.

---

turnip   *Turnip data for Example 5*

Description

Mead (1988, p. 323) describes an experiment on spacing effects with turnips, which was laid out in three complete blocks. Five different seed rates (0.5, 2, 8, 20, 32 lb/acre) were tested in combination with four row widths (4, 8, 16, 32 inches), giving rise to a total of 20 treatments. Turnip yields (in lb per plot) were logarithmically transformed for analysis because this stabilized the variance (Mead, 1988; also see Figure 12). Mead, R. (1988). The design of experiments. Statistical principles for practical application. Cambridge: Cambridge University Press.

Usage

```
data(turnip)
```

Format

An object of class data.frame with 60 rows and 6 columns.

Index