Rice Example: Strip-Plot Analysis

Strip Plot design is also called a split block.

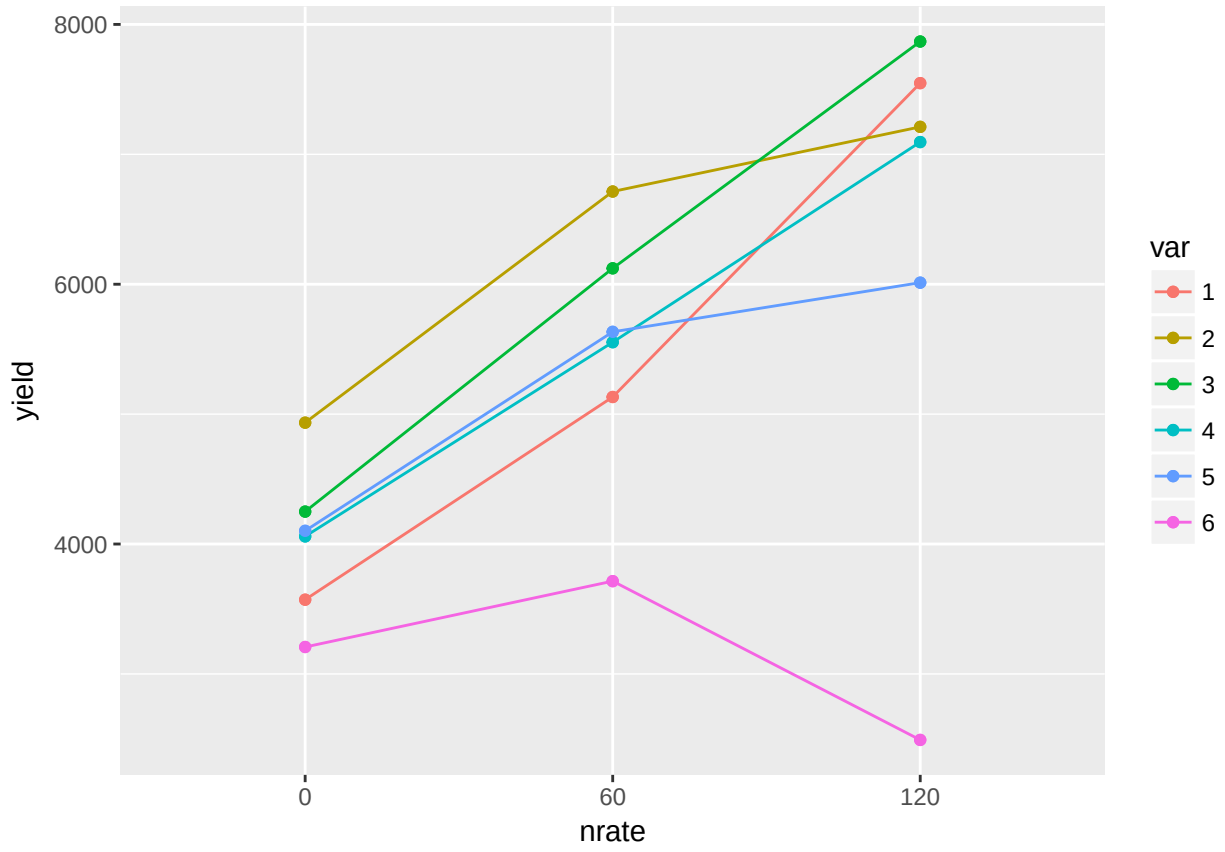
```
library(ggplot2)
library(lme4)
library(lmerTest)
library(pbkrtest)
library(emmeans)
Rice <- read.csv("C:/hess/STAT512/RNotes/Random2/R2_Rice_StripPlot.csv")
str(Rice)
```

```
## 'data.frame':    54 obs. of  4 variables:
##  $ var  : int  1 1 1 1 1 1 1 1 2 ...
##  $ nrate: int  0 0 0 60 60 60 120 120 120 0 ...
##  $ block: int  1 2 3 1 2 3 1 2 3 1 ...
##  $ yield: int  2373 3958 4384 4076 6431 4889 7254 6808 8582 4007 ...
```

```
#Important: Need to define things as.factor!!!
Rice$var <- as.factor(Rice$var)
Rice$nrate <- as.factor(Rice$nrate)
Rice$block <- as.factor(Rice$block)
#Interaction Plot
AvgData <- aggregate(yield ~ var + nrate, data = Rice, mean)
str(AvgData)
```

```
## 'data.frame':    18 obs. of  3 variables:
##  $ var  : Factor w/ 6 levels "1","2","3","4",..: 1 2 3 4 5 6 1 2 3 4 ...
##  $ nrate: Factor w/ 3 levels "0","60","120": 1 1 1 1 1 1 2 2 2 2 ...
##  $ yield: num  3572 4934 4250 4059 4102 ...
```

```
p <- qplot(x = nrate, y = yield, colour = var, group = var, data = AvgData)
p + geom_line() + geom_point()
```




```
Model1 <- lmer(yield ~ var*nrate + (1|block) + (1|block:var) + (1|block:nrate), data = Rice)
summary(Model1)
```

```
## Linear mixed model fit by REML t-tests use Satterthwaite approximations
##   to degrees of freedom [lmerMod]
## Formula:
## yield ~ var * nrate + (1 | block) + (1 | block:var) + (1 | block:nrate)
##    Data: Rice
##
## REML criterion at convergence: 607.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.52993 -0.52843  0.05394  0.51466  1.46903
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  block:var   (Intercept) 360205   600.2
##  block:nrate (Intercept)  55347   235.3
##  block       (Intercept) 154785   393.4
##  Residual                411646   641.6
## Number of obs: 54, groups:  block:var, 18; block:nrate, 9; block, 3
##
## Fixed effects:
##               Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)    3571.67     572.13    17.72   6.243 7.40e-06 ***
```

```
## var2           1362.67     717.33    20.77   1.900   0.0715 .
## var3            678.00     717.33    20.77   0.945   0.3554
## var4            487.33     717.33    20.77   0.679   0.5044
## var5            530.00     717.33    20.77   0.739   0.4683
## var6           -364.33     717.33    20.77  -0.508   0.6169
## nrate60        1560.33     557.97    21.80   2.796   0.0106 *
## nrate120       3976.33     557.97    21.80   7.126 4.01e-07 ***
## var2:nrate60    219.00     740.85    19.10   0.296   0.7707
## var3:nrate60    312.33     740.85    19.10   0.422   0.6780
## var4:nrate60    -65.67     740.85    19.10  -0.089   0.9303
## var5:nrate60    -28.67     740.85    19.10  -0.039   0.9695
## var6:nrate60  -1053.33     740.85    19.10  -1.422   0.1712
## var2:nrate120 -1699.33     740.85    19.10  -2.294   0.0333 *
## var3:nrate120  -357.67     740.85    19.10  -0.483   0.6347
## var4:nrate120  -941.00     740.85    19.10  -1.270   0.2193
## var5:nrate120 -2066.00     740.85    19.10  -2.789   0.0117 *
## var6:nrate120 -4691.67     740.85    19.10  -6.333 4.35e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(Model1, ddf="Kenward-Roger")
```

```
## Analysis of Variance Table of type III  with  Kenward-Roger
## approximation for degrees of freedom
##             Sum Sq  Mean Sq NumDF DenDF F.value    Pr(>F)
## var       15751300  3150260     5    10   7.653 0.0033722 **
## nrate     28048730 14024365     2     4  34.069 0.0030746 **
## var:nrate 23877979  2387798    10    20   5.801 0.0004271 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(Model1, pairwise ~ var|nrate)
```

```
## $emmeans
## nrate = 0:
##  var   emmean       SE    df lower.CL upper.CL
##  1   3571.667 572.1257 17.76 2368.490 4774.843
##  2   4934.333 572.1257 17.76 3731.157 6137.510
##  3   4249.667 572.1257 17.76 3046.490 5452.843
##  4   4059.000 572.1257 17.76 2855.823 5262.177
##  5   4101.667 572.1257 17.76 2898.490 5304.843
##  6   3207.333 572.1257 17.76 2004.157 4410.510
##
## nrate = 60:
##  var   emmean       SE    df lower.CL upper.CL
##  1   5132.000 572.1257 17.76 3928.823 6335.177
##  2   6713.667 572.1257 17.76 5510.490 7916.843
##  3   6122.333 572.1257 17.76 4919.157 7325.510
##  4   5553.667 572.1257 17.76 4350.490 6756.843
##  5   5633.333 572.1257 17.76 4430.157 6836.510
##  6   3714.333 572.1257 17.76 2511.157 4917.510
##
## nrate = 120:
##  var   emmean       SE    df lower.CL upper.CL
##  1   7548.000 572.1257 17.76 6344.823 8751.177
```

```
##  2   7211.333 572.1257 17.76 6008.157 8414.510
##  3   7868.333 572.1257 17.76 6665.157 9071.510
##  4   7094.333 572.1257 17.76 5891.157 8297.510
##  5   6012.000 572.1257 17.76 4808.823 7215.177
##  6   2492.000 572.1257 17.76 1288.823 3695.177
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
## nrate = 0:
##  contrast    estimate       SE   df t.ratio p.value
##  1 - 2    -1362.66667 717.3336 20.9  -1.900  0.4295
##  1 - 3     -678.00000 717.3336 20.9  -0.945  0.9298
##  1 - 4     -487.33333 717.3336 20.9  -0.679  0.9824
##  1 - 5     -530.00000 717.3336 20.9  -0.739  0.9746
##  1 - 6      364.33333 717.3336 20.9   0.508  0.9953
##  2 - 3      684.66667 717.3336 20.9   0.954  0.9271
##  2 - 4      875.33333 717.3336 20.9   1.220  0.8222
##  2 - 5      832.66667 717.3336 20.9   1.161  0.8500
##  2 - 6     1727.00000 717.3336 20.9   2.408  0.1987
##  3 - 4      190.66667 717.3336 20.9   0.266  0.9998
##  3 - 5      148.00000 717.3336 20.9   0.206  0.9999
##  3 - 6     1042.33333 717.3336 20.9   1.453  0.6958
##  4 - 5      -42.66667 717.3336 20.9  -0.059  1.0000
##  4 - 6      851.66667 717.3336 20.9   1.187  0.8379
##  5 - 6      894.33333 717.3336 20.9   1.247  0.8092
##
## nrate = 60:
##  contrast    estimate       SE   df t.ratio p.value
##  1 - 2    -1581.66667 717.3336 20.9  -2.205  0.2769
##  1 - 3     -990.33333 717.3336 20.9  -1.381  0.7377
##  1 - 4     -421.66667 717.3336 20.9  -0.588  0.9908
##  1 - 5     -501.33333 717.3336 20.9  -0.699  0.9800
##  1 - 6     1417.66667 717.3336 20.9   1.976  0.3876
##  2 - 3      591.33333 717.3336 20.9   0.824  0.9596
##  2 - 4     1160.00000 717.3336 20.9   1.617  0.5970
##  2 - 5     1080.33333 717.3336 20.9   1.506  0.6644
##  2 - 6     2999.33333 717.3336 20.9   4.181  0.0049
##  3 - 4      568.66667 717.3336 20.9   0.793  0.9657
##  3 - 5      489.00000 717.3336 20.9   0.682  0.9821
##  3 - 6     2408.00000 717.3336 20.9   3.357  0.0310
##  4 - 5      -79.66667 717.3336 20.9  -0.111  1.0000
##  4 - 6     1839.33333 717.3336 20.9   2.564  0.1508
##  5 - 6     1919.00000 717.3336 20.9   2.675  0.1229
##
## nrate = 120:
##  contrast    estimate       SE   df t.ratio p.value
##  1 - 2      336.66667 717.3336 20.9   0.469  0.9967
##  1 - 3     -320.33333 717.3336 20.9  -0.447  0.9974
##  1 - 4      453.66667 717.3336 20.9   0.632  0.9872
##  1 - 5     1536.00000 717.3336 20.9   2.141  0.3054
##  1 - 6     5056.00000 717.3336 20.9   7.048  <.0001
##  2 - 3     -657.00000 717.3336 20.9  -0.916  0.9380
```

```
##  2 - 4      117.00000 717.3336 20.9   0.163  1.0000
##  2 - 5     1199.33333 717.3336 20.9   1.672  0.5636
##  2 - 6     4719.33333 717.3336 20.9   6.579  <.0001
##  3 - 4      774.00000 717.3336 20.9   1.079  0.8842
##  3 - 5     1856.33333 717.3336 20.9   2.588  0.1445
##  3 - 6     5376.33333 717.3336 20.9   7.495  <.0001
##  4 - 5     1082.33333 717.3336 20.9   1.509  0.6627
##  4 - 6     4602.33333 717.3336 20.9   6.416  <.0001
##  5 - 6     3520.00000 717.3336 20.9   4.907  0.0009
##
## P value adjustment: tukey method for comparing a family of 6 estimates
```

```
emmeans(Model1, pairwise ~ nrate|var)
```

```
## $emmeans
## var = 1:
##  nrate   emmean       SE    df lower.CL upper.CL
##  0     3571.667 572.1257 17.76 2368.490 4774.843
##  60    5132.000 572.1257 17.76 3928.823 6335.177
##  120   7548.000 572.1257 17.76 6344.823 8751.177
##
## var = 2:
##  nrate   emmean       SE    df lower.CL upper.CL
##  0     4934.333 572.1257 17.76 3731.157 6137.510
##  60    6713.667 572.1257 17.76 5510.490 7916.843
##  120   7211.333 572.1257 17.76 6008.157 8414.510
##
## var = 3:
##  nrate   emmean       SE    df lower.CL upper.CL
##  0     4249.667 572.1257 17.76 3046.490 5452.843
##  60    6122.333 572.1257 17.76 4919.157 7325.510
##  120   7868.333 572.1257 17.76 6665.157 9071.510
##
## var = 4:
##  nrate   emmean       SE    df lower.CL upper.CL
##  0     4059.000 572.1257 17.76 2855.823 5262.177
##  60    5553.667 572.1257 17.76 4350.490 6756.843
##  120   7094.333 572.1257 17.76 5891.157 8297.510
##
## var = 5:
##  nrate   emmean       SE    df lower.CL upper.CL
##  0     4101.667 572.1257 17.76 2898.490 5304.843
##  60    5633.333 572.1257 17.76 4430.157 6836.510
##  120   6012.000 572.1257 17.76 4808.823 7215.177
##
## var = 6:
##  nrate   emmean       SE    df lower.CL upper.CL
##  0     3207.333 572.1257 17.76 2004.157 4410.510
##  60    3714.333 572.1257 17.76 2511.157 4917.510
##  120   2492.000 572.1257 17.76 1288.823 3695.177
##
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
##
## $contrasts
```

```
## var = 1:
##  contrast   estimate       SE    df t.ratio p.value
##  0 - 60   -1560.3333 557.9682 22.43  -2.796  0.0270
##  0 - 120  -3976.3333 557.9682 22.43  -7.126  <.0001
##  60 - 120 -2416.0000 557.9682 22.43  -4.330  0.0007
##
## var = 2:
##  contrast   estimate       SE    df t.ratio p.value
##  0 - 60   -1779.3333 557.9682 22.43  -3.189  0.0112
##  0 - 120  -2277.0000 557.9682 22.43  -4.081  0.0013
##  60 - 120  -497.6667 557.9682 22.43  -0.892  0.6508
##
## var = 3:
##  contrast   estimate       SE    df t.ratio p.value
##  0 - 60   -1872.6667 557.9682 22.43  -3.356  0.0076
##  0 - 120  -3618.6667 557.9682 22.43  -6.485  <.0001
##  60 - 120 -1746.0000 557.9682 22.43  -3.129  0.0128
##
## var = 4:
##  contrast   estimate       SE    df t.ratio p.value
##  0 - 60   -1494.6667 557.9682 22.43  -2.679  0.0349
##  0 - 120  -3035.3333 557.9682 22.43  -5.440  <.0001
##  60 - 120 -1540.6667 557.9682 22.43  -2.761  0.0292
##
## var = 5:
##  contrast   estimate       SE    df t.ratio p.value
##  0 - 60   -1531.6667 557.9682 22.43  -2.745  0.0302
##  0 - 120  -1910.3333 557.9682 22.43  -3.424  0.0065
##  60 - 120  -378.6667 557.9682 22.43  -0.679  0.7781
##
## var = 6:
##  contrast   estimate       SE    df t.ratio p.value
##  0 - 60    -507.0000 557.9682 22.43  -0.909  0.6406
##  0 - 120    715.3333 557.9682 22.43   1.282  0.4196
##  60 - 120  1222.3333 557.9682 22.43   2.191  0.0947
##
## P value adjustment: tukey method for comparing a family of 3 estimates
```