# Case Study : A RCBD trial for wheat - a simple analysis using fixed models

Prof L. Gentzbittel Skoltech, Digital Agriculture Laboratory *

Prof C. Ben, Skoltech, Digital Agriculture Laboratory †

April, 2nd 2021 - Skoltech

## CASE STUDY PRESENTATION

Seven winter wheat cultivars were assessed for yield in a RCBD with four blocks

## PREPARATION OF THE WORKING INTERFACE IN R

```
### I. Set working directory
# On RStudio: tab 'Session'-> Set Working Directory -> Choose Directory.
# Choose the directory containing the datafile and the associated R script.

### II. Installation R packages needed for the analysis on RStudio:
# Click on the 'Packages' tab in the bottom-right window of R Studio interface->'Install Packages'
# Comment #1: R package installation requires a connection to internet
# Comment #2: Once packages have been installed, no need to re-install
# them again when you close-open again RStudio.

### III. Initialisation of the working space
# To erase all graphs
graphics.off()
# To erase objects from the working space - Clean up of the memory
rm(list = ls())
# use of the constraint 'set-to-zero' for ANOVAs ## will see later in this script
options(contrasts=c('contr.treatment','contr.poly'))
# we can also use 'contr.sum' for a 'sum-to-zero' constraint
```

## LOADING REQUIRED METHODS FOR ANALYSIS

```
## Loading of the R packages needed for the analysis.
library(ggplot2) # Needed for some graphs (e.g. bwplots)
library(gridExtra)
library(agricolae) # For multiple mean comparisons
library(multcomp)  # for alternative multiple mean comparisons
library(multcompView)
library(emmeans)   # for alternative multiple mean comparisons
library(car)       # for Levene's test
library(openxlsx)   ## to import Excel files
```

*l.gentzbittel@skoltech.ru

†c.ben@skoltech.ru

# STARTING THE ANALYSIS

```
###################
# Data import in R
###################

WheatYield  <- read.xlsx("02_Wheat7Var4Blocks.xlsx", sheet = 1)

WheatYield
```

```
##     Genotype   Block     Yield
## 1   Variety6  Block1  74.35477
## 2   Variety3  Block1  75.88057
## 3   Variety7  Block1  74.30482
## 4   Variety1  Block1  74.21797
## 5   Variety2  Block1  78.17027
## 6   Variety4  Block1  71.18263
## 7   Variety5  Block1  76.11654
## 8   Variety6  Block2  77.35632
## 9   Variety3  Block2  79.57677
## 10  Variety7  Block2  75.53062
## 11  Variety1  Block2  78.75834
## 12  Variety2  Block2  81.70394
## 13  Variety4  Block2  77.11604
## 14  Variety5  Block2  76.73514
## 15  Variety6  Block3  72.35833
## 16  Variety3  Block3  68.24946
## 17  Variety7  Block3  71.01684
## 18  Variety1  Block3  71.74571
## 19  Variety2  Block3  71.73163
## 20  Variety4  Block3  72.35958
## 21  Variety5  Block3  71.41571
## 22  Variety6  Block4  75.51941
## 23  Variety3  Block4  75.99345
## 24  Variety7  Block4  72.62122
## 25  Variety1  Block4  73.98632
## 26  Variety2  Block4  78.70578
## 27  Variety4  Block4  72.62981
## 28  Variety5  Block4  74.81868
```

```
str(WheatYield )  ## check if all columns are of the expected type: numeric, or factors, ...
```

```
## 'data.frame':    28 obs. of  3 variables:
##  $ Genotype: chr  " Variety6" " Variety3" " Variety7" " Variety1" ...
##  $ Block   : chr  " Block1" " Block1" " Block1" " Block1" ...
##  $ Yield   : chr  " 74.35477" " 75.88057" " 74.30482" " 74.21797" ...
```

```
## need to convert characters data into factor data -- a weakness of read.xlsx()
WheatYield$Genotype  <- as.factor(WheatYield$Genotype)
WheatYield$Block  <- as.factor(WheatYield$Block)
WheatYield$Yield <- as.numeric(WheatYield$Yield)   ## may not be required on your computer.
## also a weakness of read.xlsx()
str(WheatYield )
```

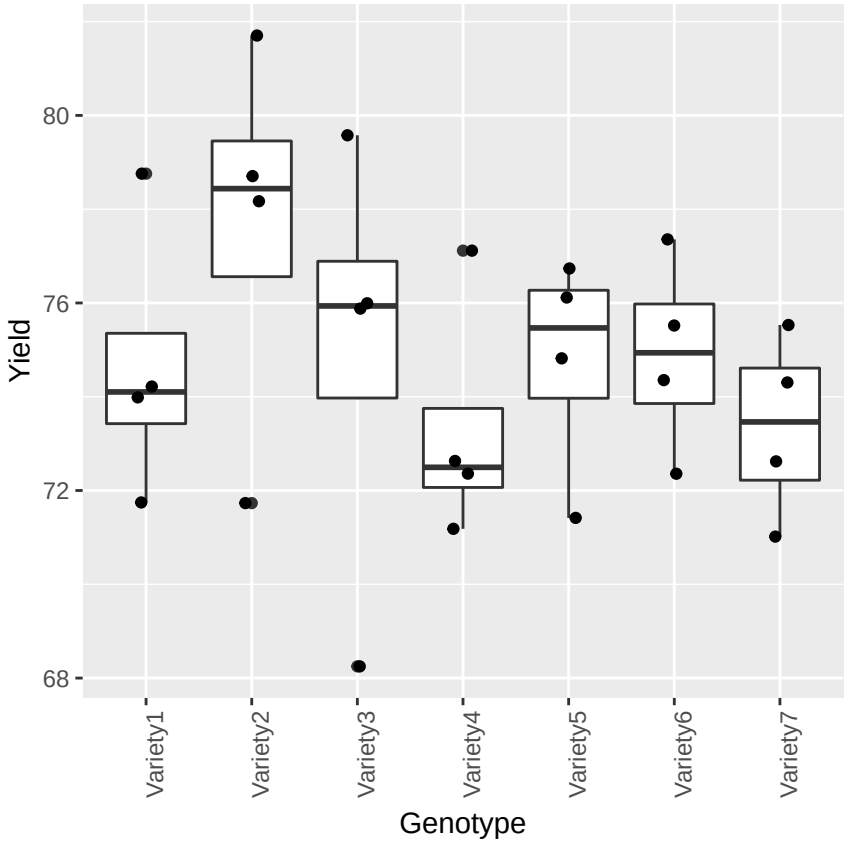
```
## 'data.frame':    28 obs. of  3 variables:
##  $ Genotype: Factor w/ 7 levels " Variety1"," Variety2",..: 6 3 7 1 2 4 5 6 3 7 ...
##  $ Block   : Factor w/ 4 levels " Block1"," Block2",..: 1 1 1 1 1 1 1 2 2 2 ...
##  $ Yield   : num  74.4 75.9 74.3 74.2 78.2 ...
```

```
attach(WheatYield )# It avoids having to specify the name of the dataframe in R commands
## i.e. it is no more useful to write Dataframe$factor or Dataframe$variable
```

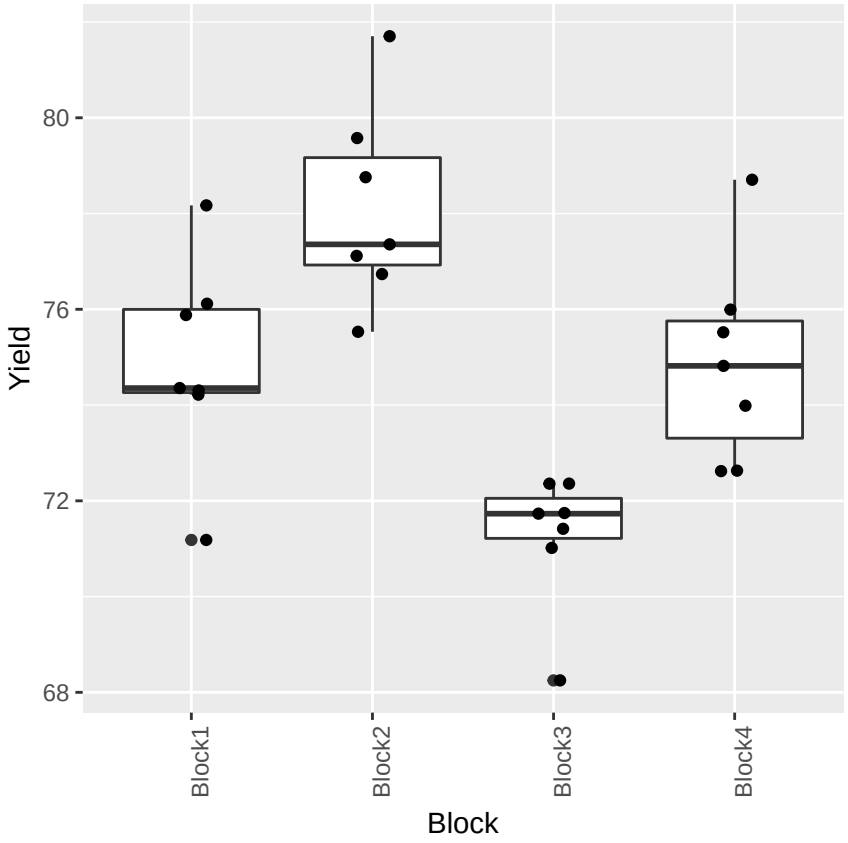
```r
##Check for balanced dataset :
table(Genotype,Block)
```

```
##            Block
## Genotype     Block1  Block2  Block3  Block4
##    Variety1       1       1       1       1
##    Variety2       1       1       1       1
##    Variety3       1       1       1       1
##    Variety4       1       1       1       1
##    Variety5       1       1       1       1
##    Variety6       1       1       1       1
##    Variety7       1       1       1       1
```

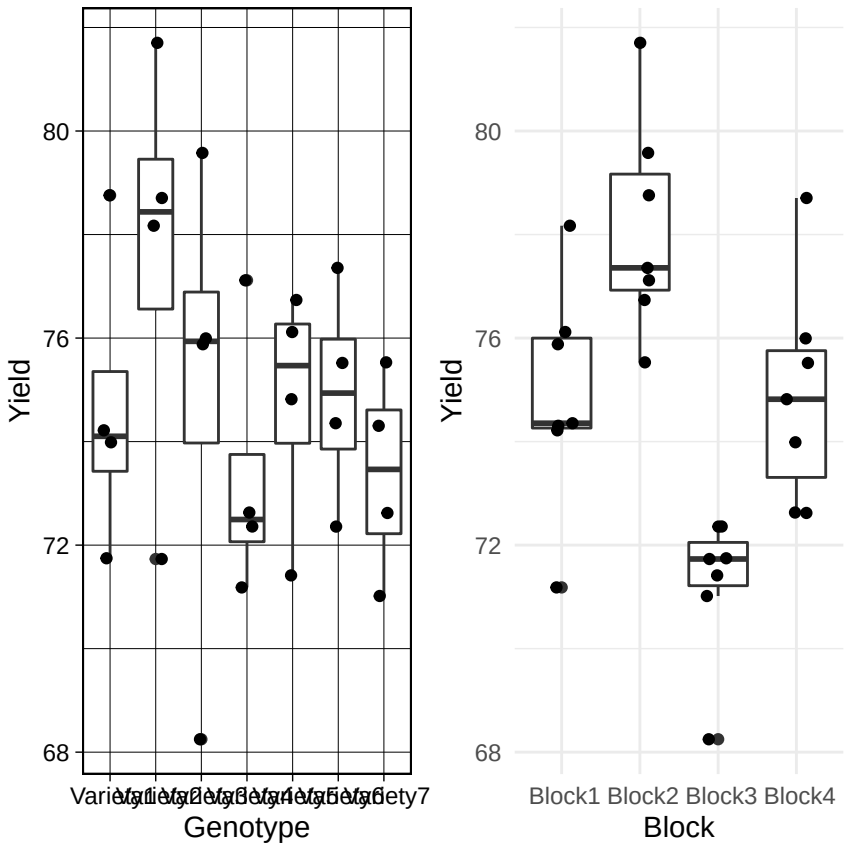
```r
## of course the phenotypic value is expected to be the mean of a microplot
## or of randomly chosen plants ; not one plant !
## if several plants per plot ->  better work with the mean per plot to reduce variance
## variance of mean = variance of raw data / nbr of plants

#########################
# CHECK POINT !
# Identify the factors,
# propose a practical set up of this design, in the field or in greenhouse
# which practical data (or informations) are lacking ?
#
#########################

#########################
# Graphic visualizations
#########################

## Boxplots to reveal the distribution and variance of the measured traits
## depending on the different factors of interest

#Individual graphs
x11()
ggplot(WheatYield) +
    aes(x = Genotype, y = Yield) +
    geom_boxplot() + geom_jitter(width = 0.1) + theme(axis.text.x = element_text(angle = 90))
```



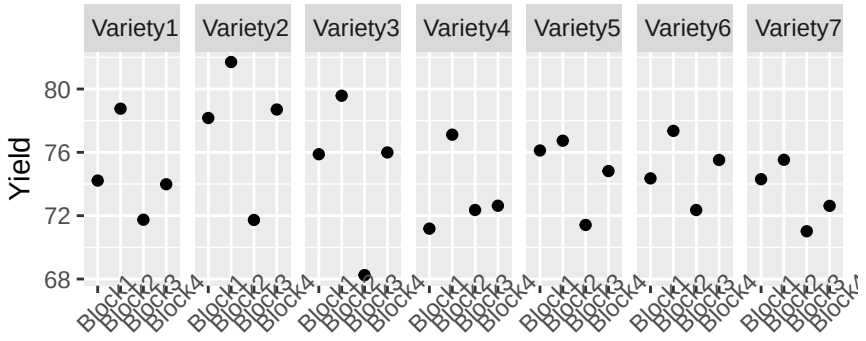

```
x11()
ggplot(WheatYield) +
  aes(x = Block, y = Yield) +
  geom_boxplot() + geom_jitter(width = 0.10)+
  theme(axis.text.x = element_text(angle = 90))
```

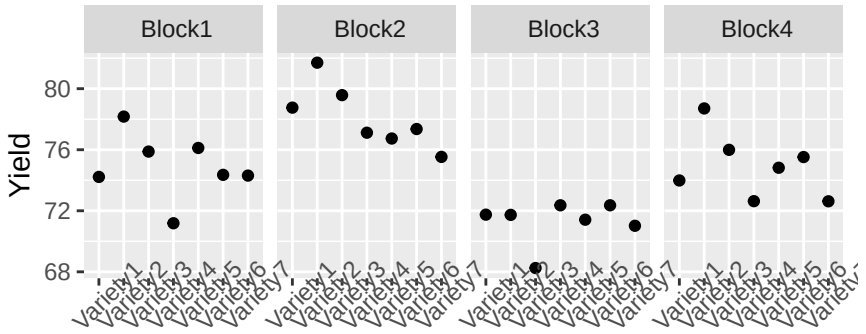



```
#2 graphs on the same window. Put each graphic in an object
graf1 <- ggplot(WheatYield) +
    aes(x = Genotype, y = Yield) +
    geom_boxplot() +
    geom_jitter(width = 0.15) +
    theme(axis.text.x = element_text(angle = 90)) +
    theme_linedraw()  ## to evidence quick customisation of figures

graf2 <- ggplot(WheatYield) +
     aes(x = Block, y = Yield) +
    geom_boxplot() +
    geom_jitter(width = 0.15)+
    theme(axis.text.x = element_text(angle = 90)) +
    theme_minimal()  ## to evidence quick customisation of figures

x11()
grid.arrange(graf1, graf2, ncol = 2, nrow = 1)  ## display the graphical objects
```




```
#xyplots if you need to see any individual performances
graf3 <- ggplot(WheatYield) +
    aes(x = Block, y = Yield) +
    geom_point() +
    theme(axis.text.x = element_text(angle = 45)) +
    facet_grid( . ~ Genotype)


graf4 <- ggplot(WheatYield) +
    aes(x = Genotype, y = Yield) +
    geom_point() +
    theme(axis.text.x = element_text(angle = 45)) +
    facet_grid( . ~ Block)

x11()
grid.arrange(graf3, graf4, ncol = 1, nrow = 2)
```




```
################  ANOVA
model1 <- aov( Yield ~ Block + Genotype )
summary(model1)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## Block        3 164.18   54.73   21.91 3.11e-06 ***
## Genotype     6  47.96    7.99    3.20   0.0256 *
## Residuals   18  44.96    2.50
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## conclusions ?


#Test for ANOVA pre-requisites
#Normality of ANOVA residuals
shapiro.test(residuals(model1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model1)
## W = 0.94247, p-value = 0.1277
```

```
#Variance homogeneity of ANOVA residuals
leveneTest(Yield, Genotype)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  6  0.2412 0.9576
##       21
```

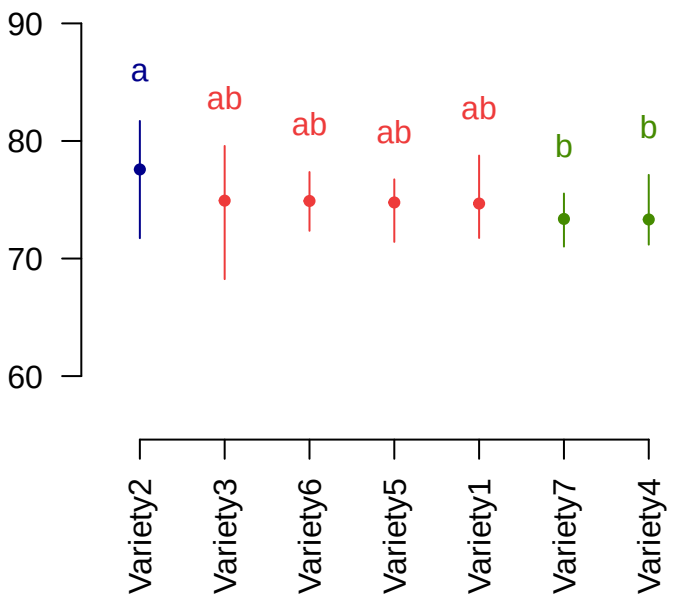
```r
leveneTest(Yield, Block)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  3  0.4483 0.7208
##       24
```

```r
## because only one residual per combination Block x Genotype


#### Multiple mean comparisons - Tukey HSD
print(HSD.test(model1,"Genotype"))
```

```
## $statistics
##    MSerror Df     Mean       CV     MSD
##   2.498045 18 74.79131 2.113241 3.69299
##
## $parameters
##    test   name.t ntr StudentizedRange alpha
##   Tukey Genotype   7         4.673132  0.05
##
## $means
##              Yield      std r      Min      Max      Q25      Q50      Q75
##  Variety1 74.67709 2.940382 4 71.74571 78.75834 73.42617 74.10215 75.35306
##  Variety2 77.57791 4.196272 4 71.73163 81.70394 76.56061 78.43803 79.45532
##  Variety3 74.92506 4.769923 4 68.24946 79.57677 73.97279 75.93701 76.88928
##  Variety4 73.32201 2.606212 4 71.18263 77.11604 72.06534 72.49470 73.75137
##  Variety5 74.77152 2.375470 4 71.41571 76.73514 73.96794 75.46761 76.27119
##  Variety6 74.89721 2.095591 4 72.35833 77.35632 73.85566 74.93709 75.97864
##  Variety7 73.36838 1.969788 4 71.01684 75.53062 72.22012 73.46302 74.61127
##
## $comparison
## NULL
##
## $groups
##              Yield groups
##  Variety2 77.57791      a
##  Variety3 74.92506     ab
##  Variety6 74.89721     ab
##  Variety5 74.77152     ab
##  Variety1 74.67709     ab
##  Variety7 73.36838      b
##  Variety4 73.32201      b
##
## attr(,"class")
## [1] "group"
```

```r
x11()
plot(HSD.test(model1,"Genotype"), las = 2)
```

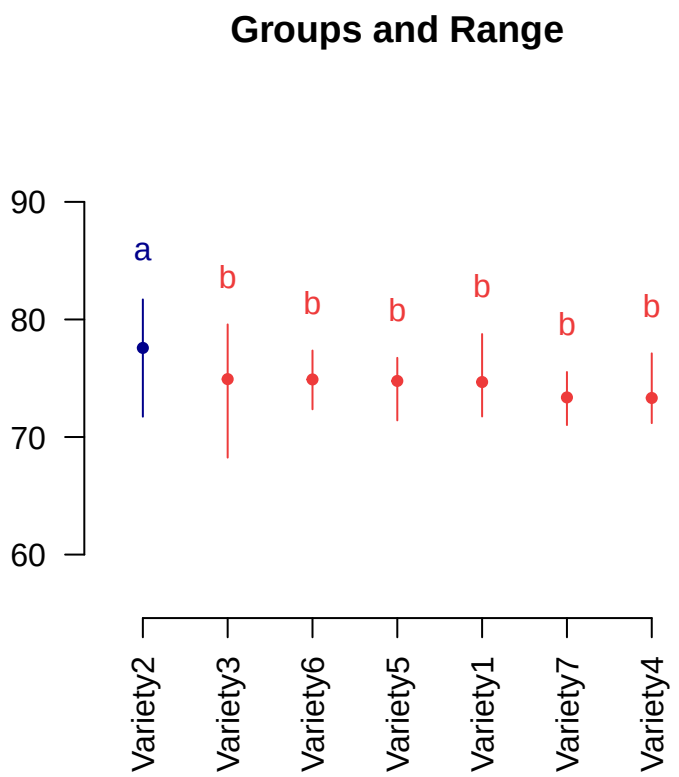



```
## this graph is done with "base graphics" but not with "ggplot graphics". Syntax and options are differen

## Multiple mean comparisons - Newman-Keuls
print(SNK.test(model1,"Genotype"))
```

```
## $statistics
##    MSerror Df     Mean       CV
##   2.498045 18 74.79131 2.113241
##
## $parameters
##   test   name.t ntr alpha
##    SNK Genotype   7  0.05
##
## $snk
##      Table CriticalRange
## 2 2.971152      2.347984
## 3 3.609304      2.852289
## 4 3.996978      3.158653
## 5 4.276293      3.379384
## 6 4.494420      3.551761
## 7 4.673132      3.692990
##
## $means
##              Yield      std r      Min      Max      Q25      Q50      Q75
##  Variety1 74.67709 2.940382 4 71.74571 78.75834 73.42617 74.10215 75.35306
##  Variety2 77.57791 4.196272 4 71.73163 81.70394 76.56061 78.43803 79.45532
##  Variety3 74.92506 4.769923 4 68.24946 79.57677 73.97279 75.93701 76.88928
##  Variety4 73.32201 2.606212 4 71.18263 77.11604 72.06534 72.49470 73.75137
##  Variety5 74.77152 2.375470 4 71.41571 76.73514 73.96794 75.46761 76.27119
##  Variety6 74.89721 2.095591 4 72.35833 77.35632 73.85566 74.93709 75.97864
##  Variety7 73.36838 1.969788 4 71.01684 75.53062 72.22012 73.46302 74.61127
```

```
##
## $comparison
## NULL
##
## $groups
##              Yield groups
##  Variety2 77.57791      a
##  Variety3 74.92506      b
##  Variety6 74.89721      b
##  Variety5 74.77152      b
##  Variety1 74.67709      b
##  Variety7 73.36838      b
##  Variety4 73.32201      b
##
## attr(,"class")
## [1] "group"
```

```
x11()
plot(SNK.test(model1,"Genotype"), las = 2)
```




```
## this graph is done with "base graphics" but not with "ggplot graphics". Syntax and options are differen


## Adjusted means (because the design is balanced, adjusted means = means from data)
AdjustMoys1 <- emmeans(model1,
                       pairwise ~ Genotype,
                       adjust = "tukey")
## Options are "tukey", "scheffe", "sidak", "bonferroni", "dunnettx", "mvt", and "none"
AdjustMoys1
```

```
## $emmeans
##  Genotype  emmean   SE df lower.CL upper.CL
```

```
##   Variety1   74.7 0.79 18     73.0     76.3
##   Variety2   77.6 0.79 18     75.9     79.2
##   Variety3   74.9 0.79 18     73.3     76.6
##   Variety4   73.3 0.79 18     71.7     75.0
##   Variety5   74.8 0.79 18     73.1     76.4
##   Variety6   74.9 0.79 18     73.2     76.6
##   Variety7   73.4 0.79 18     71.7     75.0
##
## Results are averaged over the levels of: Block
## Confidence level used: 0.95
##
## $contrasts
##  contrast              estimate   SE df t.ratio p.value
##   Variety1 -  Variety2  -2.9008 1.12 18 -2.596  0.1848
##   Variety1 -  Variety3  -0.2480 1.12 18 -0.222  1.0000
##   Variety1 -  Variety4   1.3551 1.12 18  1.212  0.8804
##   Variety1 -  Variety5  -0.0944 1.12 18 -0.084  1.0000
##   Variety1 -  Variety6  -0.2201 1.12 18 -0.197  1.0000
##   Variety1 -  Variety7   1.3087 1.12 18  1.171  0.8961
##   Variety2 -  Variety3   2.6528 1.12 18  2.374  0.2645
##   Variety2 -  Variety4   4.2559 1.12 18  3.808  0.0181
##   Variety2 -  Variety5   2.8064 1.12 18  2.511  0.2126
##   Variety2 -  Variety6   2.6807 1.12 18  2.399  0.2545
##   Variety2 -  Variety7   4.2095 1.12 18  3.767  0.0197
##   Variety3 -  Variety4   1.6030 1.12 18  1.434  0.7771
##   Variety3 -  Variety5   0.1535 1.12 18  0.137  1.0000
##   Variety3 -  Variety6   0.0279 1.12 18  0.025  1.0000
##   Variety3 -  Variety7   1.5567 1.12 18  1.393  0.7986
##   Variety4 -  Variety5  -1.4495 1.12 18 -1.297  0.8447
##   Variety4 -  Variety6  -1.5752 1.12 18 -1.409  0.7901
##   Variety4 -  Variety7  -0.0464 1.12 18 -0.041  1.0000
##   Variety5 -  Variety6  -0.1257 1.12 18 -0.112  1.0000
##   Variety5 -  Variety7   1.4031 1.12 18  1.256  0.8628
##   Variety6 -  Variety7   1.5288 1.12 18  1.368  0.8111
##
## Results are averaged over the levels of: Block
## P value adjustment: tukey method for comparing a family of 7 estimates
```

```
## Multiple comparisons. another way to compute them
## Note that the adjust= option should also be applied to the cld function if a compact letter display is
(CompMoys1 <- multcomp::cld(AdjustMoys1[[1]],
    alpha = 0.05 ,
    Letters = letters ,
    adjust = "tukey"
    )
)
```

```
##  Genotype  emmean   SE df lower.CL upper.CL .group
##   Variety4   73.3 0.79 18     70.9     75.7  a
##   Variety7   73.4 0.79 18     71.0     75.8  a
##   Variety1   74.7 0.79 18     72.3     77.1  ab
##   Variety5   74.8 0.79 18     72.4     77.2  ab
##   Variety6   74.9 0.79 18     72.5     77.3  ab
##   Variety3   74.9 0.79 18     72.5     77.3  ab
##   Variety2   77.6 0.79 18     75.2     80.0   b
##
## Results are averaged over the levels of: Block
## Confidence level used: 0.95
## Conf-level adjustment: sidak method for 7 estimates
## P value adjustment: tukey method for comparing a family of 7 estimates
```

```
## significance level used: alpha = 0.05
```

```
########################
# CHECK POINT !
########################
#
# what if we did not define blocks ?
#
########################

ReducedModel <- aov( Yield ~ Genotype)

summary(ReducedModel)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Genotype     6  47.96   7.994   0.803  0.579
## Residuals   21 209.14   9.959
```

```
## Multiple mean comparisons - Newman-Keuls
print(SNK.test(ReducedModel, "Genotype"))
```

```
## $statistics
##    MSerror Df     Mean       CV
##   9.959068 21 74.79131 4.219473
##
## $parameters
##   test   name.t ntr alpha
##    SNK Genotype   7  0.05
##
## $snk
##      Table CriticalRange
## 2 2.941018      4.640631
## 3 3.564625      5.624620
## 4 3.941878      6.219887
## 5 4.212995      6.647683
## 6 4.424353      6.981185
## 7 4.597302      7.254080
##
## $means
##              Yield      std r      Min      Max      Q25      Q50      Q75
##  Variety1 74.67709 2.940382 4 71.74571 78.75834 73.42617 74.10215 75.35306
##  Variety2 77.57791 4.196272 4 71.73163 81.70394 76.56061 78.43803 79.45532
##  Variety3 74.92506 4.769923 4 68.24946 79.57677 73.97279 75.93701 76.88928
##  Variety4 73.32201 2.606212 4 71.18263 77.11604 72.06534 72.49470 73.75137
##  Variety5 74.77152 2.375470 4 71.41571 76.73514 73.96794 75.46761 76.27119
##  Variety6 74.89721 2.095591 4 72.35833 77.35632 73.85566 74.93709 75.97864
##  Variety7 73.36838 1.969788 4 71.01684 75.53062 72.22012 73.46302 74.61127
##
## $comparison
## NULL
##
## $groups
##              Yield groups
##  Variety2 77.57791      a
##  Variety3 74.92506      a
##  Variety6 74.89721      a
##  Variety5 74.77152      a
##  Variety1 74.67709      a
##  Variety7 73.36838      a
##  Variety4 73.32201      a
```

```
##
## attr(,"class")
## [1] "group"
```

```
## alternative computation
AdjustMoys4 <- emmeans(ReducedModel,
                    pairwise ~ Genotype,
                    adjust = "tukey")
AdjustMoys4
```

```
## $emmeans
##  Genotype  emmean   SE df lower.CL upper.CL
##   Variety1   74.7 1.58 21     71.4     78.0
##   Variety2   77.6 1.58 21     74.3     80.9
##   Variety3   74.9 1.58 21     71.6     78.2
##   Variety4   73.3 1.58 21     70.0     76.6
##   Variety5   74.8 1.58 21     71.5     78.1
##   Variety6   74.9 1.58 21     71.6     78.2
##   Variety7   73.4 1.58 21     70.1     76.6
##
## Confidence level used: 0.95
##
## $contrasts
##  contrast              estimate   SE df t.ratio p.value
##   Variety1 -  Variety2  -2.9008 2.23 21 -1.300  0.8444
##   Variety1 -  Variety3  -0.2480 2.23 21 -0.111  1.0000
##   Variety1 -  Variety4   1.3551 2.23 21  0.607  0.9959
##   Variety1 -  Variety5  -0.0944 2.23 21 -0.042  1.0000
##   Variety1 -  Variety6  -0.2201 2.23 21 -0.099  1.0000
##   Variety1 -  Variety7   1.3087 2.23 21  0.586  0.9966
##   Variety2 -  Variety3   2.6528 2.23 21  1.189  0.8907
##   Variety2 -  Variety4   4.2559 2.23 21  1.907  0.4974
##   Variety2 -  Variety5   2.8064 2.23 21  1.258  0.8631
##   Variety2 -  Variety6   2.6807 2.23 21  1.201  0.8859
##   Variety2 -  Variety7   4.2095 2.23 21  1.886  0.5098
##   Variety3 -  Variety4   1.6030 2.23 21  0.718  0.9899
##   Variety3 -  Variety5   0.1535 2.23 21  0.069  1.0000
##   Variety3 -  Variety6   0.0279 2.23 21  0.012  1.0000
##   Variety3 -  Variety7   1.5567 2.23 21  0.698  0.9913
##   Variety4 -  Variety5  -1.4495 2.23 21 -0.650  0.9940
##   Variety4 -  Variety6  -1.5752 2.23 21 -0.706  0.9907
##   Variety4 -  Variety7  -0.0464 2.23 21 -0.021  1.0000
##   Variety5 -  Variety6  -0.1257 2.23 21 -0.056  1.0000
##   Variety5 -  Variety7   1.4031 2.23 21  0.629  0.9950
##   Variety6 -  Variety7   1.5288 2.23 21  0.685  0.9921
##
## P value adjustment: tukey method for comparing a family of 7 estimates
```

```
(CompMoys4 <- cld(AdjustMoys4[[1]],
    alpha = 0.05 ,
    Letters = letters ,
    adjust = "tukey"
    )
)
```

```
##  Genotype  emmean   SE df lower.CL upper.CL .group
##   Variety4   73.3 1.58 21     68.6     78.0  a
##   Variety7   73.4 1.58 21     68.7     78.1  a
##   Variety1   74.7 1.58 21     70.0     79.4  a
##   Variety5   74.8 1.58 21     70.1     79.5  a
```

```
##   Variety6   74.9 1.58 21     70.2     79.6  a
##   Variety3   74.9 1.58 21     70.2     79.6  a
##   Variety2   77.6 1.58 21     72.9     82.3  a
##
## Confidence level used: 0.95
## Conf-level adjustment: sidak method for 7 estimates
## P value adjustment: tukey method for comparing a family of 7 estimates
## significance level used: alpha = 0.05
```

```
# compare to CompMoys1
## we can follow up by doing formal tests on the power of the experiment.
```

# A POINT OF THEORY : HOW ARE *really* COMPUTED ANOVA TABLES

```
## aov() is fitting a GLM to test factor effects
## To explore the **reality** of computing ANOVA tables
## We will explore this point in "regular course", for unbalanced data

(SST <- t(Yield) %*% Yield)  ## Total sum-of-squares
```

```
##          [,1]
## [1,] 156881.8
```

```
M0 <- aov( Yield ~ 1)  ## adjusting a mean
summary(M0)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Residuals   27  257.1   9.522
```

```
M1 <- aov( Yield ~ 1 + Genotype)
## eq to aov( Yield ~ Genotype) allows to show we are fitting the mean AND Genotype
summary(M1)
```

```
##             Df Sum Sq Mean Sq F value Pr(>F)
## Genotype     6  47.96   7.994   0.803  0.579
## Residuals   21 209.14   9.959
```

```
M2 <- aov( Yield ~ 1 + Block)
## eq to aov( Yield ~ Block) allows to shom we are fitting the mean AND Block
summary(M2)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## Block        3 164.18   54.73   14.13 1.64e-05 ***
## Residuals   24  92.93    3.87
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Mfinal <- aov( Yield ~ 1 + Genotype + Block)
summary(Mfinal)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## Genotype     6  47.96    7.99    3.20   0.0256 *
## Block        3 164.18   54.73   21.91 3.11e-06 ***
## Residuals   18  44.96    2.50
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```