LSD Vs. RBD & CRD (relative efficiencies)

Pranav Gopalkrishna, 1940223

2021-11-02

# DATA SET

This data set contains crop yield data, wherein crops are grown in 3 different plots, A, B and C, and there are three levels of soil density (low, medium, high). These are the blocking factors.

On top of this, we also have fertilizer types being applied to each plot of crops. Every plot and soil density category have been treated by every fertilizer type once.  
setwd("~/Documents/Study/computerScience/programming/r/data/")  
myData = read.csv("cropYieldTruncated.csv")[-3]  
head(myData)

## yield block density fertilizer  
## 1 90 A low N  
## 2 95 A low P  
## 3 107 A low NP  
## 4 92 A medium N  
## 5 89 A medium P  
## 6 92 A medium NP

## DATA USED

t = myData$fertilizer   
y = myData$yield

# FUNCTIONS USED

The following functions and values are used to facilitate the comparisons of efficiencies between the different experimental designs.

# Correction factor  
cf = sum(y)^2 / length(y)

# (Treatment level mean square)  
lms = function(level, regressor, response)  
{  
 sum = 0  
 n = 0  
 for(i in c(1:(length(regressor))))  
 {  
 if(regressor[i] == level)  
 {  
 sum = sum + response[i]  
 n = n + 1  
 }  
 }  
 return(sum^2 / n)  
}

# (Regression sum of squares)  
rss = function(regressor, response)  
# (i.e. Tretament or block sum of squares)  
{  
 sum = 0  
 levels = unique(regressor)  
 for(level in levels)  
 {  
 sum = sum + lms(level, regressor, response)  
 }  
 return(sum - cf)  
}  
  
# Total sum of squares  
tss = 0  
for(i in y)  
{  
 tss = tss + i^2  
}  
tss = tss - cf

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# LATIN SQUARE DESIGN

Latin square design or LSD is an experimental design. Hence, it is applied for the data collection process, not the analysis. To do this, we must have at least three factors and a response. For best results, all factors must have equal number of levels. Two of these factors may be usable in classify the experimental units. If yes, then these are the blocking factors. Together, they form a grid for experimental units to sit in. The third factor is considered as the treatment, and each level is applied randomly to the experimental units so that every cell of th grid i.e. blocking factor combo is subjected to every treatment at only once. It is assumed that there is no interaction between any of the factors.

b1 = myData$block  
b2 = myData$density  
data = data.frame(y, t, b1, b2)

## LINEAR REGRESSION MODEL

data = data.frame(y, t, b1, b2)  
model = lm(y~., data)  
summary(model)

##   
## Call:  
## lm(formula = y ~ ., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -33.35 -12.00 -1.89 12.15 40.67   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 92.6957 5.3638 17.282 <2e-16 \*\*\*  
## tNP 12.0193 4.9603 2.423 0.0182 \*   
## tP 7.3013 4.9473 1.476 0.1448   
## b1B 2.6357 4.9777 0.530 0.5983   
## b1C 8.7688 4.9646 1.766 0.0820 .   
## b2low -10.0776 4.9734 -2.026 0.0468 \*   
## b2medium 0.4321 4.9734 0.087 0.9310   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 17.12 on 65 degrees of freedom  
## Multiple R-squared: 0.1978, Adjusted R-squared: 0.1237   
## F-statistic: 2.671 on 6 and 65 DF, p-value: 0.02227

***We see that the error degrees of freedom is 65.***

error\_df = 65

## ERROR MEAN SQUARE

# Error sum of squares...  
ess = tss - rss(t, y) - rss(b1, y) - rss(b2, y)  
# Error mean square...  
ems\_lsd = ess / error\_df  
ems\_lsd

## [1] 288.4479

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# RANDOMISED BLOCK DESIGN

We will use the same blocking factors as LSD, but will make separate RBD models for each.

## LINEAR REGRESSION MODEL

# Multiple linear regression model for two regressors...  
model = lm(y~t + b1)  
summary(model)

##   
## Call:  
## lm(formula = y ~ t + b1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -38.29 -10.88 -1.57 11.14 43.46   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 89.077 4.529 19.668 <2e-16 \*\*\*  
## tNP 11.983 5.098 2.351 0.0217 \*   
## tP 7.283 5.084 1.432 0.1567   
## b1B 3.459 5.084 0.680 0.4986   
## b1C 9.210 5.098 1.807 0.0753 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 17.6 on 67 degrees of freedom  
## Multiple R-squared: 0.1267, Adjusted R-squared: 0.07453   
## F-statistic: 2.429 on 4 and 67 DF, p-value: 0.05612

**Here, we see error DF as 67.**

error\_df = 67

## # ERROR MEAN SQUARE FOR RBD WITH BLOCKING FACTOR 1 (b1)

# Error sum of squares...  
ess = tss - rss(t, y) - rss(b1, y)  
# Error mean square...  
ems\_rbd\_b1 = ess / error\_df  
ems\_rbd\_b1

## [1] 306.0817

## # ERROR MEAN SQUARE FOR RBD WITH BLOCKING FACTOR 2 (b2)

# Error sum of squares...  
ess = tss - rss(t, y) - rss(b2, y)  
# Error mean square...  
ems\_rbd\_b2 = ess / error\_df  
ems\_rbd\_b2

## [1] 298.8155

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# COMPLETELY RANDOMISED DESIGN

## LINEAR REGRESSION MODEL

# Multiple linear regression model for one regressor...  
model = lm(y~t)  
summary(model)

##   
## Call:  
## lm(formula = y ~ t)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -32.917 -13.083 -2.917 13.354 43.083   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 92.917 3.627 25.622 <2e-16 \*\*\*  
## tNP 12.750 5.129 2.486 0.0153 \*   
## tP 7.667 5.129 1.495 0.1395   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 17.77 on 69 degrees of freedom  
## Multiple R-squared: 0.08324, Adjusted R-squared: 0.05667   
## F-statistic: 3.132 on 2 and 69 DF, p-value: 0.04987

**Here, we see error DF as 69.**

error\_df = 69

## ERROR MEAN SQUARE

# Error sum of squares...  
ess = tss - rss(t, y)  
# Error mean square...  
ems\_crd = ess / error\_df  
ems\_crd

## [1] 315.6377

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# RELATIVE EFFICIENCIES

**LSD VS. RBD WITH BLOCKING FACTOR 1**  
100\*(1/ems\_lsd)/(1/ems\_rbd\_b1)

## [1] 106.1133

**LSD VS. RBD WITH BLOCKING FACTOR 2**  
100\*(1/ems\_lsd)/(1/ems\_rbd\_b2)

## [1] 103.5943

**LSD VS. CRD**  
100\*(1/ems\_lsd)/(1/ems\_crd)

## [1] 109.4263

As we can see, LSD is 6.1133% and 3.5943% more efficient than the two RBD models respectively. LSD is 9.4263% more efficient than CRD model.Note that greater efficiency in a model means that it takes smaller samples to achieve more accurate predictions and estimations (predictions about the significance of effects of the factors, and estimations about the linear regression model that gets created before the ANOVA test).