QUESTION 3 - LSD MISSING VALUE ESTIMATION

Pranav Gopalkrishna, 1940223

2021-11-30

# AIM

Demonstrate the concept of missing plot technique of Latin Square Design (LSD) using a suitable example and give your interpretation. (Only one missing value case)

#========================

# FORMULA

X = [t(R’ + C’ + T’) - 2G’]/[(t - 2)(t - 2)]

# where...  
# X is the missing value's estimate  
# R' is the total of the available values in the row blocking factor.  
# C' is the total of the available values in the column blocking factor.  
# T' is the total of the available values in the treatment.  
# t is the number of levels in the treatment.  
# G' is the total of the available values in the data set.  
#========================

# DATA SET

setwd("/Users/pranav/Documents/Study/computerScience/programming/r/data/")  
data = read.csv("weatherSouthAfrica.csv")[c(2, 5, 22, 23)]  
head(data)

## Location Rainfall RainToday RainTomorrow  
## 1 Cape Town 0.6 High Low  
## 2 Cape Town 2.7 Medium Low  
## 3 Cape Town 4.0 Low Medium  
## 4 Cape Town 4.3 Low Low  
## 5 Cape Town 1.0 Medium Medium  
## 6 Cape Town 0.2 Low Low

tr = data$Location  
r = data$RainToday  
c = data$RainTomorrow  
y = data$Rainfall  
  
# Summary about the data  
summary(data.frame(as.factor(tr), as.factor(r), as.factor(c), y))

## as.factor.tr. as.factor.r. as.factor.c. y   
## Cape Town :10 High : 8 High : 8 Min. : 0.000   
## Durban :11 Low :13 Low :12 1st Qu.: 1.100   
## Johannesberg:11 Medium:11 Medium:12 Median : 2.700   
## Mean : 3.748   
## 3rd Qu.: 4.950   
## Max. :12.000   
## NA's :1

#========================

# MISSING VALUE ESTIMATION

## Checking how many missing values are there in the response

sum(is.na(y))

## [1] 1

## Locating the missing value

for(i in c(1:length(y)))  
{  
 if(is.na(y[i]))  
 {  
 r\_at\_X = r[i]  
 c\_at\_X = c[i]  
 tr\_at\_X = tr[i]  
 i\_at\_X = i  
 }  
}

## Calculating R’

R\_prime = 0  
for(i in c(1:length(y)))  
{  
 if(r[i] == r\_at\_X & i != i\_at\_X){R\_prime = R\_prime + y[i]}  
}  
R\_prime

## [1] 29.2

## Calculating C’

C\_prime = 0  
for(i in c(1:length(y)))  
{  
 if(c[i] == c\_at\_X & i != i\_at\_X){C\_prime = C\_prime + y[i]}  
}  
C\_prime

## [1] 33.3

## Calculating T’

T\_prime = 0  
for(i in c(1:length(y)))  
{  
 if(tr[i] == tr\_at\_X & i != i\_at\_X){T\_prime = T\_prime + y[i]}  
}  
T\_prime

## [1] 48.5

## Calculating G’

G\_prime = 0  
for(i in c(1:length(y)))  
{  
 if(i != i\_at\_X){G\_prime = G\_prime + y[i]}  
}  
G\_prime

## [1] 116.2

## Number of treatments

t = length(unique(tr))  
t

## [1] 3

## Estimating X (missing value)

X = (t\*(R\_prime + C\_prime + T\_prime) - 2\*G\_prime)/(t\*t)  
# t\*t was chosen instead of (t-1)(t-2) since t is too small, and was yielding grossly deviating esimations  
X

## [1] 11.17778

#========================

# CONCLUSIONS

We get the missing value esimate as 11.17778, which is much higher than the average, and close to the maximum. This may be due to the following factors:

# - Error in my calculations  
# - Small and potentially biased sample