

## Calculo de la Eficiencia de los diseños.

La eficiencia relativa de un diseño se mide respecto a otro diseño mas simple, por ejemplo la eficiencia del diseño de bloques frente al diseño completamente al azar.

La variancia de la media de tratamientos = variancia del error / rep. Y es una medida de la precision de las medias de tratamiento estimadas. Esta precision se controla entonces variando el numero de repeticiones y disminuyendo la variacion del error, entonces el uso de la variancia del error proporciona un medio para comparar la eficiencia relativas de dos diseños experimentales.

Según Fisher, 1960, calculo la cantidad de informacion que proporciona la diferencia estimada entre dos medias respecto a la diferencia real entre las medias de las poblaciones.

$$I = \frac{(f+1)}{\sigma^2(f+3)}; f = \text{gl error. } \sigma^2 \text{ es la variancia del error.}$$

Se compara los indices de los diseños mediante una relacion de los dos indices.

Interpretacion.

Eficiencia relativa del diseño 1 y 2.

Relative Efficiency;  $RE = \frac{I_1}{I_2}$ ; si  $RE = 1$ , la informacion de los dos diseños es la misma.

$RE > 1$ , el diseño 1 es mas eficiente,  $RE < 1$  el diseño 2 es mas eficiente.

Eficiencia de los diseños de bloques incompletos.

El indice de Fisher no es apropiado para comparar este diseño con otro, porque no es posible estimar la variancia del error para el otro diseño, a partir del diseño de bloques incompletos.

Si se utiliza la variancia para comparar dos medias de tratamientos mediante la diferencia de medias.

En el diseño de bloques completos al azar.

$\text{Var}(t_i - t_j) = 2\sigma^2_1/r$ ;  $\sigma^2_1$  es la variancia del error del diseño de bloques completos al azar

En el diseño de bloques incompletos

$\text{Var}(t_i - t_j) = 2k\sigma^2_2/\lambda t$ ;  $\sigma^2_2$  es la variancia del error del diseño de bloques incompletos.

Eficiencia del diseño de bloques incompletos frente al diseño de bloques completos:

$$RE = (1/\text{Var}(t_i - t_j) \text{ del BIB}) / (1/\text{Var}(t_i - t_j) \text{ del DBCA}) = \text{Var}(t_i - t_j)_{\text{DBCA}} / \text{Var}(t_i - t_j)_{\text{BIB}}.$$

Eficiencia =  $(\sigma_1^2/\sigma_2^2) (\lambda t / rk)$ ;  $E = \lambda t / rk$  se conoce como factor de eficiencia.

Entonces la eficiencia del diseño bloques incompletos balanceados puede ser medido solo por el factor de eficiencia, donde  $r$ =rep,  $t$ =tratamientos,  $k$ =tamaño del bloque y  $\lambda$ = la repeticion de los pares de tratamientos.

### **Implicancia de la unidades perdidas en los diseños balanceados.**

El DCA puede ser balanceado o no, el analisis las sumas de cuadrados de tratamientos no se ve afectado.

En el caso de los diseños DBCA, DCL y BIB si tiene efecto, las sumas de cuadrados de tratamientos se ve afectado.

Para el caso de DBCA y DCL, las unidades perdidas puedes ser encontradas, asumiendo que el error en la unidad perdida es cero. Esto significa que si se construye el modelo sin la existencia de la unidad, los programas, caso el `aov()` del R, hallara las sumas de cuadrados correctamente, porque el proceso es en forma matricial.

```
modelo <- aov( y ~ row + col + trt)
anova(modelo)
```

Los valores perdidos pueden ser estimado mediante la prediccion del modelo encontrado.

```
predict( modelo, data.frame( predictores) )
```

Sin el uso de las computadoras, se llegaron a a determinar formulas, partiendo del supuesto que el error es cero en la unidad perdida.

$$\text{Caso de bloques: } x = \frac{tT + rR - G}{(t-1)(r-1)}; Sd = \sqrt{\sigma^2 \left( \frac{2}{r} + \frac{t}{r(r-1)(t-1)} \right)}$$

$$\text{Caso de DCL: } x = \frac{t(R + C + T) - 2G}{(t-1)(t-2)}; Sd = \sqrt{\sigma^2 \left( \frac{2}{r} + \frac{t}{(r-1)(r-2)} \right)}$$

$$\text{Sesgo en DCL: } \frac{(G - R - C - (t-1)T)^2}{((t-1)(t-2))^2}$$

Aplicacion en el DCL.

```
> A
      plots row col trt resp
1         1   1   1   3    2
2         2   1   2   2   NA
3         3   1   3   1    4
4         4   1   4   4    3
5         5   2   1   2    5
6         6   2   2   1    6
7         7   2   3   4    7
8         8   2   4   3    5
9         9   3   1   4    8
10        10   3   2   3    7
11        11   3   3   2    4
12        12   3   4   1    5
13        13   4   1   1    6
14        14   4   2   4    4
15        15   4   3   3    3
16        16   4   4   2    2
```

```
> B<-t(matrix(A[,5],c(4,4)))
> B
```

```
      [,1] [,2] [,3] [,4]
[1,]    2  NA    4    3
[2,]    5    6    7    5
[3,]    8    7    4    5
[4,]    6    4    3    2
```

```
> T<-t(matrix(A[,4],c(4,4)))
> T
```

```
      [,1] [,2] [,3] [,4]
[1,] "3"  "2"  "1"  "4"
[2,] "2"  "1"  "4"  "3"
[3,] "4"  "3"  "2"  "1"
[4,] "1"  "4"  "3"  "2"
```

```
> m1<-aov(resp ~ row+col+trt,A)
> anova(m1)
```

Analysis of Variance Table

Response: resp

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
row	3	23.4333	7.8111	6.0085	0.04113 *
col	3	5.5000	1.8333	1.4103	0.34299
trt	3	11.5000	3.8333	2.9487	0.13734
Residuals	5	6.5000	1.3000		

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> G<- sum(B,na.rm=T)
> G
[1] 71

> tapply.stat(A$row,A[,5],function(x) sum(x,na.rm=T))
  A$row A[, 5]
1      1      9
2      2     23
3      3     24
4      4     15
> tapply.stat(A$col,A[,5],function(x) sum(x,na.rm=T))
  A$col A[, 5]
1      1     21
2      2     17
3      3     18
4      4     15
> tapply.stat(A$trt,A[,5],function(x) sum(x,na.rm=T))
  A$trt A[, 5]
1      1     21
2      2     11
3      3     17
4      4     22

# Unidad perdida y el sesgo mediante la formula

> (4*(9+17+11)-2*71)/6
[1] 1
> sesgo <- (71-9-17-3*11)^2/36
> sesgo
[1] 4

Prediccion del valor perdido mediante el modelo.

> predict(m1,data.frame(row="1", col="2", trt="2") )
1
1

```

Caso con 2 unidades perdidas.

```
> A
  plots row col trt resp
1     1   1   1   3    2
2     2   1   2   2   NA
3     3   1   3   1    4
4     4   1   4   4    3
5     5   2   1   2   NA
6     6   2   2   1    6
7     7   2   3   4    7
8     8   2   4   3    5
9     9   3   1   4    8
10    10   3   2   3    7
11    11   3   3   2    4
12    12   3   4   1    5
13    13   4   1   1    6
14    14   4   2   4    4
15    15   4   3   3    3
16    16   4   4   2    2
```

```
> B
      [,1] [,2] [,3] [,4]
[1,]     2    X    4    3
[2,]     Y    6    7    5
[3,]     8    7    4    5
[4,]     6    4    3    2
```

```
> T
      [,1] [,2] [,3] [,4]
[1,] "3"  "2"  "1"  "4"
[2,] "2"  "1"  "4"  "3"
[3,] "4"  "3"  "2"  "1"
[4,] "1"  "4"  "3"  "2"
```

$$X = (4(\text{row} + \text{col} + \text{trt}) - 2G)/6$$

Ecuacion para x

$$[4(9 + 17 + 6+y) - 2(66+y)]/6 = x$$

Ecuacion para y

$$[4(18 + 16 + 6+x) - 2(66+x)]/6 = y$$

$$128 + 4y - 132 - 2y = 6x$$

$$160 + 4x - 132 - 2x = 6y$$

$$\begin{aligned} -6x + 2y &= 4 \\ 2x - 6y &= -28 \end{aligned}$$

Solucion:  $x = 1, y = 5$

```
> predict(m3,data.frame(row="1",col="2",trt="2"))
1
> predict(m3,data.frame(row="2",col="1",trt="2"))
5
```

$$\text{Sesgo } x = ((66)-9-17-3*(6))^2/36 = 4$$

$$\text{Sesgo } y = ((66)-18-16-3*(6))^2/36 = 4$$

Solo datos originales:

```
> anova(m3)
```

Analysis of Variance Table

Response: resp

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
row	3	24.1071	8.0357	4.9451	0.07832 .
col	3	7.9500	2.6500	1.6308	0.31657
trt	3	8.3000	2.7667	1.7026	0.30335
Residuals	4	6.5000	1.6250		

---

```
> m4<- aov(formula = resp ~ row + col + trt, data = A)
> anova(m4)
```

Analysis of Variance Table

Response: resp

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
row	3	33.500	11.167	10.3077	0.008795 **
col	3	4.500	1.500	1.3846	0.335117
trt	3	15.500	5.167	4.7692	0.049744 *
Residuals	6	6.500	1.083		

---

Ejemplo 2.

```
> A
      plots row col trt  y
1         1   1   1   4   1
2         2   1   2   3   3
3         3   1   3   2   6
4         4   1   4   1   5
5         5   2   1   3   1
6         6   2   2   2   2
7         7   2   3   1  NA
8         8   2   4   4   6
9         9   3   1   2   5
10        10   3   2   1   8
11        11   3   3   4   2
12        12   3   4   3  NA
13        13   4   1   1   3
14        14   4   2   4   4
15        15   4   3   3   2
16        16   4   4   2   1
```

```
> modelo <- aov(y ~ row + col + trt, data=A)
> anova(modelo)
```

Analysis of Variance Table

Response: y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
row	3	25.187	8.396	1.1352	0.4074
col	3	15.687	5.229	0.7070	0.5820
trt	3	15.687	5.229	0.7070	0.5820
Residuals	6	44.375	7.396		

	T	R	C	G
x =	16		9	10 49 + y
y =	6		15	12 49 + x

```
X = (4(16+9+10)-2(49+y))/6
Y = (4(6+15+12)-2(49+x))/6
6X = (140-98-2y)
6y = (132-98-2x)
6X = (42-2y)
6y = (34-2x)
```

```
3x + y = 21
X + 3y = 17
```

```
> M <- rbind(c(3,1),c(1,3))
> b <- c(21,17)
> solve(M,b)
5.75 3.75
```

```
predict(modelo,data.frame(row="2",col="3",trt="1"))
5.75
predict(modelo,data.frame(row="3",col="4",trt="3"))
3.75
```