

Plans expérimentaux pour l'agronomie avec R

Benjamin Heuclin, Philippe Letourmy, Eric Gozé

Cirad, UR AIDA, 2023

Contents

Introduction	2
Plans à un facteur étudié	5
Plan à un facteur étudié en randomisation totale	5
Plans à un facteur étudié avec contrôle de l'erreur	16
Plan en blocs complets à un facteur de contrôle	16
Plans en ligne colonne : carrés latins	22
Plans en blocs incomplets : bie, bipe,	28
Plans en Blocs Incomplets Equilibrés (BIE)	28
Plan Alpha Lattice : BIE avec des groupes de répétition	36
Plans factoriels	44
Plans factoriels en randomisation totale	46
Plans factoriels en randomisation totale sans répétition	46
Plans factoriels en randomisation totale avec répétitions	50
Plan factoriel en blocs à un facteur de contrôle d'hétérogénéité	57
Plan en bloc complet en randomisation totale (RCB design)	57
Plans factoriels avec plusieurs unités expérimentales emboîtées ou croisées	61
Split-plot : des effets principaux sont confondus avec des blocs	61
Plan criss-cross (en bande croisées) :	70
Plan à 2 facteurs étudier avec contrôle d'hétérogénéité sur les lignes et les colonnes : carrés gréco-latins ou latins orthogonaux	75
Références	80

Ce guide pratique a pour but d'introduire les principaux plans d'expérimentation pour l'agronomie, d'expliquer comment les générer et les analyser avec R. Il fait appel à des notions de base en statistique comme la régression, le modèle linéaire et l'analyse de variance (ANOVA) et plus évoluées comme le modèle linéaire mixte. Ce guide n'a pas pour but d'introduire la théorie statistique et nous renvoyons le lecteur vers les livres de références suivants si besoin :

- Rencher, A. C., & Schaalje, G. B. (2008). Linear models in statistics. John Wiley & Sons.
- McCulloch, C. E., & Searle, S. R. (2004). Generalized, linear, and mixed models. John Wiley & Sons.
- Pinheiro, J., & Bates, D. (2000). Mixed-effects models in S and S-PLUS. Springer science & business media.

Librairies nécessaires pour utiliser les fonctions présentées dans ce guide :

```
library(tidyverse)
library(gridExtra)
library(nlme)
library(lmerTest)
library(agricolae)
library(agridat)
library(desplot)
```

Introduction

Deux étapes sont cruciales dans toute étude statistique :

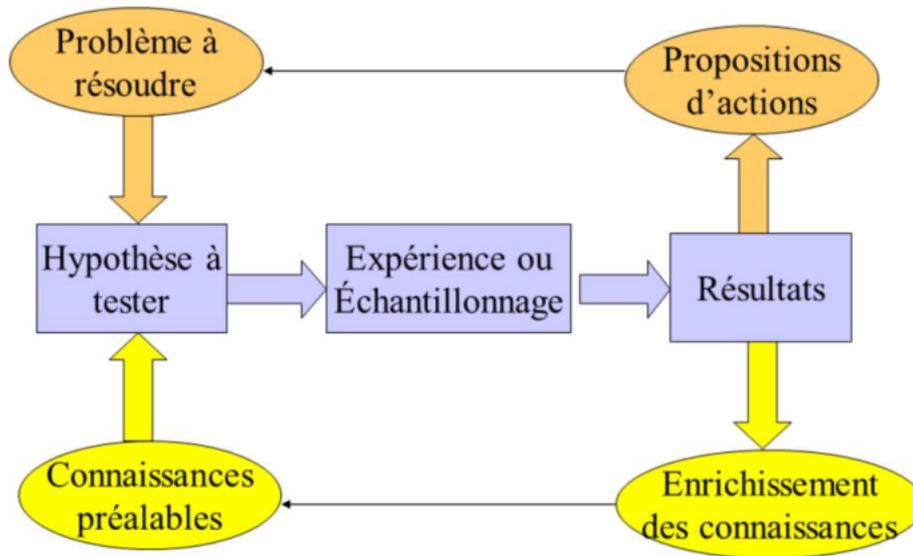
1. Le recueil des données
2. L'analyse de ces données

Souvent, l'accent est mis sur l'ensemble des méthodes d'analyse statistique des données, mais la question des méthodes de recueil peut être négligée. Toute étude statistique passe par un recueil de données.

Tout recueil d'informations est conçu en fonction d'un objectif, et cet objectif conditionne la manière dont seront collectées les données et les questions auxquelles on tentera d'apporter une réponse.

On considère deux modes de recueil planifié de l'information : l'expérimentation et l'échantillonnage. L'expérience recherche analytiquement des relations de cause à effet. L'échantillonnage recherche la généralisation de conclusions d'observations à toute une population.

Démarche utilisée



Les problèmes rencontrés (en agriculture, en médecine, dans l'industrie...) et les connaissances préalables concourent à la définition d'une hypothèse à tester par la planification du recueil de données, ce qui aboutit à des résultats, sous la forme d'une hypothèse nouvelle. Ces résultats débouchent d'une part sur des propositions d'actions concrètes, d'autre part sur un enrichissement des connaissances, qui peuvent aussi être utilisées pour résoudre des problèmes similaires.

Vocabulaires :

- Facteur : toute série d'éléments de même nature conditionnant le phénomène étudié (on distingue le facteur étudié et celui de contrôle, mais aussi facteur qualitatif ou quantitatif)
- Niveau ou modalité : un des éléments qui constituent un facteur
- Objet ou traitement : toute combinaison de niveaux ou de modalités de tous les facteurs étudiés
- Facteurs de contrôles (blocs) : facteurs liés à la variabilité du milieu et introduits de façon à ce que leurs effets puissent être éliminés.
- Unité expérimentale (ue) : unité statistique sur laquelle chaque traitement est affecté au hasard par la randomisation du dispositif et sur laquelle est faite chaque mesure
- Champ d'expérience : ensemble des unités expérimentales
- Champ d'application : domaine sur lequel les résultats sont applicables
- Observation : mesure de caractère quantitatif ou qualitatif effectuée sur chaque unité expérimentale
- Plan d'expérience : il définit la manière dont les traitements sont affectés aux unités expérimentales
- Erreur expérimentale/résiduelle = erreur technique/de mesure + erreur unitaire/de randomisation (erreur due à l'hétérogénéité des unités expérimentales)
- Plan factoriel : dispositif dans lequel chaque traitement est une combinaison de niveaux de plusieurs facteurs.

Règle des 3 R :

- **R**épétition : Estimation de la variance de la loi des erreurs

- **R**andomisation : loi de la statistique sous H_0
- **cont**Rôle de l'erreur

Pourquoi randomiser ?

Reprenons l'expérience décrite par Fisher dans son livre "The Design of Experiments" de 1935.

Mlle Bristol affirme pouvoir faire la différence si le thé est servi avant le lait et inversement. Pour vérifier cela, Ronald Aylmer Fisher met au point l'expérience suivante : 4 tasses avec thé en premier (traitement T) et 4 tasses avec lait en premier (traitement L) vont lui être soumises. Les tasses sont positionnées successivement et numérotées de 1 à 8. La question est alors de savoir comment affecter les traitements aux tasses :

- de manière ordonnée : le traitement L pour les tasses 1 à 4 puis T pour les tasses 5 à 8 par exemple
- ou de manière totalement aléatoire : on choisit 4 numéros de tasse entre 1 et 8 auxquelles on affecte le traitement T et on affecte le traitement L aux 4 restantes.

Bien souvent, des effets indésirables, bien souvent non-contrôlables, peuvent venir perturber notre expérience, par exemple un thé plus fort au fond de la théière, les 4 premières tasses servies qui vont refroidir plus vite que les 4 dernières, Si les tasses sont présentées de manière ordonnée, alors ces effets indésirables peuvent aider Mlle Bristol à distinguer les 4 premières tasses des 4 dernières et finalement à réussir le test : l'absence de randomisation rend probable la confusion entre un effet indésirable et les capacités de Mlle Bristol à identifier l'ordre de service du thé et du lait, et l'introduction d'un biais. De plus, si l'ordre dans lequel les tasses versées lait en premier ou thé en premier n'est pas aléatoire, Mlle Bristol peut essayer de le deviner.

L'affectation des traitements de manière totalement aléatoire permet ainsi d'éviter toute confusion avec des effets indésirables et permet un résultat sans biais. On appelle cela la randomisation. Une démonstration plus mathématique est donnée par Bergonzini dans son livre "Analyse et planification des expériences" (annexe 2 du chapitre 1).

La randomisation doit être faite indépendamment pour chaque essai. Aucun plan ne doit être réutilisé sans une nouvelle randomisation.

Pourquoi répéter l'expérience ?

Sur le même matériel végétal cultivé dans les mêmes conditions, deux parcelles différentes d'un même champ donneront deux résultats différents : une variabilité naturelle est inévitable. De même en médecine, en génie des procédés, en zootechnie les résultats mesurés sont variables d'une unité expérimentale à l'autre.

Par conséquent, à moins de connaître à l'avance l'ampleur de cette variabilité naturelle, si pour comparer deux traitements (par exemple deux variétés) on se contente de les placer dans deux parcelles distinctes, on ne pourra pas savoir si la différence entre les deux résultats qui vont en sortir est la conséquence des traitements ou de la variabilité naturelle : il nous faut dans un même dispositif expérimental, dans une même expérience, mesurer à la fois la différence entre traitements et l'ampleur de la variabilité naturelle.

Pour cela, le plus simple est de répéter chacun des traitements sur des parcelles différentes, des unités expérimentales distinctes. Alors à l'aide de l'analyse de variance ou de méthodes plus sophistiquées si cela est nécessaire les répétitions permettent d'évaluer d'une part la variabilité expliquée par le traitement (variabilité inter-traitements) et d'autre part la variabilité non expliquée par le traitement aussi appelée variabilité résiduelle.

Nous verrons également que le nombre de répétitions influe directement sur la puissance des tests qu'on fera pour déterminer si effet du traitement il y a. Dans le cas où il y aurait réellement un effet, plus il y a de répétitions, plus la probabilité de ne pas détecter d'effet (erreur de type II) sera faible et donc la puissance du test sera forte.

Attention répéter les mesures sur la même unité expérimentale n'est pas répéter les traitements.

Pourquoi le contrôle de l'erreur ?

Il faut réduire la part non contrôlée de l'expérience pour diminuer la part de variabilité résiduelle.

Enfin, il est nécessaire d'avoir une erreur expérimentale/résiduelle aussi petite que possible, c'est-à-dire de contrôler l'erreur : choisir un champ expérimental aussi homogène que possible.

Pour cela, il ne faut pas implanter un essai à l'emplacement d'un précédent essai qui a forcément introduit une hétérogénéité sur le terrain. C'est évident pour les essais de fumure, mais également tous les autres essais provoquent des exportations de minéraux différentes selon les parcelles ; les allées provoquent également une hétérogénéité supplémentaire. C'est pourquoi il est indispensable de faire une ou deux années de "repos" du terrain entre deux essais (jachère travaillée, multiplication d'une même variété, ou autres).

Lorsqu'une hétérogénéité est reconnue sur le terrain, on peut définir des blocs formés de groupes de parcelles homogènes. Ceci permet de contrôler un facteur parasite non étudié (d'où le nom de facteur de contrôle) et d'améliorer la précision sur les facteurs étudiés. C'est la famille des plans en blocs que nous aborderons dans ce manuscrit.

Plans à un facteur étudié

Plan à un facteur étudié en randomisation totale

Il ne permet aucun contrôle d'hétérogénéité, il faut donc supposer que le terrain est homogène (mêmes caractéristiques de sol, d'exposition, etc). Il est alors découpé en N unités expérimentales (parcelles). On veut étudier les effets de t traitements (niveaux d'un unique facteur ou combinaisons de niveaux de plusieurs facteurs (plan factoriel en randomisation totale, voir plus bas pour ce type de plan)), répétés chacun n_1, n_2, \dots, n_t fois avec $N = n_1 + n_2 + \dots + n_t$. Le plan en randomisation totale consiste à affecter aléatoirement les traitements aux N parcelles.

```
library(agricolae)

# noms des différents traitements
Traitement <- c("T1", "T2", "T3", "T4")

# nombre de répétitions pour chaque traitement
repetition <- c(3, 3, 3, 3)

# La fonction design.crd permet de générer le plan en randomisation totale. Le résultat renvoyé est une
design <- design.crd(trt = Traitement, r = repetition)

# le plan généré est obtenu dans l'élément book de la liste.
design_crd <- design$book

# on change le nom "plot" en "Parcelles"
names(design_crd)[1] <- "Parcelles"

# on change "r" en "Répétitions"
names(design_crd)[2] <- "Répétitions"
design_crd
```

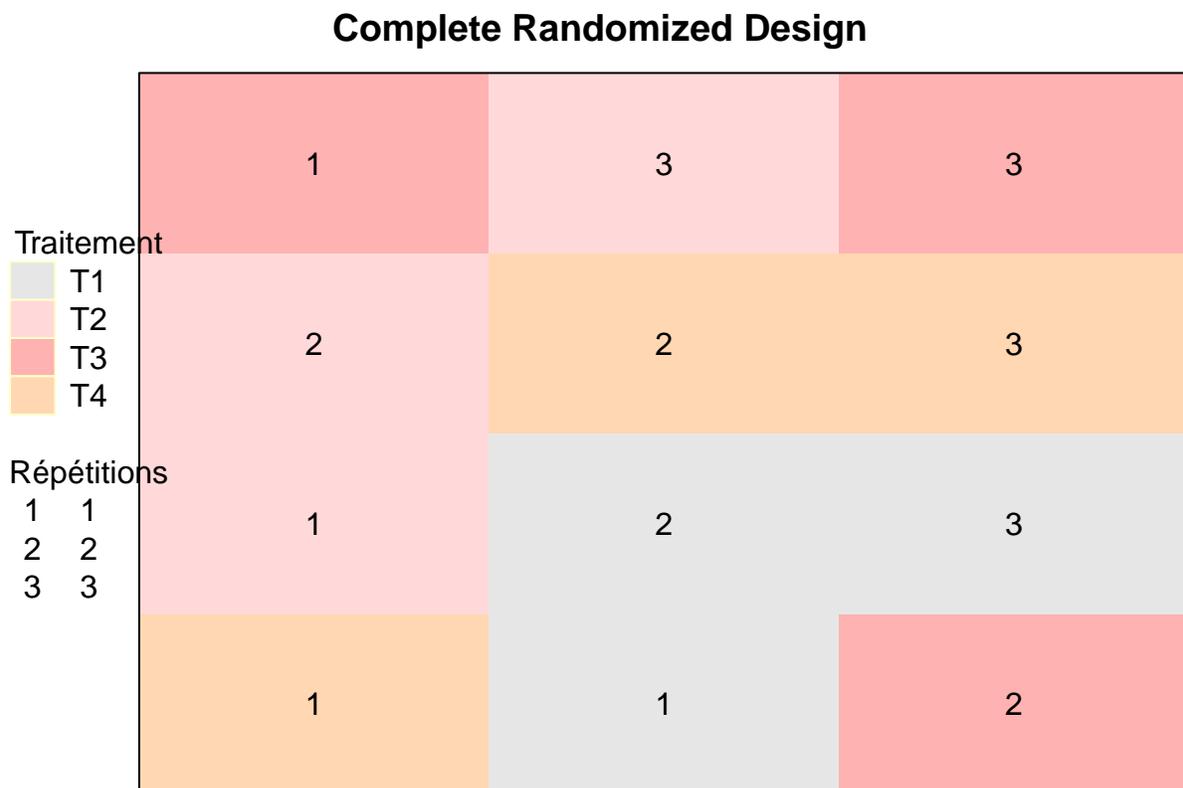
```
##   Parcelles Répétitions Traitement
## 1      101           1           T4
## 2      102           1           T2
## 3      103           2           T2
```

```
## 4      104      1      T3
## 5      105      1      T1
## 6      106      2      T1
## 7      107      2      T4
## 8      108      3      T2
## 9      109      2      T3
## 10     110      3      T1
## 11     111      3      T4
## 12     112      3      T3
```

Si on connaît la disposition des parcelles, on peut représenter les parcelles avec différentes couleurs en fonction des traitements. Imaginons que l'on ait 4 lignes de parcelles et 3 colonnes :

```
design_crd$row = rep(1:4,3)
design_crd$col = rep(1:3, each = 4)
# design_crd

library(desplot)
desplot(form=Traitement ~ col+row, data=design_crd,
        cex=1, num=Répétitions, main="Complete Randomized Design")
```



Modèle statistique :

$$y_{i,j} = \mu + \alpha_i + \varepsilon_{i,j}, \quad \varepsilon_{i,j} \sim N(0, \sigma^2)$$

avec :

- $y_{i,j}$: l'observation associée à la j ème répétition du traitement i
- μ l'intercept (moyenne générale)
- α_i l'effet du traitement i
- $\varepsilon_{i,j}$ l'erreur résiduelle associée à la j ème répétition du traitement i

ATTENTION ! En l'état, le modèle n'est pas estimable. Il faut ajouter une contrainte. Il en existe 2.

La contrainte de type référence : Elle consiste à fixer l'effet du premier traitement (α_1) à 0. Ainsi, l'intercept est égale à la moyenne des observations (ici les rendements) dans le traitement 1. L'effet du traitement 2 (α_2) s'interprète alors comme la différence entre le traitement 1 et le traitement 2. Il en est de même pour les autres traitements. On dira que le traitement 1 est pris en référence (généralement un traitement conventionnel) et qu'on va venir comparer les autres traitements à ce traitement de référence.

C'est la contrainte par défaut dans la fonction `lm`

La contrainte de type Somme : Cette contrainte consiste à faire en sorte que la somme des effets des traitements vaut 0. L'intercept μ s'interprète alors comme la moyenne des moyennes des observations intra-traitements.

Analyse :

On souhaite savoir si le traitement a une influence sur la variable à expliquer Y , soit si les moyennes dans les différents traitements sont différentes les une des autres (les variances des groupes sont supposées égales).

On va donc construire un **test d'analyse de variance (ANOVA)** de la forme :

- H_0 : $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4$ (pas d'influence du traitement)
- H_1 : Il existe au moins deux traitements qui ont des effets différents sur Y $\alpha_l \neq \alpha_k$ (le traitement a une influence)

L'idée du test ANOVA est de décomposer l'erreurs total en une somme d'une erreur inter-groupe et d'une erreur intra-groupe :

$$(y_{ij} - \bar{y}) = (\bar{y}_i - \bar{y}) + (y_{ij} - \bar{y}_i)$$

On élève ensuite au carré les membres de cette relation et on en fait ensuite la somme sur toutes les observations. On obtient alors la somme des carrées :

$$\sum_{ij} (y_{ij} - \bar{y})^2 = \sum_i n_i (\bar{y}_i - \bar{y})^2 + \sum_{ij} (y_{ij} - \bar{y}_i)^2$$

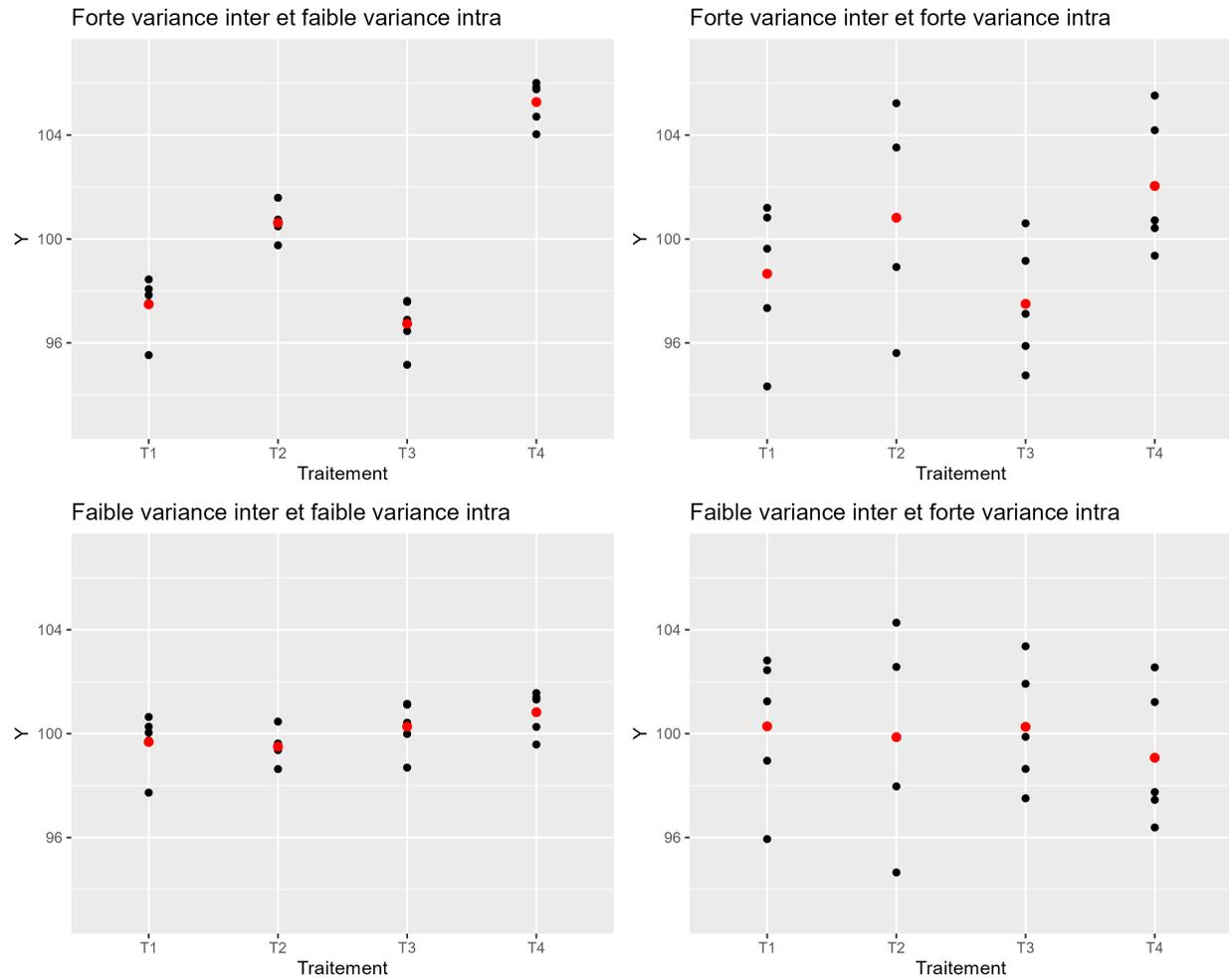
$$SCT = SCE + SCR$$

Remarques :

- $\frac{SCT}{n-1}$ est un estimateur de la variance totale
- $\frac{SCE}{I-1}$ est un estimateur de la variance inter-groupes, expliqué par le traitement (avec I le nombre de niveaux)
- $\frac{SCR}{n-I}$ est un estimateur de la variance intra-groupe, autrement dit la variance résiduelle

En comparant ces deux variances, on peut en déduire si le traitement influe sur Y .

Si le traitement a un effet alors on s'attend à ce que les moyennes intra-groupes soient contrastées et donc que la variance de ces moyennes intra-groupes soit forte. C'est la variance inter-groupe. Cependant, il faut aussi regarder la variance intra-groupe (résiduelle) car si cette variance est très élevée alors on ne peut rien dire.



Le principe du test de Fisher est de comparer la part de variance expliquée par le traitement (variance inter) par rapport à la celle non expliquée par le traitement (variance intra/résiduelle). On calcule donc le rapport entre les deux. Cela nous donne le tableau d'analyse de variance (ANOVA) suivant :

Variation	Sommes des carrés	d.d.l.	Carrés moyens	Valeur F_{obs}	$P_{H_0}(F > F_{obs})$
Facteur A	$SCE_A = \sum_{i=1}^I n_i (\bar{y}_i - \bar{y})^2$ variabilité inter	$I - 1$	$CME_A = \frac{SCE_A}{I-1}$ variance inter, expliquée	$\frac{CME_A}{CMR}$	p-value
Erreur	$SCR = \sum_{i=1}^I \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2$ variabilité intra	$n - I$	$CMR = \frac{SCR}{n-I}$ variance intra, résiduelle		
Totale	$SCT = \sum_{i=1}^I \sum_{j=1}^{n_i} (y_{ij} - \bar{y})^2$	$n - 1$			

⇒ on utilise alors la statistique de test suivante (hypothèse de normalité des résidus nécessaire) :

$$F = \frac{\text{var expliquée par A}}{\text{var résiduelle}} = \frac{CME_A}{CMR} \underset{H_0}{\sim} \mathcal{F}_{I-1, n-I}$$

On calcule ensuite la probabilité d'obtenir une F-value théorique sous l'hypothèse que H0 est vraie (pas d'effet du traitement) supérieur à celle observée. C'est ce qu'on appelle la p-value.

Interprétation de la p-value :

- Si la p-value est inférieur à 5%, alors il est peut probable d'observer une telle F-value sous l'hypothèse que H0 est vrai. On rejette donc H0. Autrement dit le traitement influe sur Y.
- Si la p-value est supérieur à 5%, alors on ne rejette pas H0. On ne peut rien dire sur l'effet du traitement sur Y. Soit il n'y a pas d'effet, soit on n'a pas assez d'observations pour détecter un effet. C'est la notion de puissance statistique.

Et ensuite ?

- Il faut ensuite faire le diagnostic du modèle pour vérifier les hypothèses (la normalité des résidus, variance résiduelle homogène, indépendance des résidus). La fonction `plot.lm` appliquée à un objet de type `lm` permet de sortir 4 graphiques de diagnostic. Voir l'aide de la fonction pour plus de détails :

```
?plot.lm
```

- Si le diagnostic est bon et qu'il y a un effet du traitement, on peut ensuite comparer les moyennes ajustées inter-traitements à l'aide de comparaison multiple de Tukey. Voir l'exemple ci-dessous.

Exemple : sweetpotato

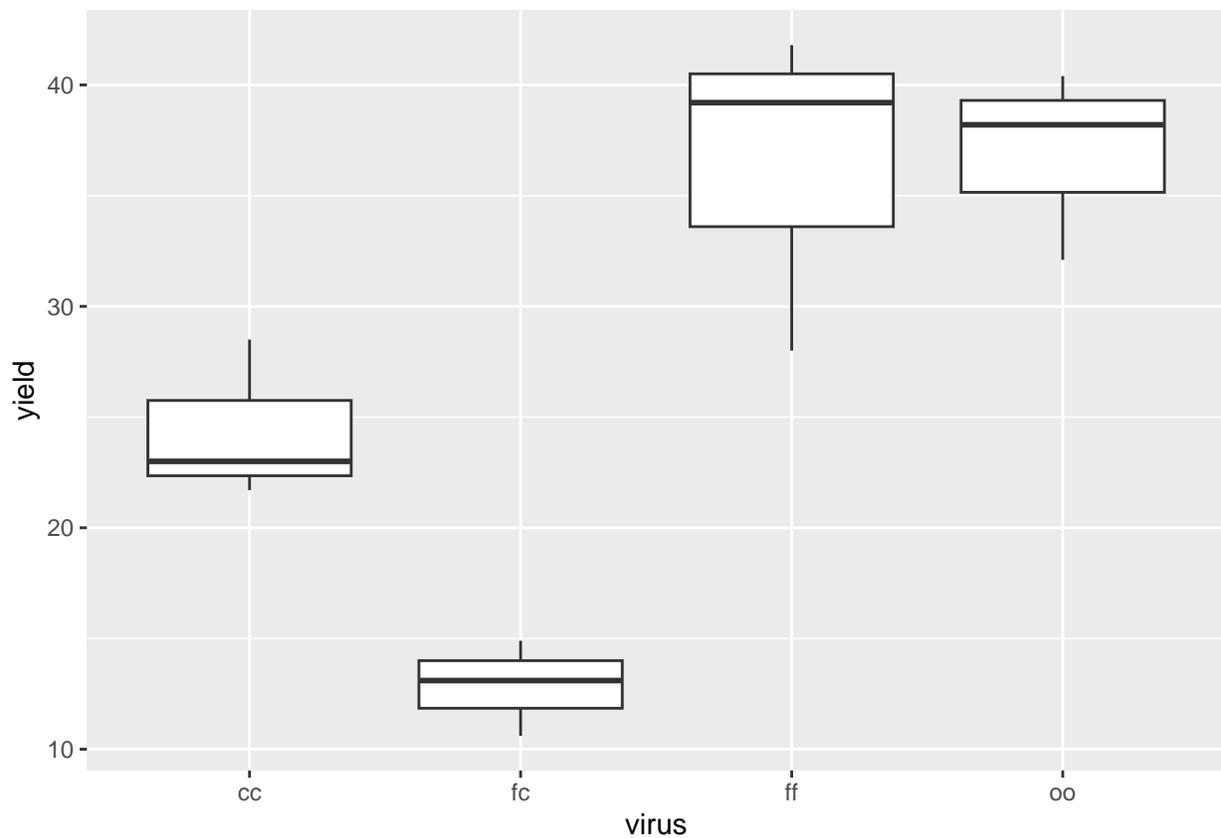
Les données correspondent à une expérience sur la patate douce costanero réalisée dans la localité du département de Tacna, au sud du Pérou. L'effet de deux virus (Spmv et Spcsv) a été étudié. Les traitements

étaient les suivants CC (Spcsv) = patate douce chlorotique naine, FF (Spfmv) = marbrure plumeuse, FC (Spfmv y Spcsv) = complexe viral et OO (témoin) plantes saines. Dans chaque parcelle, 50 plants de patate douce ont été semés et 12 parcelles ont été utilisées. Chaque traitement a été effectué avec 3 répétitions et à la fin de l'expérience, le poids total en kilogrammes a été évalué. La transmission du virus a été effectuée dans les boutures et celles-ci ont été semées dans le champ. Elles sont disponibles dans la librairie `agricolae`.

```
library(agricolae)
data(sweetpotato)
sweetpotato
```

```
##   virus yield
## 1    cc  28.5
## 2    cc  21.7
## 3    cc  23.0
## 4    fc  14.9
## 5    fc  10.6
## 6    fc  13.1
## 7    ff  41.8
## 8    ff  39.2
## 9    ff  28.0
## 10   oo  38.2
## 11   oo  40.4
## 12   oo  32.1
```

```
ggplot(sweetpotato, aes(x=virus, y=yield)) + geom_boxplot()
```



Ici le traitement que l'on souhaite tester est le virus. Le témoin est le niveau "oo". On commence par le spécifier à R avec la fonction `factor` et l'argument `levels` en mettant "oo" en premier :

```
sweetpotato$virus = factor(sweetpotato$virus, levels = c("oo", "cc", "fc", "ff"))
sweetpotato$virus
```

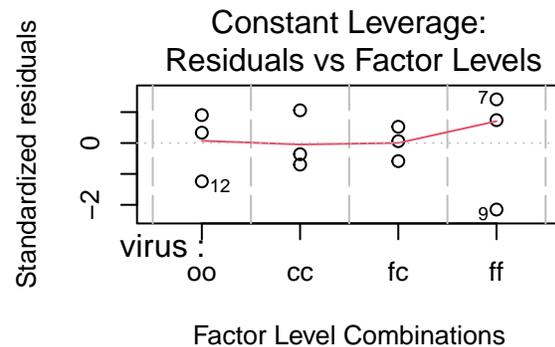
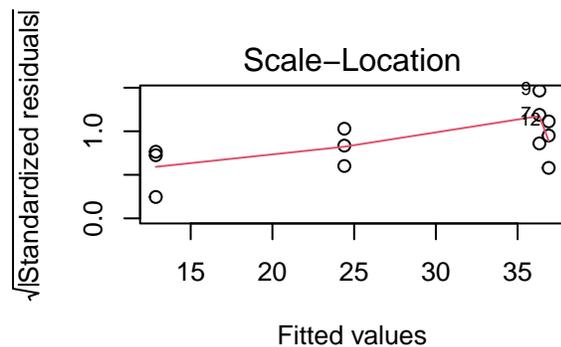
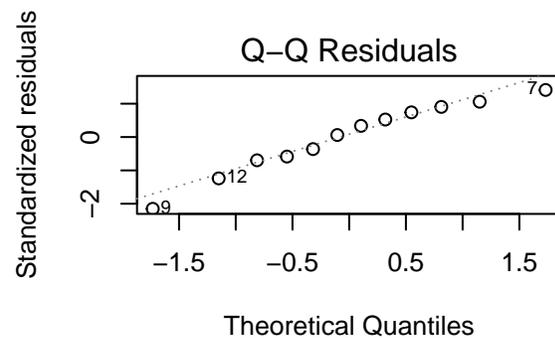
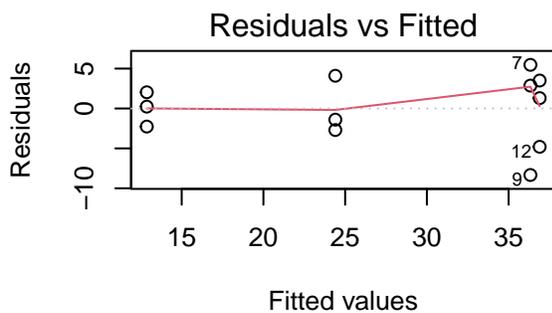
```
## [1] cc cc cc fc fc fc ff ff ff oo oo oo
## Levels: oo cc fc ff
```

```
fit = lm(yield~virus, sweetpotato)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: yield
##          Df Sum Sq Mean Sq F value    Pr(>F)
## virus      3 1170.21   390.07  17.345 0.0007334 ***
## Residuals  8  179.91    22.49
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Diagnostic du modèle :

```
par(mfrow=c(2,2))
plot(fit)
```



On peut aussi tester la normalité des résidus avec le **test de Shapiro** :

- H_0 : l'échantillon testé (ici les résidus) est normalement distribué
- H_1 : l'échantillon ne suit pas une loi normale

Interprétation de la p-value :

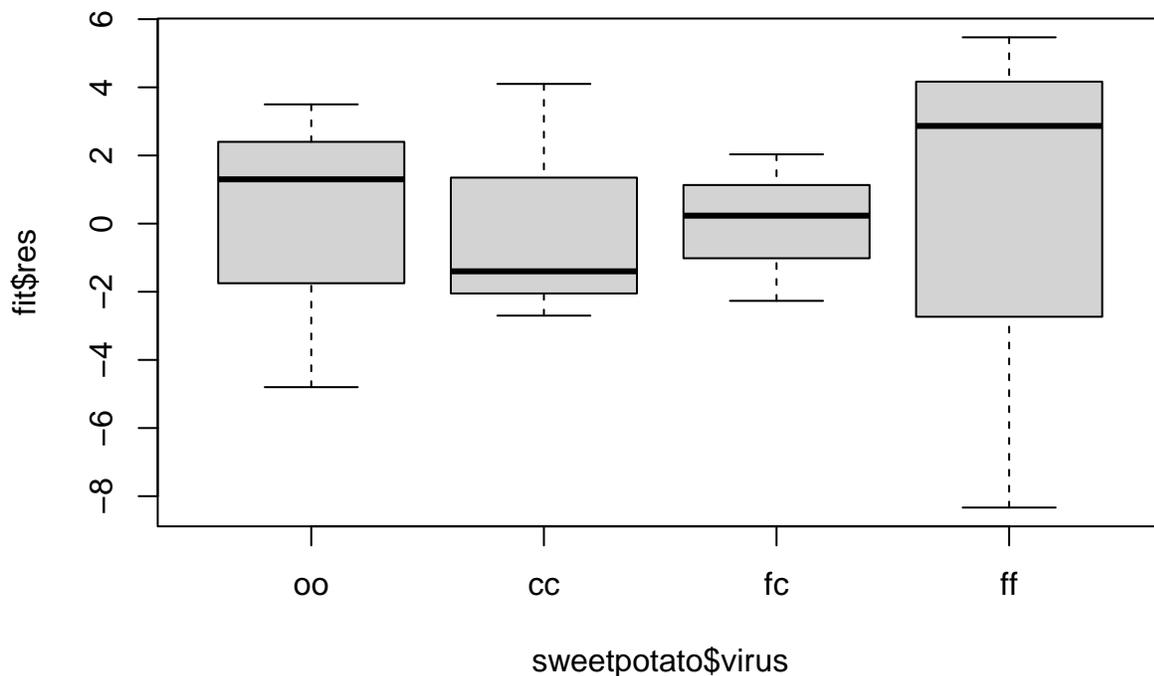
- Si la p-value est inférieur à 5%, alors on rejette H_0 . Autrement dit, il est probable que l'échantillon ne suive pas une loi normale.
- Si la p-value est supérieur à 5%, alors on ne rejette pas H_0 . On accepte l'hypothèse d'une distribution parente normale.

```
shapiro.test(fit$res)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: fit$res  
## W = 0.95878, p-value = 0.7663
```

On peut aussi représenter les résidus par traitement. Idéalement, la répartition des résidus doit être identique d'un traitement à un autre et centré en 0.

```
boxplot(fit$res~sweetpotato$virus)
```



Ici, on peut voir que les résidus sont plus dispersés dans le traitement “ff”.

On peut faire un **test de Levene** pour voir si les variances intra-traitement des résidus sont homogènes

- H0 : $\sigma_{oo}^2 = \sigma_{cc}^2 = \sigma_{fc}^2 = \sigma_{ff}^2$
- H1 : Il existe au moins 2 variances différentes : $\sigma_l^2 \neq \sigma_k^2$

Interprétation de la p-value :

- Si la p-value est inférieur à 5%, alors on rejette H0. Autrement dit les variances résiduelles intra-traitement des résidus ne sont pas homogène
- Si la p-value est supérieur à 5%, alors on ne rejette pas H0. On accepte l’hypothèse homogénéité des variances des résidus intra-traitement.

```
library(car)
```

```
## Le chargement a nécessité le package : carData
```

```
##
```

```
## Attachement du package : 'car'
```

```
## L'objet suivant est masqué depuis 'package:dplyr':
```

```
##
```

```
##   recode
```

```
## L'objet suivant est masqué depuis 'package:purrr':
```

```
##
```

```
##   some
```

```
leveneTest(fit$res ~ virus, data = sweetpotato)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
```

```
##      Df F value Pr(>F)
```

```
## group 3  0.4004 0.7566
```

```
##      8
```

Ici on ne rejette pas H0. On a donc vérifié qu’il n’y avait pas d’hétérogénéité des variances des résidus intra-traitement.

Il existe aussi le test d’homogénéité des variances de Barlett. Les hypothèses sont les mêmes que pour le test de Levene.

```
bartlett.test(fit$res, sweetpotato$virus)
```

```
##
```

```
## Bartlett test of homogeneity of variances
```

```
##
```

```
## data: fit$res and sweetpotato$virus
```

```
## Bartlett's K-squared = 2.3886, df = 3, p-value = 0.4958
```

Calcul des moyennes ajustées et comparaisons 2 à 2 :

Maintenant que l'on sait que le traitement (virus) a un effet sur la variable réponse (rendement), on peut se demander si certains traitements ont un effet significativement différent des autres.

Pour cela, on va calculer les moyennes ajustées et les comparer 2 à 2 à l'aide de tests de Tukey.

Il y a différents packages qui permettent de faire cela :

```
library(agricolae)
out = HSD.test(fit,"virus", group=FALSE,console=TRUE,
              main="Yield of sweetpotato\nDealt with different virus")
```

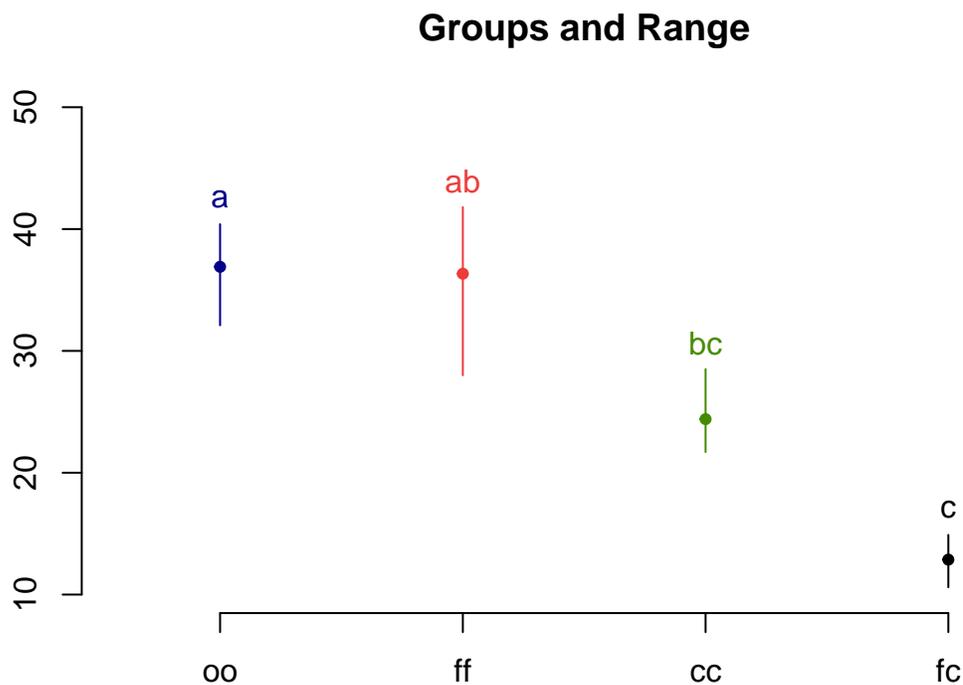
```
##
## Study: Yield of sweetpotato
## Dealt with different virus
##
## HSD Test for yield
##
## Mean Square Error:  22.48917
##
## virus, means
##
##      yield      std r      se Min Max  Q25  Q50  Q75
## cc 24.40000 3.609709 3 2.737953 21.7 28.5 22.35 23.0 25.75
## fc 12.86667 2.159475 3 2.737953 10.6 14.9 11.85 13.1 14.00
## ff 36.33333 7.333030 3 2.737953 28.0 41.8 33.60 39.2 40.50
## oo 36.90000 4.300000 3 2.737953 32.1 40.4 35.15 38.2 39.30
##
## Alpha: 0.05 ; DF Error: 8
## Critical Value of Studentized Range: 4.52881
##
## Comparison between treatments means
##
##      difference pvalue signif.      LCL      UCL
## cc - fc  11.5333333 0.0686      .  -0.8663365  23.9330031
## cc - ff -11.9333333 0.0592      . -24.3330031   0.4663365
## cc - oo -12.5000000 0.0482      * -24.8996698 -0.1003302
## fc - ff -23.4666667 0.0014      ** -35.8663365 -11.0669969
## fc - oo -24.0333333 0.0012      ** -36.4330031 -11.6336635
## ff - oo  -0.5666667 0.9988      -12.9663365  11.8330031
```

```
out = HSD.test(fit,"virus", group=TRUE,console=TRUE,
              main="Yield of sweetpotato\nDealt with different virus")
```

```
##
## Study: Yield of sweetpotato
## Dealt with different virus
##
## HSD Test for yield
##
## Mean Square Error:  22.48917
##
## virus, means
```

```
##
##      yield      std r      se  Min  Max  Q25  Q50  Q75
## cc 24.40000 3.609709 3 2.737953 21.7 28.5 22.35 23.0 25.75
## fc 12.86667 2.159475 3 2.737953 10.6 14.9 11.85 13.1 14.00
## ff 36.33333 7.333030 3 2.737953 28.0 41.8 33.60 39.2 40.50
## oo 36.90000 4.300000 3 2.737953 32.1 40.4 35.15 38.2 39.30
##
## Alpha: 0.05 ; DF Error: 8
## Critical Value of Studentized Range: 4.52881
##
## Minimum Significant Difference: 12.39967
##
## Treatments with the same letter are not significantly different.
##
##      yield groups
## oo 36.90000    a
## ff 36.33333   ab
## cc 24.40000   bc
## fc 12.86667    c
```

```
plot(out)
```



Il y a aussi la fonction `emmeans` du package du même nom qui permettent de faire des comparaisons de moyennes ajustées.

```
library(emmeans )
emmeans (fit, pairwise~virus, adjust="tukey")
```

Il y a aussi la fonction TukeyHSD mais qui prend un modèle ajusté avec la fonction aov et non lm.

```
fit_bis <- aov(yield ~ virus, data=sweetpotato)
TukeyHSD(fit_bis)
plot(TukeyHSD(fit_bis))
```

Plans à un facteur étudié avec contrôle de l'erreur

Lorsque des sources d'hétérogénéité sont constatées sur le terrain (différences de fertilité, de luminosité, présence d'un cours d'eau ...), on regroupe les parcelles (unités expérimentales) en différents blocs. Le but de cette opération est d'obtenir une hétérogénéité aussi faible que possible au sein de chaque bloc, et ainsi de contrôler l'hétérogénéité en prenant en compte ces blocs au moment de la planification et de l'analyse statistique. L'exemple classique illustratif du plan en blocs est donné par la présence d'une source d'hétérogénéité unidirectionnelle (par exemple, un gradient d'humidité de la parcelle en fonction de sa localisation par rapport à une rivière). Les blocs sont alors formés le long de cette direction, c'est-à-dire allongés perpendiculairement à celle-ci.

Les blocs constituent un facteur de contrôle.

Si on ne tient pas compte des blocs dans l'analyse (analyse comme pour un plan en randomisation totale)

- les erreurs expérimentales/residuelles proviennent de toutes les sources de variabilité qui affectent les différents blocs
- l'erreur sur la moyenne des différences entre deux traitements provient de la variabilité inter- et intra-bloc.

Si on tient compte des blocs dans l'analyse :

- les erreurs expérimentales dans chacun des blocs ne proviennent que des sources de variabilité qui affectent les parcelles de ce bloc
- l'erreur sur la moyenne des différences entre deux traitements calculées pour chaque expérience ne provient que de la variabilité intra-expérience
- les variations entre blocs ne contribuent pas à l'erreur résiduelle

Il y a donc un réel intérêt à faire des blocs du point de vue statistique car cela va diminuer l'erreur résiduelle est donc augmenter la puissance du test pour déterminer s'il y a un effet du traitement.

On parle de plans en blocs complets lorsque tous les traitements sont présents au moins une fois dans chaque bloc.

On parle de plans en blocs incomplets lorsque les blocs sont de taille inférieure au nombre de traitements. Tous les traitements ne peuvent donc pas être présents dans chacun des blocs.

Plan en blocs complets à un facteur de contrôle

A l'intérieur de chaque bloc, chaque traitement est représenté une fois et une seule.

A l'intérieur de chaque bloc, l'association traitement-unité expérimentale s'effectue au hasard.

Générer le plan d'expérimentation : Imaginons que l'on souhaite faire une expérimentation de 6 traitements sur une parcelle ayant un gradient d'hétérogénéité verticale. Après réflexion, on décide de découper

la parcelle en 3 bandes horizontales qui formeront donc 3 blocs. L'objectif est d'avoir des conditions environnementales qui seront les plus homogènes à l'intérieur de chaque bloc (faible variance intra). Une seule répétition des traitements sera faite à l'intérieur des blocs (on peut en faire plus si on le souhaite).

Pour générer le plan d'expérimentation, on va alors utiliser la fonction `design.rcbd` de la librairie `agricolae` (`rcbd` pour "randomized complete block design").

```
library (agricolae)

Trt <- c("T1", "T2", "T3", "T4", "T5", "T6")
nb_bloc <- 3
nb_rep_per_bloc <- 1
design_rcbd <- design.rcbd(trt = Trt, r = nb_bloc*nb_rep_per_bloc, seed=13)$book

design_rcbd
```

```
##      plots block Trt
## 1    101      1  T6
## 2    102      1  T3
## 3    103      1  T1
## 4    104      1  T4
## 5    105      1  T5
## 6    106      1  T2
## 7    201      2  T5
## 8    202      2  T1
## 9    203      2  T2
## 10   204      2  T6
## 11   205      2  T4
## 12   206      2  T3
## 13   301      3  T1
## 14   302      3  T2
## 15   303      3  T4
## 16   304      3  T3
## 17   305      3  T6
## 18   306      3  T5
```

On peut maintenant représenter graphiquement notre plan d'expérimentation :

```
# Add Row and Col
design_rcbd <- design_rcbd %>%
  mutate(Row = block %>% as.integer) %>%
  group_by(Row) %>%
  mutate(Col = 1:n()) %>%
  ungroup()

# Plot field layout
desplot(Trt ~ Col + Row, flip = TRUE,
  text = Trt, cex = 1,
  out1 = block,
  data = design_rcbd,
  main = "randomized complete block design",
  key.cex = 1)
```

randomized complete block design



Modèle statistique et analyse :

Le facteur bloc (contrôle) peut être pris en effet fixe ou en effet aléatoire (s'il comporte un nombre de niveaux suffisant).

Ici, nous considérerons le facteur bloc comme un effet fixe tout comme le traitement.

$$y_{i,j} = \mu + \alpha_i + b_j + \varepsilon_{i,j}, \quad \varepsilon_{i,j} \sim N(0, \sigma^2)$$

avec :

- $y_{i,j}$: l'observation associée à la j ème répétition et au traitement i ,
- μ l'intercept (moyenne générale),
- α_i l'effet du traitement i ,
- b_j effet du bloc j ,
- $\varepsilon_{i,j}$ l'erreur résiduelle associée à l'observation ij .

On pourra alors tester la pertinence de :

- l'effet bloc (est-ce que $b_j = 0$ pour tout j ?)
- l'effet traitement (est-ce que $\alpha_i = 0$ pour tout i ?)

Identifiabilité : Là encore le modèle n'est pas identifiable mathématiquement. Il faut donc imposer des contraintes. On retrouve les deux types de contraintes présentés dans le cas 1 facteur :

- La contrainte de type niveau de référence (par défaut dans R) : Le premier traitement du premier bloc est pris en référence :

- $\alpha_1 = 0$,
- $b_1 = 0$,
- La contrainte de type somme : Si nous avons un plan équilibré, la moyenne empirique générale est la référence. Sinon la moyenne des moyennes empiriques des différents groupes est la référence
- $\sum_i \alpha_i = 0$,
- $\sum_j b_j = 0$,

Test des effets L'analyse de la variance s'applique aussi dans le cas où on a plusieurs facteurs (traitement + bloc). La somme des carrés totaux se décompose alors comme une somme des carrés expliqués par le facteur traitement A , des carrés expliqués par le facteur bloc B et des carrés résiduels :

$$SCT = SCE_A + SCE_B + SCR$$

Remarque : Si on ne prenait pas en compte le facteur traitement, alors la SCR du modèle sans les blocs serait égale à $SCE_B + SCR$ du modèle avec. La variance résiduelle serait alors plus élevée dans le modèle sans l'effet bloc.

On va donc tester l'effet du facteur traitement A :

- H_0 : Pour tout i , $\alpha_i = 0$ (pas d'effet de A)
- H_1 : Il existe au moins un α_i non nul (effet de A)

Et l'effet du facteur bloc B :

- H_0 : Pour tout j , $b_j = 0$ (pas d'effet de B)
- H_1 : Il existe au moins un b_j non nul (effet de B)

On peut faire ces deux tests en même temps à l'aide de la table ANOVA suivante :

Variation	SC	d.d.l.	Carrés moyens	Valeur F_{obs}	$P_{H_0}(F > F_{obs})$
Facteur A	SCE_A	$I - 1$	$CMA = \frac{SCE_A}{I-1}$	$F_A = \frac{CMA}{CMR}$	p-value A
Facteur B	SCE_B	$J - 1$	$CMB = \frac{SCE_B}{J-1}$	$F_B = \frac{CMB}{CMR}$	p-value B
Erreur	SCR	$n - I - J + 1$	$CMR = \frac{SCR}{n-I-J+1}$		
Totale	SCT	$n - 1$			

Pour tester l'effet de A on utilise :

$$F_A = \frac{CMA}{CMR} \underset{H_0}{\sim} \mathcal{F}_{(I-1), (n-I-J+1)}$$

De même, pour tester l'effet de B on utilise :

$$F_B = \frac{CMB}{CMR} \underset{H_0}{\sim} \mathcal{F}_{(J-1), (n-I-J+1)}$$

Interprétation de la p-value pour A :

- Si la p-value est inférieur à 5%, alors on rejette H_0 . Le facteur A a un effet significatif sur la variable réponse.
- Si la p-value est supérieur à 5%, alors on ne rejette pas H_0 . On ne peut rien dire sur l'effet principale de A .

Il en est de même pour l'interprétation de la p-value du facteur bloc B .

Il faut ensuite faire un diagnostic pour vérifier les hypothèses du modèle.

Exemple : Les données haricots

Ces données sont issues d'une expérimentation au champ visant à tester l'effet de la fertilisation azotée sur la productivité de haricots à grains. Effectivement, plusieurs références font état de la faible capacité fixatrice du haricot qui ne lui permettrait pas de couvrir intégralement ces besoins en azote. On compare la production de matière sèche totale, mesurée à maturité sur des haricots fertilisés (N) ou non fertilisés (0N), et ce pour quatre génotypes.

Dans cet exemple on n'étudiera pas l'effet de la fertilisation, uniquement le génotype.

```
haricots <- read.table("data/haricots.txt", header=TRUE, sep="\t", dec=".")
```

```
haricots = haricots %>% filter(DAS == 98) %>%  
  select(DM_TOT, Bloc, Geno)
```

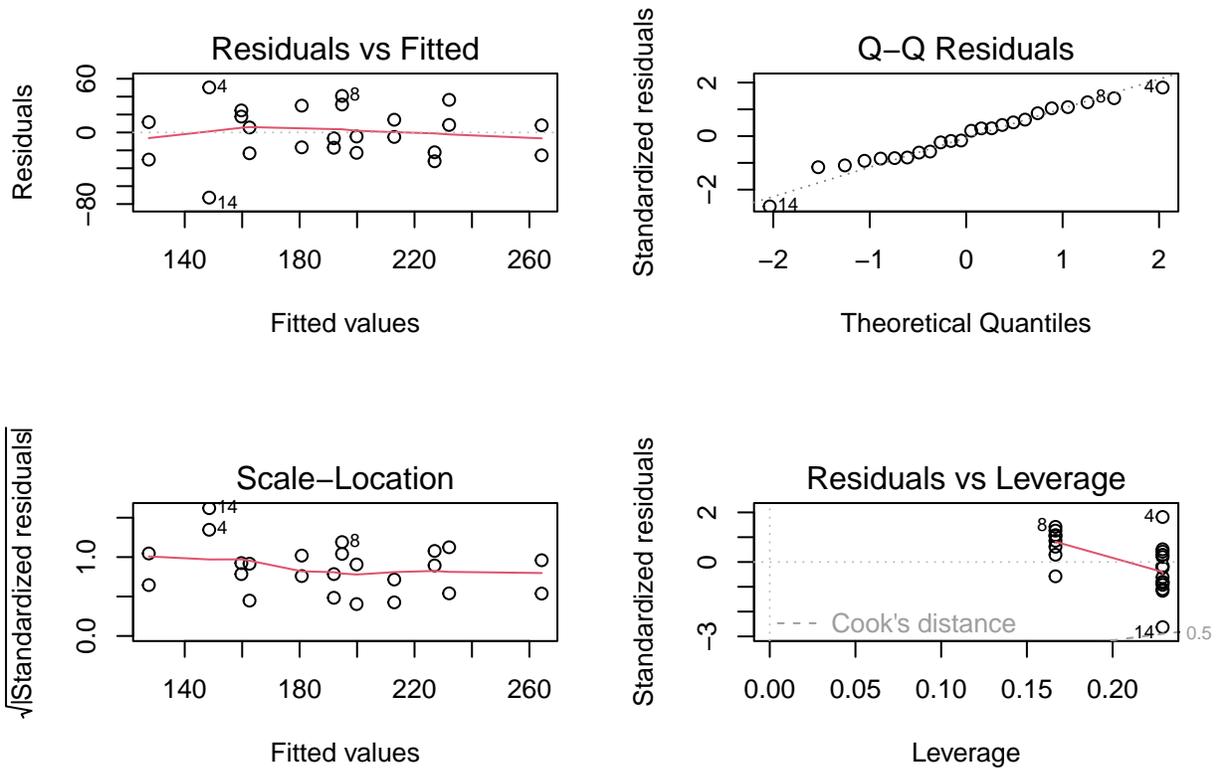
```
str(haricots)
```

```
## 'data.frame': 24 obs. of 3 variables:  
## $ DM_TOT: num 168 139 177 199 211 ...  
## $ Bloc : int 1 1 1 1 2 2 2 2 3 3 ...  
## $ Geno : chr "SER16" "SER22" "BAT 477" "DOR 364" ...
```

```
fit = lm(DM_TOT ~Geno + Bloc, data=haricots)  
anova(fit)
```

```
## Analysis of Variance Table  
##  
## Response: DM_TOT  
##          Df Sum Sq Mean Sq F value    Pr(>F)  
## Geno      3  16685  5561.7    5.585 0.0064063 **  
## Bloc      1  16606 16605.9   16.675 0.0006331 ***  
## Residuals 19  18921   995.8  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow = c(2, 2))
plot(fit)
```

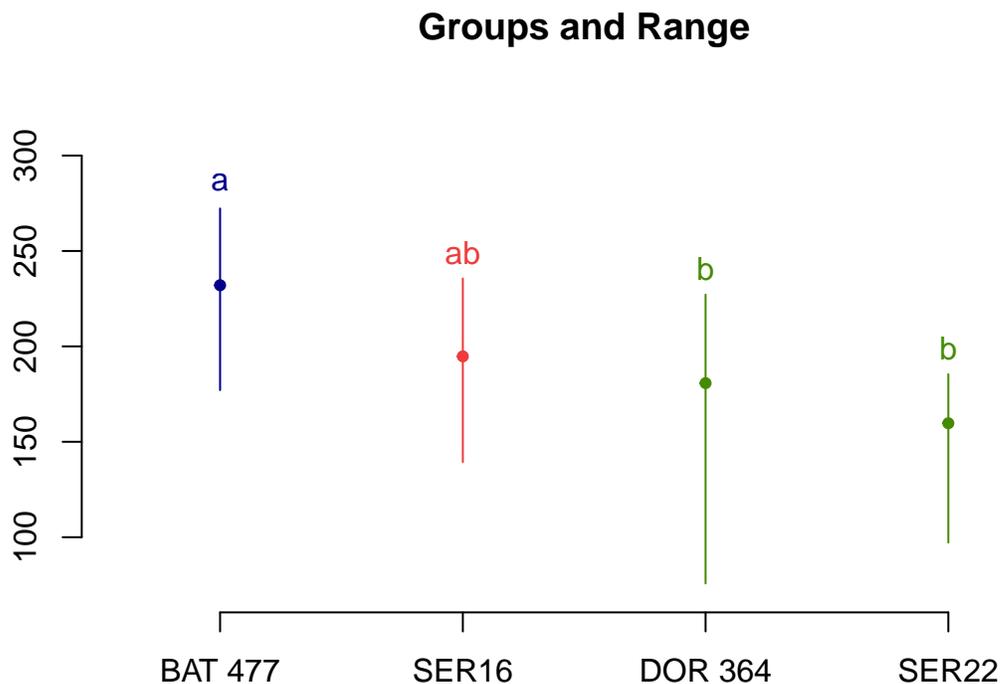


```
out = HSD.test(fit, "Geno", group=TRUE, console=TRUE,
              main="")
```

```
##
## Study:
##
## HSD Test for DM_TOT
##
## Mean Square Error: 995.8309
##
## Geno, means
##
##          DM_TOT      std r      se      Min      Max      Q25      Q50      Q75
## BAT 477 232.0542 38.55690 6 12.88301 177.070 272.270 206.1625 239.5950 261.4812
## DOR 364 180.7425 55.52501 6 12.88301  75.755 227.155 172.7900 203.3700 209.9988
## SER16 194.7758 36.13212 6 12.88301 139.310 235.595 174.7975 199.8200 220.6950
## SER22 159.7167 35.07334 6 12.88301  97.175 185.440 147.9950 176.1625 182.6062
##
## Alpha: 0.05 ; DF Error: 19
## Critical Value of Studentized Range: 3.976551
##
## Minimum Significant Difference: 51.22992
```

```
##
## Treatments with the same letter are not significantly different.
##
##          DM_TOT groups
## BAT 477 232.0542    a
## SER16  194.7758   ab
## DOR 364 180.7425    b
## SER22  159.7167    b
```

```
plot(out)
```



Plans en ligne colonne : carrés latins

Le carré latin permet d'étudier un facteur tout en considérant deux contrôles d'hétérogénéité sur les lignes et colonnes du champ. Chaque niveau du facteur étudié se retrouve une fois et une seule sur chaque ligne et chaque colonne. Le nombre de lignes est alors égal au nombre de colonne et au nombre de niveaux du facteur étudié.

Les lignes forment alors des blocs complets (tous les niveaux du traitement d'intérêt sont présents). Il en est de même pour les colonnes. Chaque croisement de ligne colonne est une parcelle élémentaire. Il y en a J^2 avec j le nombre de traitements.

Notons qu'il n'est pas possible d'étudier les interactions entre les trois facteurs car dans chacun des croisements lignes colonnes, un seul traitement est présent (il aurait fallu j^3 parcelle pour étudier les interactions).

```

library (agricolae)

Trt <- c("T1", "T2", "T3", "T4")
design_ LSD <- design.lsd(trt = Trt, serie=1, seed=42, kinds = "Super-Duper", first=TRUE, randomization=
design_ LSD

```

```

##      plots row col Trt
## 1      11   1   1  T1
## 2      12   1   2  T4
## 3      13   1   3  T3
## 4      14   1   4  T2
## 5      21   2   1  T2
## 6      22   2   2  T1
## 7      23   2   3  T4
## 8      24   2   4  T3
## 9      31   3   1  T4
## 10     32   3   2  T3
## 11     33   3   3  T2
## 12     34   3   4  T1
## 13     41   4   1  T3
## 14     42   4   2  T2
## 15     43   4   3  T1
## 16     44   4   4  T4

```

On peut maintenant représenter graphiquement notre plan d'expérimentation :

```

# Add Row and Col
design_ LSD <- design_ LSD %>%
  mutate(row = row %>% as.integer,
         col = col %>% as.integer)

# Plot field layout
desplot(Trt ~ col + row, flip = TRUE,
        text = Trt, cex = 1,
        # out1 = block,
        data = design_ LSD,
        main = "Latin Square Design",
        # key.cex = 1,
        show.key=FALSE
)

```

Latin Square Design

T1	T4	T3	T2
T2	T1	T4	T3
T4	T3	T2	T1
T3	T2	T1	T4

Modèle statistique : Anova

Ici, les blocs colonnes et lignes sont pris en effets fixes. Le modèle est alors une ANOVA :

$$y_{i,j,k} = \mu + \alpha_i + L_j + C_k + \varepsilon_{j,k}, \quad \varepsilon_{j,k} \sim N(0, \sigma^2)$$

avec :

- $y_{i,j,k}$: l'observation associée au traitement i ce trouvant sur la ligne j et colonne k
- μ l'intercept (moyenne générale)
- α_i l'effet du traitement i (contrainte d'identifiabilité : $\alpha_1 = 0$ ou $\sum_i \alpha_i = 0$)
- L_j l'effet de la ligne j (contrainte d'identifiabilité : $L_1 = 0$ ou $\sum_j L_j = 0$)
- C_k l'effet de la colonne k (contrainte d'identifiabilité : $C_1 = 0$ ou $\sum_k C_k = 0$)
- $\varepsilon_{j,k}$ l'erreur résiduelle associée à la placette de la ligne j et colonne k

On pourra alors tester la pertinence de l'effet traitement et des facteurs d'hétérogénéité ligne et colonne. Pour cela, on fait une analyse de variance (ANOVA) comme dans la section précédente où il y avait qu'un seul facteur d'hétérogénéité (RCB design).

Exemple de données : cochran.wireworms

Wireworms controlled by fumigants in a latin square

Les données proviennent du package `agridat`. Voir l'aide pour plus de détails sur les données.

W. G. Cochran (1938). Some difficulties in the statistical analysis of replicated experiments. *Empire Journal of Experimental Agriculture*, 6, 157–175.

```

data_cochran = agridat::cochran.wireworms
data_cochran = data_cochran %>%
  mutate(col = factor(col),
         row = factor(row),
         worms_sqrt = sqrt(worms+1))

```

```

# Plot field layout
desplot(trt ~ col + row, flip = TRUE,
       text = trt, cex = 1,
       # out1 = block,
       data = data_cochran,
       main = "Latin Square Design",
       # key.cex = 1,
       show.key=FALSE
)

```

```

## Warning in Ops.factor(col, row): '+' n'est pas pertinent pour des variables
## facteurs

```

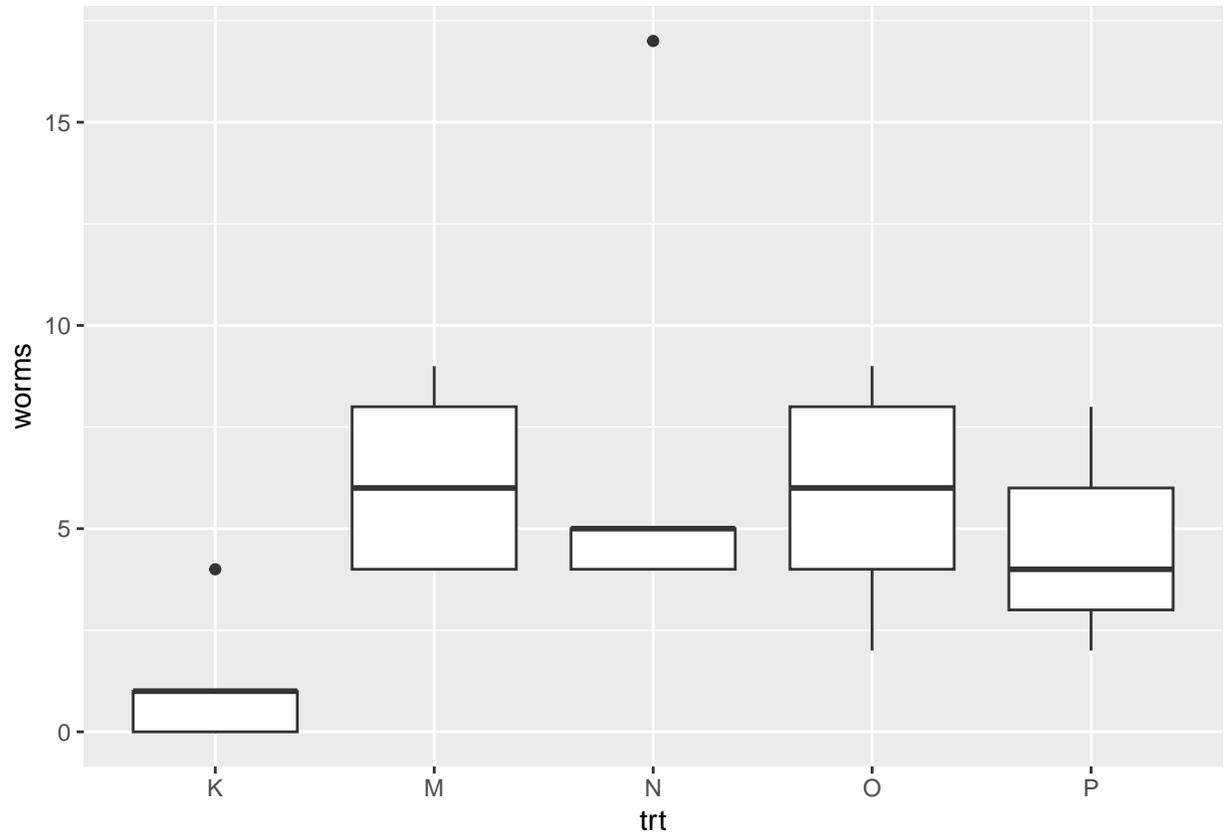
Latin Square Design

P	O	N	K	M
M	K	O	N	P
O	M	K	P	N
N	P	M	O	K
K	N	P	M	O

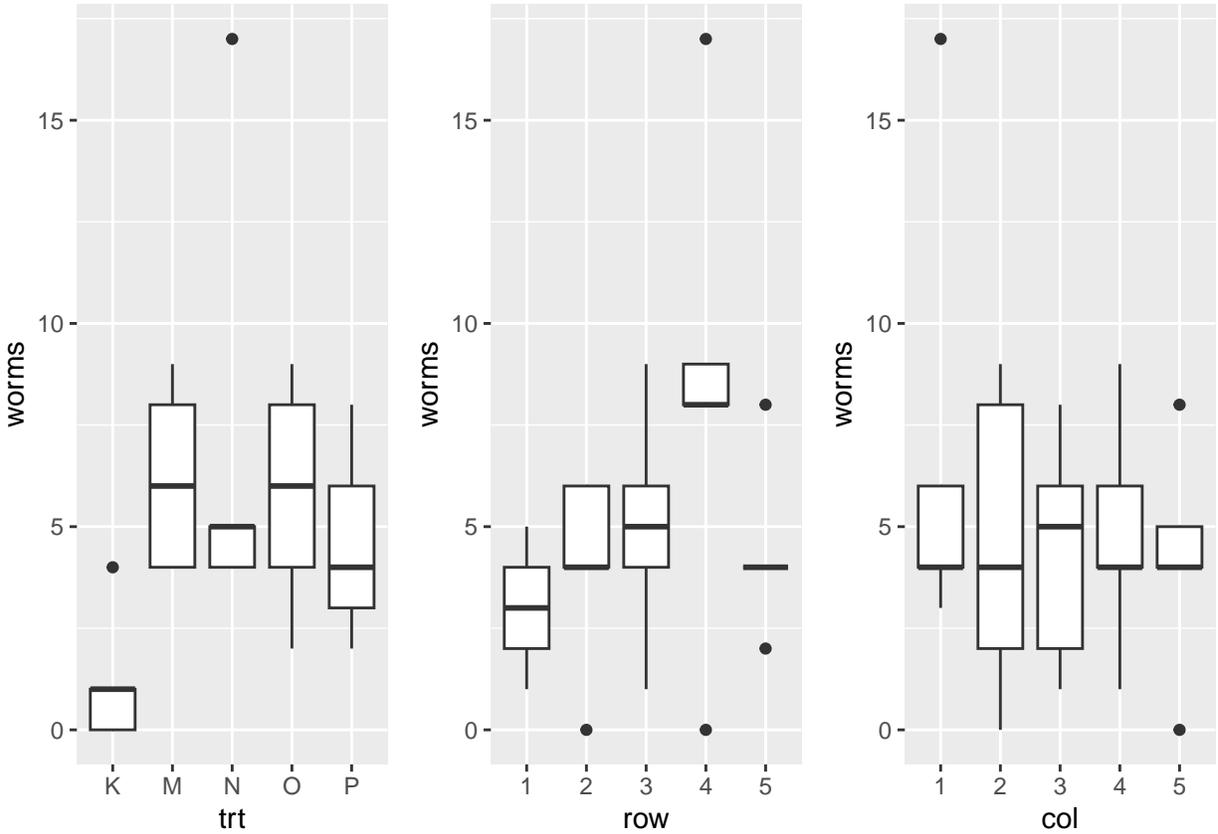
```

ggplot(data_cochran, aes(x=trt, y = worms)) + geom_boxplot()

```



```
grid.arrange(  
  ggplot(data_cochran, aes(x=trt, y = worms)) + geom_boxplot(),  
  ggplot(data_cochran, aes(x=row, y = worms)) + geom_boxplot(),  
  ggplot(data_cochran, aes(x=col, y = worms)) + geom_boxplot(),  
  nrow=1  
)
```



```
fit = lm(worms_sqrt ~ trt + col + row, data=data_cochran)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: worms_sqrt
##           Df Sum Sq Mean Sq F value Pr(>F)
## trt       4  5.7616  1.44041   4.4092 0.02014 *
## col       4  0.7787  0.19467   0.5959 0.67248
## row       4  2.3579  0.58947   1.8044 0.19285
## Residuals 12  3.9202  0.32668
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
fit = lm(worms_sqrt ~ trt + row, data=data_cochran)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: worms_sqrt
##           Df Sum Sq Mean Sq F value Pr(>F)
## trt       4  5.7616  1.44041   4.9047 0.008963 **
## row       4  2.3579  0.58947   2.0072 0.141948
## Residuals 16  4.6989  0.29368
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
fit = lm(worms_sqrt ~ trt, data=data_cochran)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: worms_sqrt
##           Df Sum Sq Mean Sq F value Pr(>F)
## trt         4  5.7616  1.44041   4.0824 0.01405 *
## Residuals  20  7.0568  0.35284
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Il ne semble pas y avoir d'hétérogénéité en fonction des lignes et colonnes. Seul le traitement est significatif.
Autre jeu de données :

```
agridat::cochran.latin
```

Plans en blocs incomplets : bie, bipe,

Dans les plans en blocs complets, lorsque le nombre de traitement est grand alors les blocs doivent être de grande taille et l'homogénéité intra-bloc n'est plus toujours respectée.

On est alors amené à constituer des blocs de taille inférieure au nombre de traitements. Un tel dispositif est appelé dispositif en blocs incomplets

Sur les animaux par exemple, "il est quasiment impossible de trouver des groupes comportant autant d'unités expérimentales que de traitements. (C'est le cas des essais où les blocs sont constitués par des animaux de même sexe provenant d'une même portée)" (itcf théorie des plans expé P113).

Dans les plans en blocs incomplets, tous les traitements ne sont pas présents dans chaque bloc.

Plans en Blocs Incomplets Equilibrés (BIE) Principe :

- Le **nombre de niveaux** J du facteur doit être au moins égal à 3 : $J \geq 3$
- Le **nombre de blocs** B considéré doit être au moins égal à J : $B \geq J$
- Soit K le **nombre d'unités expérimentales dans chaque bloc**. C'est donc le nombre de niveaux du facteur qu'on observera dans chaque bloc. Comme il est nécessaire d'observer au moins deux niveaux distincts du facteur dans chaque bloc (sinon, il y aurait confusion des effets du facteur et des effets des blocs), il vient : $2 \leq K < J$ (on voit ainsi pourquoi il est nécessaire d'avoir $J \geq 3$). Le nombre total d'observations est donc $n = BK$.
- Dans un plan équilibré, chaque niveau du facteur est répété un même nombre de fois. Soit R le **nombre de répétitions**. On a donc que $n = JR$. JR doit donc être un multiple de K , le nombre d'unités expérimentales dans chaque bloc.
- Chaque couple (j, j') de niveaux du facteur ($j \neq j'$) est observé dans le même nombre λ de blocs (condition nécessaire pour l'équilibre du plan).

Chaque traitement est répété sur R unités expérimentales. Parmi les blocs d'un BIE, R contiennent le traitement i . Donc dans ces R blocs, il reste $R(K - 1)$ unités expérimentales disponibles. Or, on sait que chaque traitement se trouve simultanément dans un même bloc avec chacun des autres traitement λ fois. Il faut donc que, dans ces $R(K - 1)$ unités expérimentales, on trouve λ fois les $(J - 1)$ autres traitements que celui i déjà mis en place. D'où la relation :

$$\lambda(J - 1) = R(K - 1)$$

Ainsi, $BK = JR$ et $\lambda = \frac{R(K-1)}{J-1}$ sont deux conditions nécessaires et λ doit être entier (comme tous les autres paramètres).

Liste des caractéristiques des BIE pour $J \leq 7$ ($L = \lambda$) :

J	B	K	R	L	n
3	3	2	2	1	6
4	4	3	3	2	12
4	6	2	3	1	12
5	5	4	4	3	20
5	10	2	4	1	20
5	10	3	6	3	30
6	6	5	5	4	30
6	10	3	5	2	30
6	15	2	5	1	30
6	15	4	10	6	60
6	20	3	10	4	60
7	7	3	3	1	21
7	7	4	4	2	28
7	7	6	6	5	42
7	21	2	6	1	42

source : <https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-modmixt4-plans.pdf>

Coefficient d'efficacité :

$$E = \frac{J\lambda}{RK}$$

K étant inférieur à J , ce coefficient est toujours inférieur à 1. Ainsi un BIE est toujours moins efficace qu'un plan en bloc complet ($E = 1$) à nombre de répétition par traitement égale. On peut dire aussi qu'un BIE à 5 répétitions avec un E de 0.80 est équivalent à un bloc complet avec 4 répétitions, pour une même variance résiduelle, c.a.d. à erreur expérimentale identique.

Réalisé un tel BIE suppose que la perte de puissance théorique par rapport à un dispositif en blocs complets $(1-E)$ sera compensée par une diminution au moins équivalente de l'erreur expérimentale grâce à un contrôle d'hétérogénéité plus fin. (Source G. Philippeau ITCF)

Générer le plan d'expérimentation :

```
library(agricolae)
```

```

trt = c("A","B","C","D") # J=4
K = 3 # nb unité expérimentale dans chaque bloc
B = 4 # nb bloc
R = 3 # nb répétition car BK=JR
# L (lambda) = 3*(3-1)/(4-1) = 2

BIE <- design.bib(trt, k=K, r=R, serie=2, seed =41, kinds = "Super-Duper")$book

```

```

##
## Parameters BIB
## =====
## Lambda      : 2
## treatmeans  : 4
## Block size  : 3
## Blocks      : 4
## Replication: 3
##
## Efficiency factor 0.8888889
##
## <<< Book >>>

```

```

BIE$Col = rep(1:R, B)
BIE$Row = 1
BIE

```

```

##   plots block trt Col Row
## 1   101     1  D   1   1
## 2   102     1  C   2   1
## 3   103     1  A   3   1
## 4   201     2  A   1   1
## 5   202     2  D   2   1
## 6   203     2  B   3   1
## 7   301     3  A   1   1
## 8   302     3  B   2   1
## 9   303     3  C   3   1
## 10  401     4  C   1   1
## 11  402     4  B   2   1
## 12  403     4  D   3   1

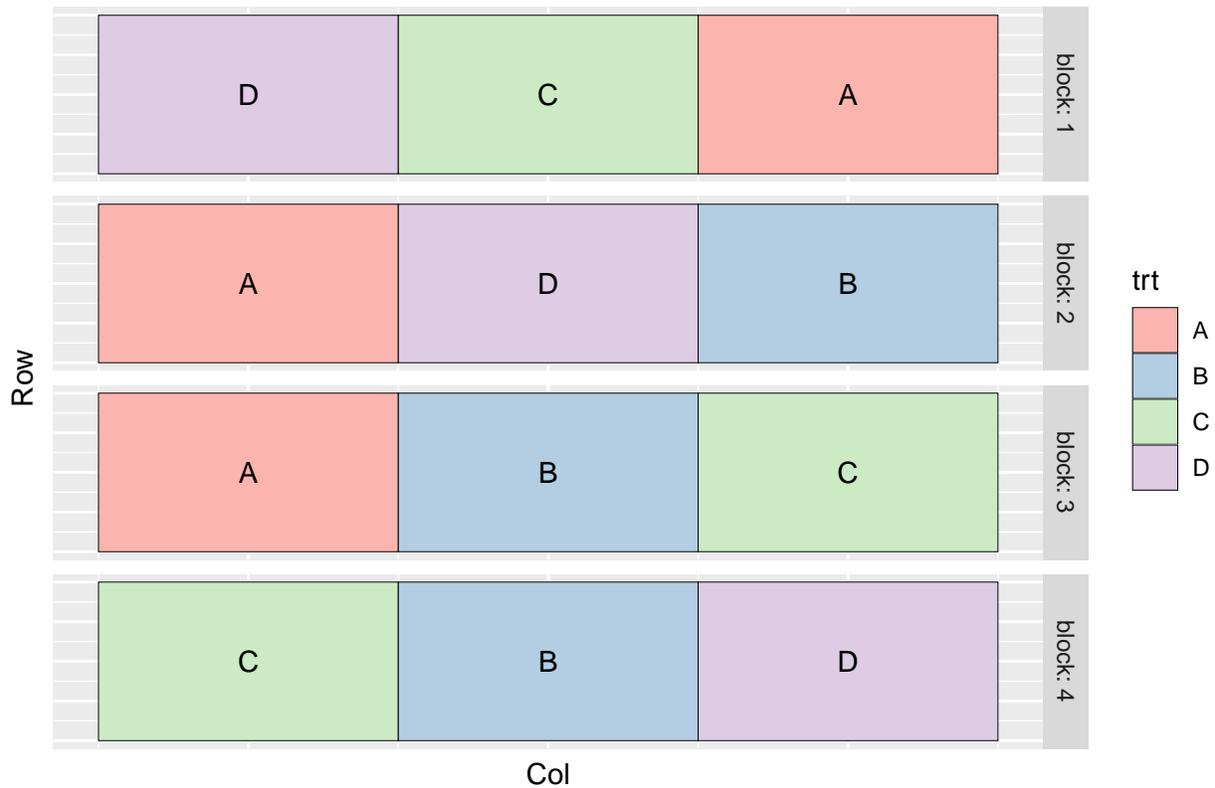
```

```

ggplot(BIE, aes(x=Col, y=Row)) +
  facet_grid(block~., labeller = label_both) +
  geom_tile(aes(fill=trt), color = "black") +
  scale_fill_brewer(palette="Pastel1") +
  # geom_tileborder(aes(group=1, grp=trt), lwd=1.5) +
  geom_text(aes(label = trt), color="black", size = 4) +
  # scale_color_brewer(palette="RdGy")+
  ggtitle("BIB design") +
  theme(axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank())
)

```

BIB design



Analyse :

L'analyse se fait au travers du modèle linéaire mixte suivant :

$$Y_{bj} = \mu + trt_j + B_b + \varepsilon_{bj}, \quad B_b \sim (iid)N(0, \sigma_B^2), \quad \varepsilon_{bj} \sim (iid)N(0, \sigma_e^2)$$

- Soit b l'indice du bloc
- Soit j l'indice du traitement (niveau du facteur)
- L'effet du bloc est aléatoire

L'effet aléatoire bloc nécessaire car il permet de capturer la variabilité inter-bloc et d'introduire des corrélations entre les observations issues d'un même bloc. La variance résiduelle quant à elle capture la variabilité intra-bloc. Cela permet ainsi d'introduire des pondérations intra-bloc et inter-blocs dans l'estimation des effets des traitements.

La encore on utilisera une analyse de la variance ANOVA pour tester l'effet du traitement.

Exemples :

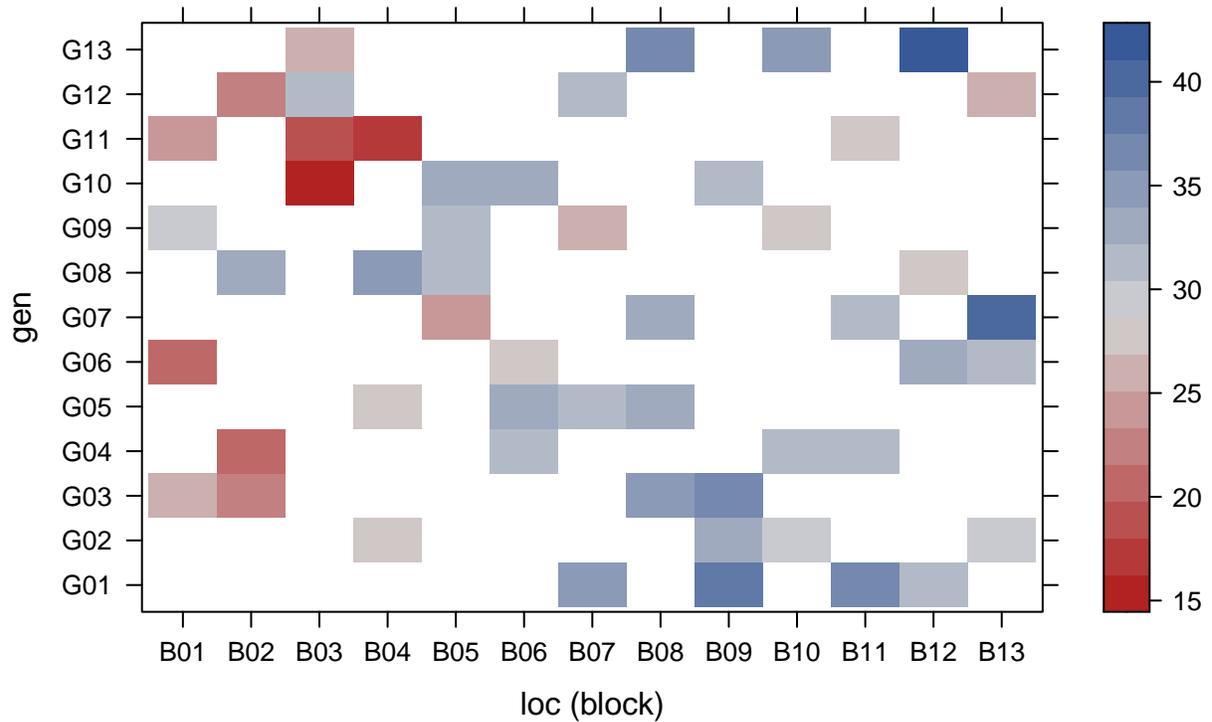
Multi-environment trial of corn, balanced incomplete block design

```
data("cochran.bib")
dat <- cochran.bib

libs(lattice)
redblue <- colorRampPalette(c("firebrick", "lightgray", "#375997"))
levelplot(yield~loc*gen, dat,
```

```
col.regions=redblue,
xlab="loc (block)", main="cochran.bib - incomplete blocks")
```

cochran.bib – incomplete blocks



```
rowSums(as.matrix(with(dat, table(gen,loc)))) # nb répétitions
```

```
## G01 G02 G03 G04 G05 G06 G07 G08 G09 G10 G11 G12 G13
## 4 4 4 4 4 4 4 4 4 4 4 4 4
```

```
colSums(as.matrix(with(dat, table(gen,loc)))) # nb unités expérimentales
```

```
## B01 B02 B03 B04 B05 B06 B07 B08 B09 B10 B11 B12 B13
## 4 4 4 4 4 4 4 4 4 4 4 4 4
```

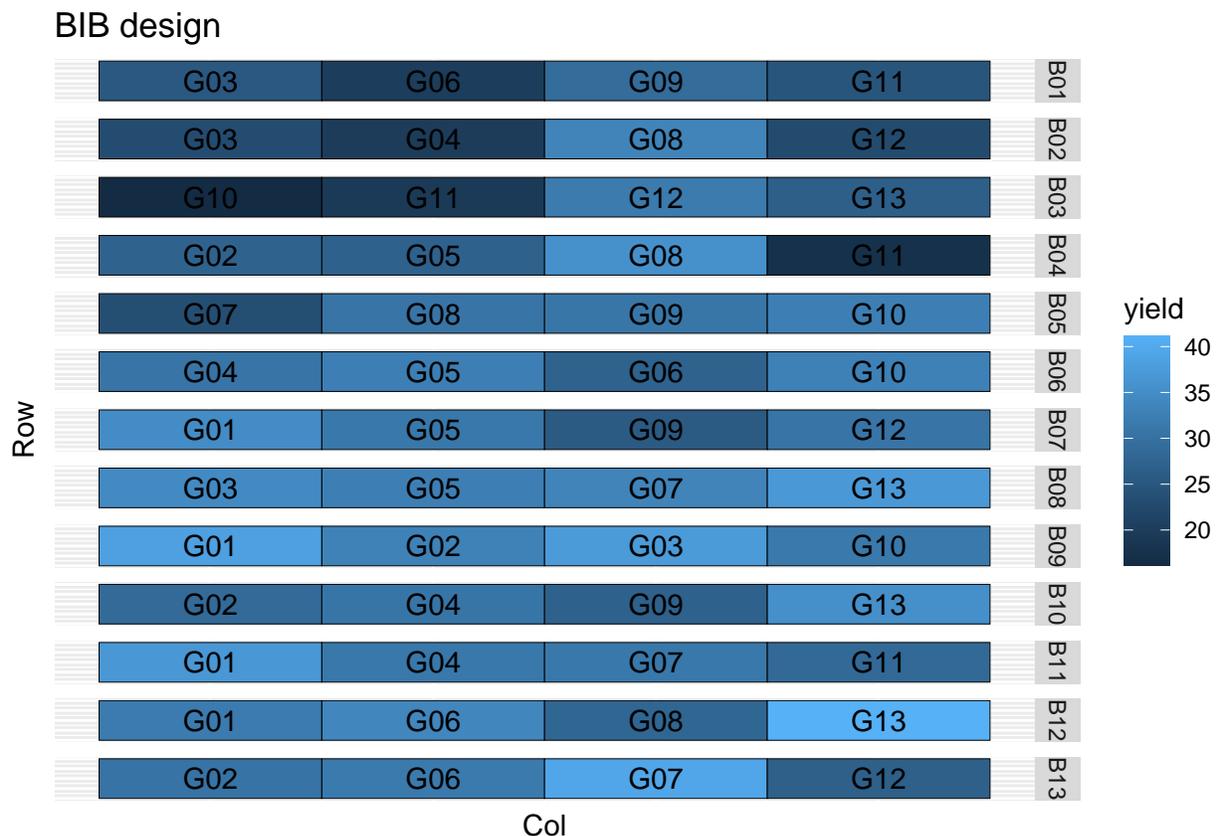
```
J = 13 # nb niveaux du facteur
K = 4 # nb unité expérimentale dans chaque bloc
B = 13 # nb bloc
R = 4 # nb répétition car BK=JR
L = R*(K-1)/(J-1); L
```

```
## [1] 1
```

```
dat$Col = rep(1:R, B)
dat$Row = 1
head(dat)
```

```
##   loc gen yield Col Row
## 1 B01 G03 25.3   1   1
## 2 B01 G06 19.9   2   1
## 3 B01 G09 29.0   3   1
## 4 B01 G11 24.6   4   1
## 5 B02 G03 23.0   1   1
## 6 B02 G04 19.8   2   1
```

```
ggplot(dat, aes(x=Col, y=Row)) +
  facet_grid(loc~.) + # , labeller = label_both
  geom_tile(aes(fill=yield), color = "black") +
  geom_text(aes(label = gen), color="black", size = 4) +
  # scale_color_brewer(palette="RdGy")+
  ggtitle("BIB design") +
  theme(axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank())
)
```



```

libs(nlme)
fit = lme(yield ~ -1 + gen, data=dat, random=~1|loc)
anova(fit)

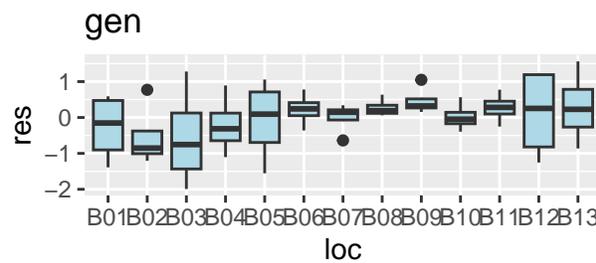
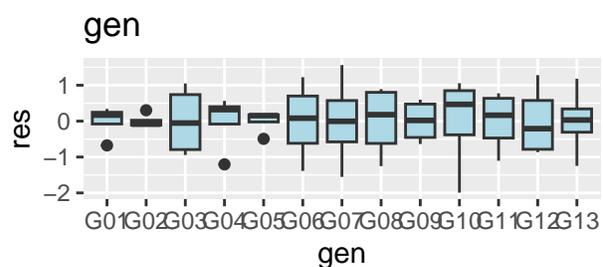
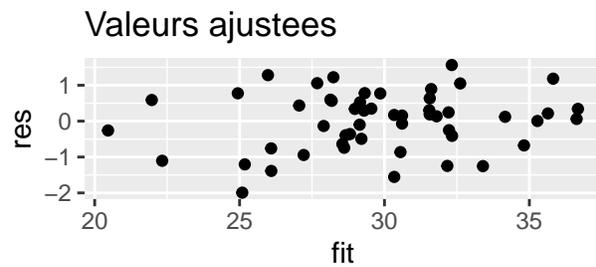
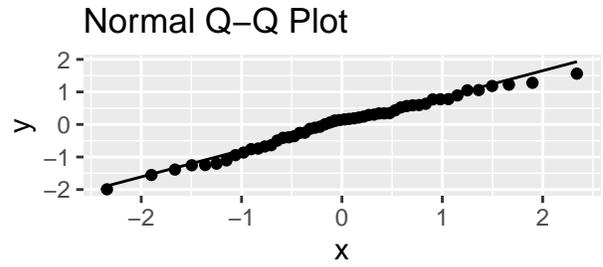
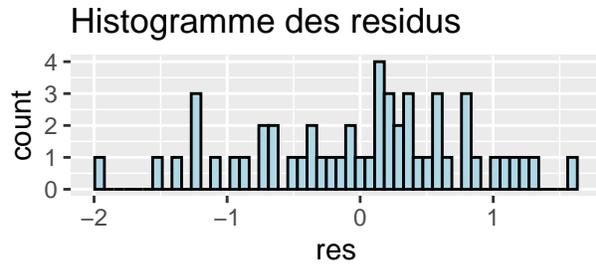
##      numDF denDF  F-value p-value
## gen      13    27 81.89795 <.0001

res <- residuals(fit, type="normalized")
fitted <- fitted(fit)

resfit <- bind_cols(res = res,
                    fit = fitted,
                    gen = dat$gen,
                    loc = dat$loc)

grid.arrange(ncol = 2,
             ggplot(resfit, aes(x = res)) +
               geom_histogram(bins = 50,
                              fill = "lightblue", color = "black")+
               labs(title = "Histogramme des residus"),
             #
             ggplot(resfit, aes(sample = res)) +
               stat_qq() +
               stat_qq_line()+
               labs(title = "Normal Q-Q Plot"),
             #
             ggplot(resfit, aes(y = res, x = fit)) +
               geom_point()+
               labs(title = "Valeurs ajustees"),
             #
             ggplot(resfit, aes(y = res, x = gen)) +
               geom_boxplot(fill = "lightblue")+
               labs(title = "gen"),
             #
             ggplot(resfit, aes(y = res, x = loc)) +
               geom_boxplot(fill = "lightblue")+
               labs(title = "gen"))

```



```
summary(fit)
```

```
## Linear mixed-effects model fit by REML
##   Data: dat
##       AIC      BIC    logLik
## 283.6421 308.5955 -126.8211
##
## Random effects:
## Formula: ~1 | loc
##      (Intercept) Residual
## StdDev:    2.460222 4.464753
##
## Fixed effects: yield ~ -1 + gen
##           Value Std.Error DF   t-value p-value
## genG01 34.17117   2.44466 27 13.977883    0
## genG02 29.04065   2.44466 27 11.879219    0
## genG03 30.10793   2.44466 27 12.315797    0
## genG04 28.07579   2.44466 27 11.484538    0
## genG05 30.34294   2.44466 27 12.411926    0
## genG06 27.59169   2.44466 27 11.286515    0
## genG07 30.75680   2.44466 27 12.581219    0
## genG08 32.75229   2.44466 27 13.397485    0
## genG09 28.55561   2.44466 27 11.680812    0
## genG10 28.10050   2.44466 27 11.494646    0
## genG11 23.46803   2.44466 27  9.599713    0
```

```

## genG12 28.98602 2.44466 27 11.856871 0
## genG13 35.17558 2.44466 27 14.388744 0
## Correlation:
## genG01 genG02 genG03 genG04 genG05 genG06 genG07 genG08 genG09 genG10
## genG02 0.071
## genG03 0.071 0.071
## genG04 0.071 0.071 0.071
## genG05 0.071 0.071 0.071 0.071
## genG06 0.071 0.071 0.071 0.071 0.071
## genG07 0.071 0.071 0.071 0.071 0.071 0.071
## genG08 0.071 0.071 0.071 0.071 0.071 0.071 0.071
## genG09 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071
## genG10 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071
## genG11 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071
## genG12 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071
## genG13 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071 0.071
## genG11 genG12
## genG02
## genG03
## genG04
## genG05
## genG06
## genG07
## genG08
## genG09
## genG10
## genG11
## genG12 0.071
## genG13 0.071 0.071
##
## Standardized Within-Group Residuals:
## Min Q1 Med Q3 Max
## -1.9919709 -0.5297679 0.1449764 0.5707380 1.5626260
##
## Number of Observations: 52
## Number of Groups: 13

```

autre exemple :

```
dat = agridat::weiss.incblock
```

Plan Alpha Lattice : BIE avec des groupes de répétition Les dispositifs alpha-lattice sont des dispositifs répétés qui divisent la répétition en blocs incomplets contenant une fraction du nombre total d'entrées.

Un groupe de répétition est un ensemble de blocs incomplets à l'intérieur duquel tous les traitements se retrouvent une fois. Soit C le nombre de groupes de répétitions. Il y a autant de groupes de répétitions que de répétitions de chaque traitement ($C = R$).

On peut obtenir ces plans en arrangeant les blocs.

Exemple : $J=4, K=2, B=6, R=3, L=1, C=3$

Groupe	Bloc	Parcelle 1	Parcelle 2
1	1	A	B
	2	C	D
2	3	A	C
	4	B	D
3	5	A	D
	6	B	C

Ainsi, comme pour l'analyse d'un plan en blocs complets, on va pouvoir prendre en compte un effet répétition qui va permettre d'éliminer la variabilité inter-répétition dans les résidus et donc d'augmenter la puissance de l'analyse.

Caractéristiques :

- $J = K^2$ le nombre de traitements est le carré du nombre d'unités expérimentales
- Les blocs incomplets sont arrangés en $C = R$ groupes de répétitions.

On peut alors réécrire tous les paramètres en fonction de K :

- $L = \frac{R}{K+1}$
- $B = K(K+1)$ nombre de blocs
- $E = \frac{K}{K+1}$ le coefficient d'efficacité

Génération du plan d'expérience :

```
library(agricolae)
#Example one
trt<-paste0("T", 1:30)
t <- length(trt)
# size block k
k<-3
# Blocks s
s<-t/k
# replications r
r <- 2
outdesign<- design.alpha(trt,k,r,serie=2)
```

```
##
## Alpha Design (0,1) - Serie I
##
## Parameters Alpha Design
## =====
## Treatmeans : 30
## Block size : 3
## Blocks      : 10
```

```
## Replication: 2
##
## Efficiency factor
## (E ) 0.6170213
##
## <<< Book >>>
```

```
book<-outdesign$book
print(book)
```

```
##   plots cols block trt replication
## 1   101   1     1 T24             1
## 2   102   2     1 T3              1
## 3   103   3     1 T25             1
## 4   104   1     2 T6              1
## 5   105   2     2 T22             1
## 6   106   3     2 T12             1
## 7   107   1     3 T16             1
## 8   108   2     3 T4              1
## 9   109   3     3 T18             1
## 10  110   1     4 T11             1
## 11  111   2     4 T21             1
## 12  112   3     4 T15             1
## 13  113   1     5 T5              1
## 14  114   2     5 T26             1
## 15  115   3     5 T17             1
## 16  116   1     6 T20             1
## 17  117   2     6 T30             1
## 18  118   3     6 T10             1
## 19  119   1     7 T8              1
## 20  120   2     7 T2              1
## 21  121   3     7 T14             1
## 22  122   1     8 T1              1
## 23  123   2     8 T13             1
## 24  124   3     8 T7              1
## 25  125   1     9 T27             1
## 26  126   2     9 T19             1
## 27  127   3     9 T28             1
## 28  128   1    10 T23             1
## 29  129   2    10 T9              1
## 30  130   3    10 T29             1
## 31  201   1    11 T30             2
## 32  202   2    11 T21             2
## 33  203   3    11 T18             2
## 34  204   1    12 T26             2
## 35  205   2    12 T22             2
## 36  206   3    12 T25             2
## 37  207   1    13 T6              2
## 38  208   2    13 T24             2
## 39  209   3    13 T16             2
## 40  210   1    14 T15             2
## 41  211   2    14 T20             2
## 42  212   3    14 T28             2
## 43  213   1    15 T13             2
```

```
## 44 214 2 15 T27 2
## 45 215 3 15 T11 2
## 46 216 1 16 T5 2
## 47 217 2 16 T9 2
## 48 218 3 16 T14 2
## 49 219 1 17 T10 2
## 50 220 2 17 T3 2
## 51 221 3 17 T4 2
## 52 222 1 18 T23 2
## 53 223 2 18 T1 2
## 54 224 3 18 T19 2
## 55 225 1 19 T29 2
## 56 226 2 19 T8 2
## 57 227 3 19 T7 2
## 58 228 1 20 T2 2
## 59 229 2 20 T17 2
## 60 230 3 20 T12 2
```

```
outdesign$sketch
```

```
## $rep1
##      [,1] [,2] [,3]
## [1,] "T24" "T3"  "T25"
## [2,] "T6"  "T22" "T12"
## [3,] "T16" "T4"  "T18"
## [4,] "T11" "T21" "T15"
## [5,] "T5"  "T26" "T17"
## [6,] "T20" "T30" "T10"
## [7,] "T8"  "T2"  "T14"
## [8,] "T1"  "T13" "T7"
## [9,] "T27" "T19" "T28"
## [10,] "T23" "T9"  "T29"
##
## $rep2
##      [,1] [,2] [,3]
## [1,] "T30" "T21" "T18"
## [2,] "T26" "T22" "T25"
## [3,] "T6"  "T24" "T16"
## [4,] "T15" "T20" "T28"
## [5,] "T13" "T27" "T11"
## [6,] "T5"  "T9"  "T14"
## [7,] "T10" "T3"  "T4"
## [8,] "T23" "T1"  "T19"
## [9,] "T29" "T8"  "T7"
## [10,] "T2"  "T17" "T12"
```

Modèle statistique et Analyse :

Comme dans le cas du BIE, l'analyse se fait au travers du modèle linéaire mixte. Dans un plan alpha lattice, on rajoute l'effet répétition en effet fixe :

$$Y_{bkj} = \mu + trt_j + rep_k + B_b + \varepsilon_{bkj}, \quad B_b \sim (iid)N(0, \sigma_B^2), \quad \varepsilon_{bkj} \sim (iid)N(0, \sigma_e^2)$$

- Soit b l'indice du bloc

- Soit k l'indice de la répétition
- Soit j l'indice du traitement (niveau du facteur)
- L'effet du bloc est aléatoire et est supposé suivre une loi normale.

L'effet aléatoire bloc permet de capter la variabilité inter-bloc et d'introduire des corrélations entre les observations issues d'un même bloc. La variance résiduelle, quant à elle, capte la variabilité intra-bloc. Cela permet ainsi d'introduire des pondérations intra-bloc et inter-blocs dans l'estimation des effets.

La encore on utilisera une analyse de la variance ANOVA pour tester l'effet du traitement. Attention, pour cette étape il faut utiliser l'estimation "ML3 et non"REML".

Exemple : `burgueno.alpha`

This dataset is available in the R package `agridat`. It corresponds to a field experiment with 3 reps, 4 blocks per rep, laid out as an alpha design. It is a data frame with 48 observations on the following 6 variables:

- `rep` rep factor, 3 levels
- `block` block factor, 12 levels
- `row` row ordinate
- `col` column ordinate
- `gen` genotype factor, 16 levels
- `yield` yield, numeric

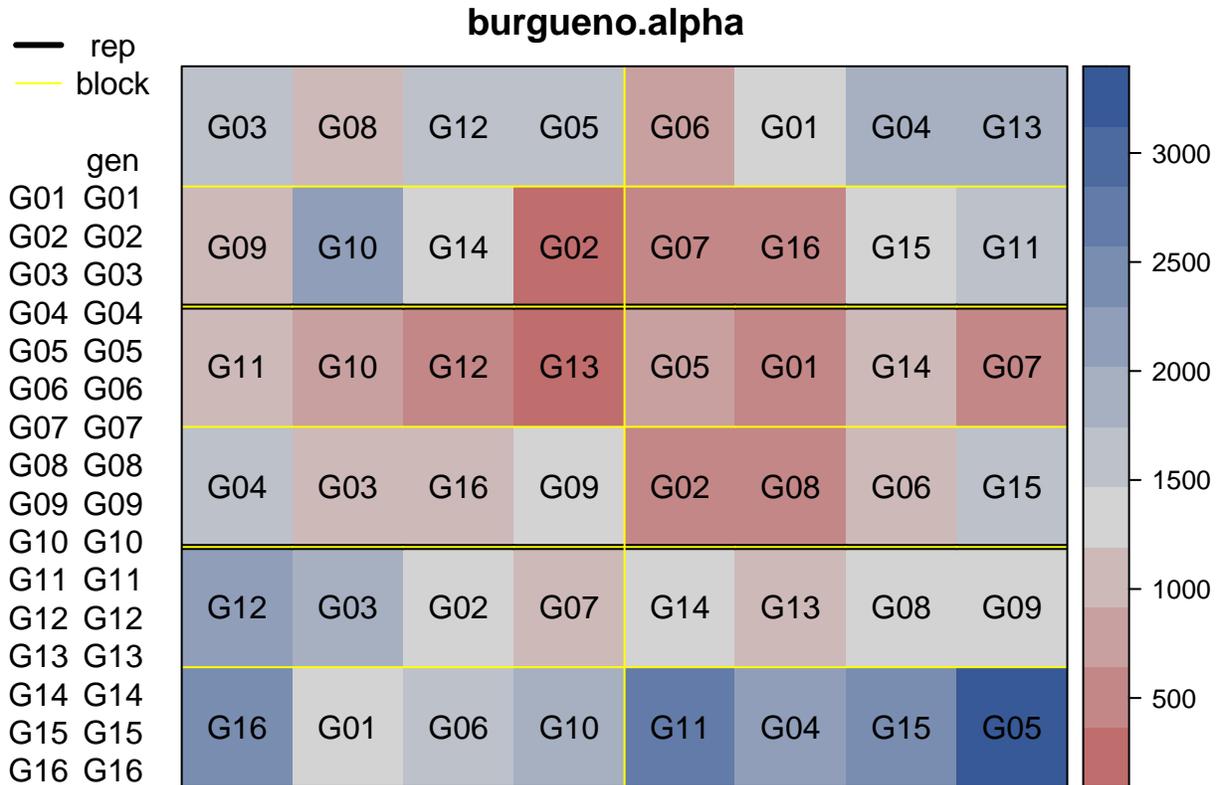
Référence : J Burgueno, A Cadena, J Crossa, M Banziger, A Gilmour, B Cullis. 2000. User's guide for spatial analysis of field variety trials using ASREML. CIMMYT.

```
library(agridat)
str(burgueno.alpha)
```

```
## 'data.frame':  48 obs. of  6 variables:
## $ rep  : Factor w/ 3 levels "R1","R2","R3": 1 1 1 1 1 1 1 1 1 1 ...
## $ block: Factor w/ 12 levels "B01","B02","B03",...: 1 1 1 1 2 2 2 2 3 3 ...
## $ row  : int  1 1 1 1 1 1 1 1 2 2 ...
## $ col  : int  1 2 3 4 5 6 7 8 1 2 ...
## $ gen  : Factor w/ 16 levels "G01","G02","G03",...: 16 1 6 10 11 4 15 5 12 3 ...
## $ yield: int  2556 1361 1567 1797 2753 2089 2531 3144 2189 1864 ...
```

```
dat <- burgueno.alpha
```

```
libs(desplot)
desplot(dat, yield~col*row,
        out1=rep, out2=block, # aspect unknown
        text=gen, cex=1, shorten="none",
        main='burgueno.alpha')
```



```
fit <- lme(yield ~ gen + rep, random=~1|block, data=burgueno.alpha, method="REML")
anova(fit)
```

```
##          numDF denDF  F-value p-value
## (Intercept)      1    21 176.82251 <.0001
## gen             15    21  2.75079  0.0166
## rep              2     9  7.68407  0.0113
```

Les deux effets sont significatifs donc on les garde dans le modèle.

```
res <- residuals(fit, type="normalized")
fitted <- fitted(fit)

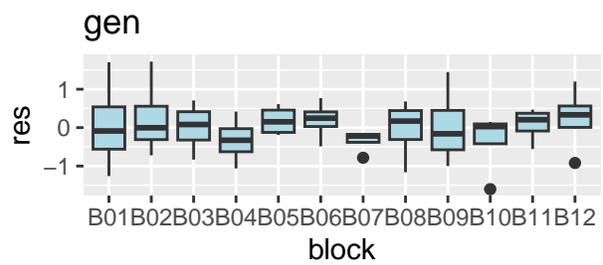
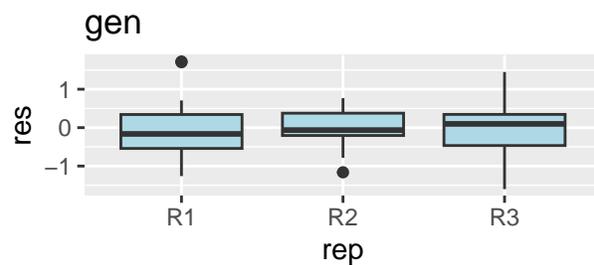
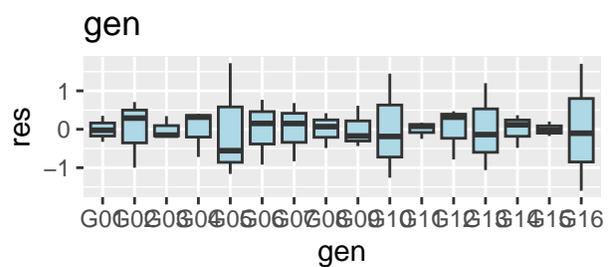
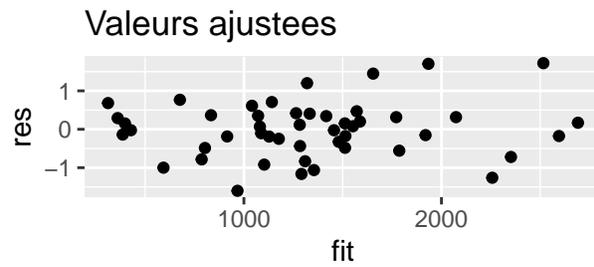
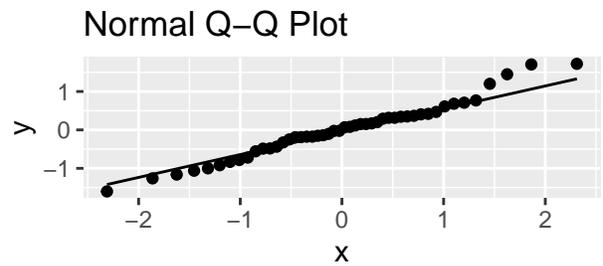
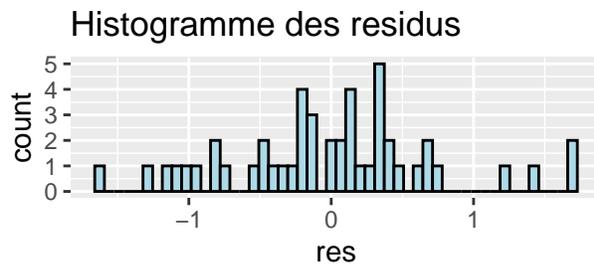
resfit <- bind_cols(res = res,
                    fit = fitted,
                    gen = burgueno.alpha$gen,
                    rep = burgueno.alpha$rep,
                    block = burgueno.alpha$block)

grid.arrange(ncol = 2,
             ggplot(resfit, aes(x = res)) +
               geom_histogram(bins = 50,
                             fill = "lightblue", color = "black")+
               labs(title = "Histogramme des residus"),
             #
```

```

ggplot(resfit, aes(sample = res)) +
  stat_qq() +
  stat_qq_line()+
  labs(title = "Normal Q-Q Plot"),
#
ggplot(resfit, aes(y = res, x = fit)) +
  geom_point()+
  labs(title = "Valeurs ajustees"),
#
ggplot(resfit, aes(y = res, x = gen)) +
  geom_boxplot(fill = "lightblue")+
  labs(title = "gen"),
#
ggplot(resfit, aes(y = res, x = rep)) +
  geom_boxplot(fill = "lightblue")+
  labs(title = "gen"),
#
ggplot(resfit, aes(y = res, x = block)) +
  geom_boxplot(fill = "lightblue")+
  labs(title = "gen")

```



```
ks.test(res, "pnorm", 0, sd(res))
```

```
##
## Exact one-sample Kolmogorov-Smirnov test
```

```
##
## data: res
## D = 0.092956, p-value = 0.7665
## alternative hypothesis: two-sided
```

```
summary(fit)
```

```
## Linear mixed-effects model fit by REML
```

```
## Data: burgueno.alpha
##      AIC      BIC    logLik
## 510.1715 538.1954 -235.0857
##
```

```
## Random effects:
```

```
## Formula: ~1 | block
##      (Intercept) Residual
## StdDev:      294.7808 364.9895
##
```

```
## Fixed effects: yield ~ gen + rep
```

	Value	Std.Error	DF	t-value	p-value
(Intercept)	1416.3116	283.3295	21	4.998815	0.0001
genG02	-284.2834	333.7360	21	-0.851821	0.4039
genG03	493.8583	335.5871	21	1.471625	0.1560
genG04	699.2937	320.9371	21	2.178912	0.0409
genG05	863.3991	323.0027	21	2.673040	0.0142
genG06	30.9743	309.6136	21	0.100042	0.9213
genG07	-115.8148	323.0027	21	-0.358557	0.7235
genG08	157.7642	333.7360	21	0.472722	0.6413
genG09	407.2421	333.5995	21	1.220752	0.2357
genG10	777.1774	320.9371	21	2.421588	0.0246
genG11	1038.4632	333.5995	21	3.112904	0.0053
genG12	648.0273	335.5871	21	1.931026	0.0671
genG13	248.1868	320.9371	21	0.773319	0.4480
genG14	405.4580	322.8616	21	1.255826	0.2230
genG15	942.4844	333.5995	21	2.825198	0.0101
genG16	453.7846	320.9371	21	1.413936	0.1720
repR2	-955.1250	245.1531	9	-3.896035	0.0036
repR3	-569.8750	245.1531	9	-2.324568	0.0451

```
## Correlation:
```

	(Intr)	genG02	genG03	genG04	genG05	genG06	genG07	genG08	genG09	genG10
genG02	-0.592									
genG03	-0.592	0.540								
genG04	-0.570	0.481	0.517							
genG05	-0.573	0.490	0.526	0.503						
genG06	-0.546	0.498	0.461	0.482	0.444					
genG07	-0.573	0.522	0.526	0.469	0.511	0.444				
genG08	-0.592	0.542	0.540	0.486	0.522	0.498	0.490			
genG09	-0.589	0.537	0.534	0.514	0.484	0.464	0.484	0.537		
genG10	-0.570	0.520	0.484	0.465	0.463	0.482	0.468	0.487	0.520	
genG11	-0.589	0.500	0.503	0.520	0.523	0.464	0.523	0.500	0.494	0.514
genG12	-0.592	0.540	0.574	0.484	0.526	0.461	0.526	0.540	0.503	0.517
genG13	-0.570	0.487	0.484	0.500	0.468	0.482	0.463	0.520	0.520	0.505
genG14	-0.570	0.523	0.487	0.463	0.500	0.444	0.500	0.523	0.556	0.503
genG15	-0.589	0.531	0.503	0.520	0.523	0.498	0.523	0.531	0.494	0.481
genG16	-0.570	0.486	0.517	0.505	0.469	0.482	0.503	0.481	0.514	0.500

```

## repR2 -0.433 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
## repR3 -0.433 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000
##      genG11 genG12 genG13 genG14 genG15 genG16 repR2
## genG02
## genG03
## genG04
## genG05
## genG06
## genG07
## genG08
## genG09
## genG10
## genG11
## genG12 0.534
## genG13 0.514 0.517
## genG14 0.478 0.487 0.503
## genG15 0.569 0.503 0.481 0.478
## genG16 0.520 0.484 0.465 0.463 0.520
## repR2 0.000 0.000 0.000 0.000 0.000 0.000
## repR3 0.000 0.000 0.000 0.000 0.000 0.000 0.500
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -1.6015675 -0.4479847 0.0221402 0.3549793 1.7202469
##
## Number of Observations: 48
## Number of Groups: 12

```

Plans factoriels

Jusqu'à présent, les traitements faisaient référence à un seul facteur (une seule variable). On peut aussi vouloir tester les effets de plusieurs facteurs simultanément sur une variable réponse ainsi que leurs interactions. Ainsi, on considérera que la combinaison de tous les niveaux de tous les facteurs forme l'ensemble des traitements. Autrement dit, un traitement est la combinaison d'un niveau de chacun des facteurs. On appelle cela les **combinaisons factoriels**.

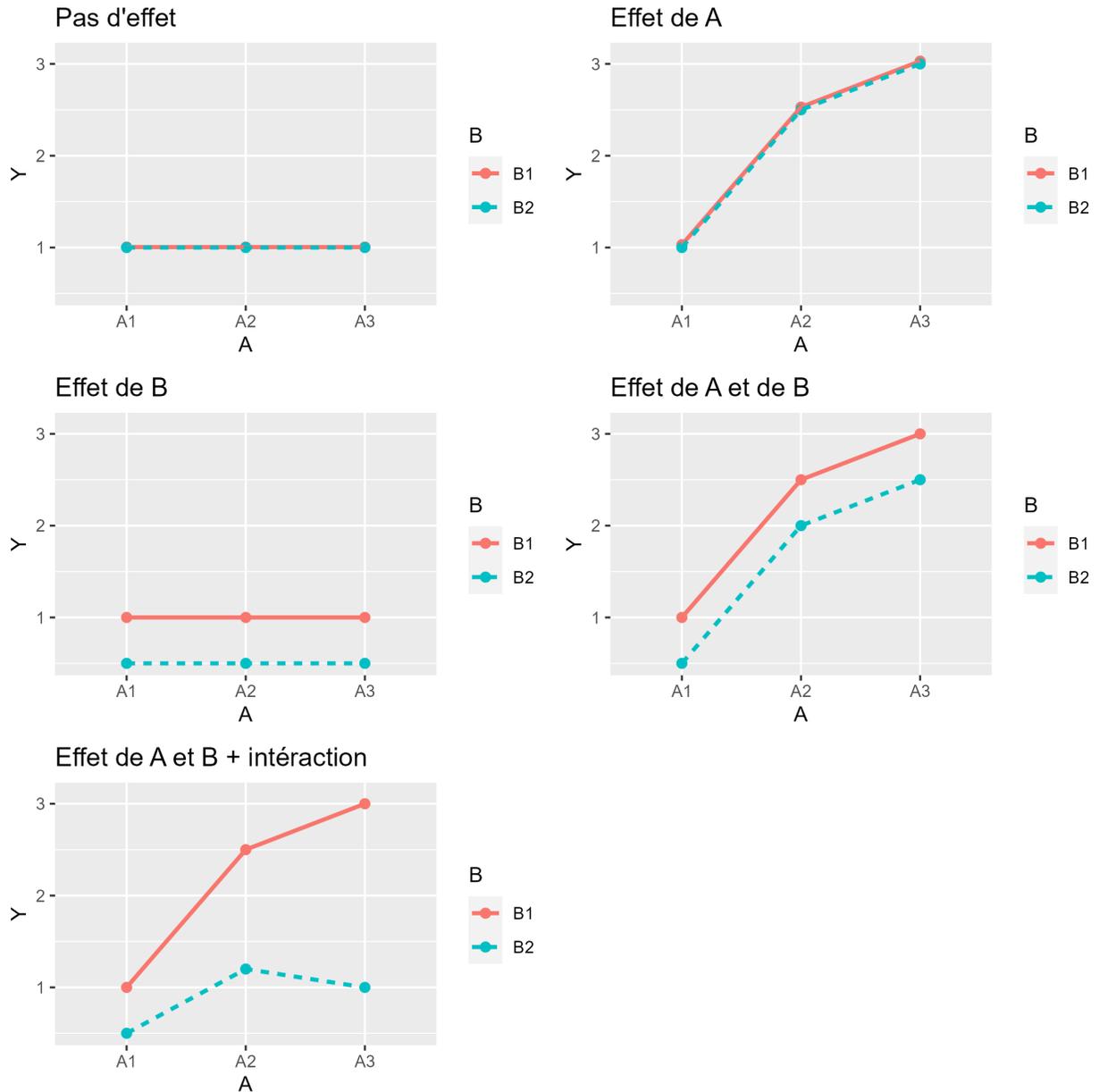
Là encore, on va retrouver les plans en randomisation totale ou en bloc. Nous en verrons également de nouveaux.

Définitions des effets principaux et des interactions : Prenons un exemple avec 2 facteurs A (3 niveaux) et B (2 niveaux).

- Effets principaux : l'effet principal de A est l'effet des niveaux du facteur A sur Y quelle que soit la valeur prise par le facteur B et vis-versa.
- Effets d'interaction : l'interaction entre A et B signifie que le facteur A va agir différemment sur Y en fonction de la valeur prise par le facteur B et vis-versa. Ces effets d'interaction ne sont estimables que si plusieurs répétitions de chaque combinaison sont réalisées.

Voici des représentations visuelles (simulations) pour mieux comprendre. Sur les 4 premières figures il n'y a pas d'interaction. Uniquement des effets principaux (parfois nuls). On le voit car les courbes sont bien parallèles.

Sur le dernier graph, on peut y voir une interaction (en plus des effets principaux) car les courbes des effets du facteur *A* sont différentes en fonction des niveaux de *B* (courbes non-parallèles). En pratique, lorsque l'on réalisera ce type de graph sur des données réelles, il ne sera pas toujours évident de voir une interaction ou non. Les modèles statistiques sont là pour nous aider à les identifier !



Plans factoriels en randomisation totale

Plans factoriels en randomisation totale sans répétition

Un plan factoriel sans répétition signifie que chaque traitement (combinaison factorielle) n'est présente qu'une seule fois dans l'expérimentation. On ne pourra donc pas estimer l'effet d'interaction entre les facteurs (pour cela il faut plus qu'une observation par combinaison).

Toutefois, les niveaux de chacun des facteurs jouent le rôle de répétitions pour chaque niveau des autres facteurs. On pourra donc estimer les effets principaux.

La génération d'un tel plan peut se faire avec la fonction `design.crd` du package `agricolae`. On donnera à l'argument `trt` toutes les combinaisons des deux facteurs :

```
A = c("a", "b", "c", "d")
B = 1:3
trt = expand.grid(A = A, B=B)
trt$Traitement = paste0(trt$A, "_", trt$B)
trt
```

```
##      A B Traitement
## 1   a 1         a_1
## 2   b 1         b_1
## 3   c 1         c_1
## 4   d 1         d_1
## 5   a 2         a_2
## 6   b 2         b_2
## 7   c 2         c_2
## 8   d 2         d_2
## 9   a 3         a_3
## 10  b 3         b_3
## 11  c 3         c_3
## 12  d 3         d_3
```

```
library(agricolae)
```

```
# La fonction design.crd permet de générer le plan en randomisation totale. Le résultat renvoyé est une
```

```
Traitement = trt$Traitement
design_crd <- design.crd(trt = Traitement, r = 1, serie = 0)$book
```

```
# on change le nom "plot" en "Parcelle"
```

```
names(design_crd)[1] <- "Parcelle"
```

```
# on change "r" en "Répétition"
```

```
names(design_crd)[2] <- "Répétition"
```

```
design_crd
```

```
##      Parcelle Répétition Traitement
## 1           1           1         b_1
## 2           2           1         d_2
## 3           3           1         d_3
## 4           4           1         c_2
## 5           5           1         b_3
## 6           6           1         b_2
```

```
## 7      7      1      d_1
## 8      8      1      a_2
## 9      9      1      c_1
## 10     10     1      c_3
## 11     11     1      a_3
## 12     12     1      a_1
```

Analyse : L'analyse se fait au travers d'un modèle linéaire. Prenons un exemple avec 2 facteurs A et B . Le modèle peut alors s'écrire :

$$Y_{i,j} = \mu + A_i + B_j + \varepsilon_{i,j}$$

avec :

- i l'indice des niveaux du facteur A
- j l'indice des niveaux du facteur B
- les $\varepsilon_{i,j}$ sont supposés indépendants et identiquement distribués suivants une loi normale $N(0, \sigma_e^2)$ avec σ_e^2 la variance résiduelle.

Ce modèle peut être mis en oeuvre avec R à l'aide des fonctions `aov` ou `lm`.

Exemple :

Clover yields in a factorial fertilizer experiment

A phosphorous by sulfur factorial experiment at Dipton in Southland, New Zealand. There were 3 reps. Plots were harvested repeatedly from Dec 1992 to Mar 1994. Yields reported are the total dry matter across all cuttings.

Sinclair AG, Risk WH, Smith LC, Morrison JD & Dodds KG (1994) Sulphur and phosphorus in balanced pasture nutrition. Proc N Z Grass Assoc, 56, 13-16.

```
library(agridat)
data(sinclair.clover)
dat <- sinclair.clover
```

```
table(dat$P, dat$S)
```

```
##
##      0 11.25 22.5 45 90
##  0   1    1    1  1  1
## 20   1    1    1  1  1
## 40   1    1    1  1  1
## 80   1    1    1  1  1
## 160  1    1    1  1  1
```

Ici, on voit bien qu'il y a qu'une seule répétition pour chaque traitement. On ne peut donc pas estimer d'effet d'interaction. On peut uniquement estimer les effets principaux.

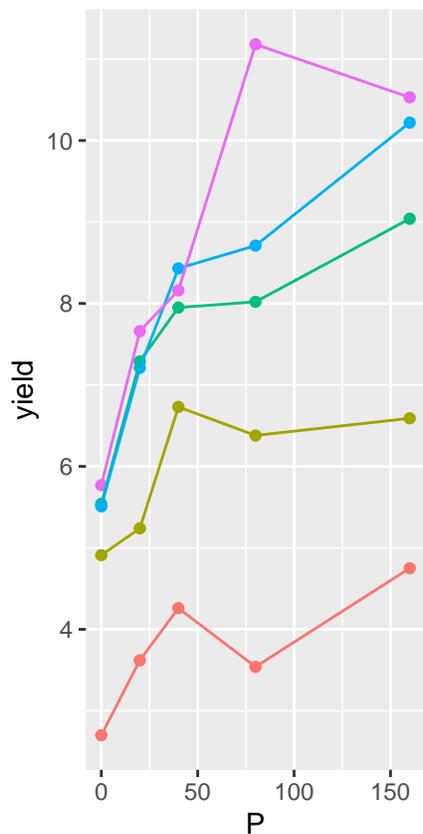
```
table(dat$P)
```

```
##
##  0 20 40 80 160
##  5  5  5  5  5
```

```
table(dat$S)
```

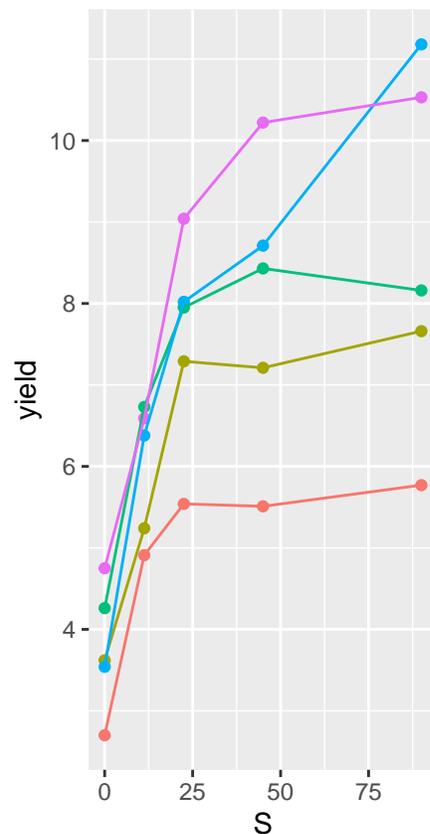
```
##  
##      0 11.25 22.5  45  90  
##      5   5   5   5   5
```

```
grid.arrange(  
  ggplot(dat, aes(x=P, y=yield, color=factor(S))) + geom_point() + geom_line(),  
  ggplot(dat, aes(x=S, y=yield, color=factor(P))) + geom_point() + geom_line(),  
  nrow=1  
)
```



factor(S)

- 0
- 11.25
- 22.5
- 45
- 90



factor(P)

- 0
- 20
- 40
- 80
- 160

Il semble y avoir un effet non-linéaire de S et de P (sqrt).

On peut considérer S et P comme des facteurs vu leurs faibles nombres de modalités (4).

```
dat$S = factor(dat$S)  
dat$P = factor(dat$P)  
  
fit=lm(yield ~ P + S, data=dat)  
anova(fit)
```

```
## Analysis of Variance Table  
##  
## Response: yield
```

```
##           Df Sum Sq Mean Sq F value    Pr(>F)
## P             4 33.662   8.4156  13.277 5.902e-05 ***
## S             4 76.868  19.2170  30.317 2.749e-07 ***
## Residuals 16 10.142   0.6339
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(fit)
```

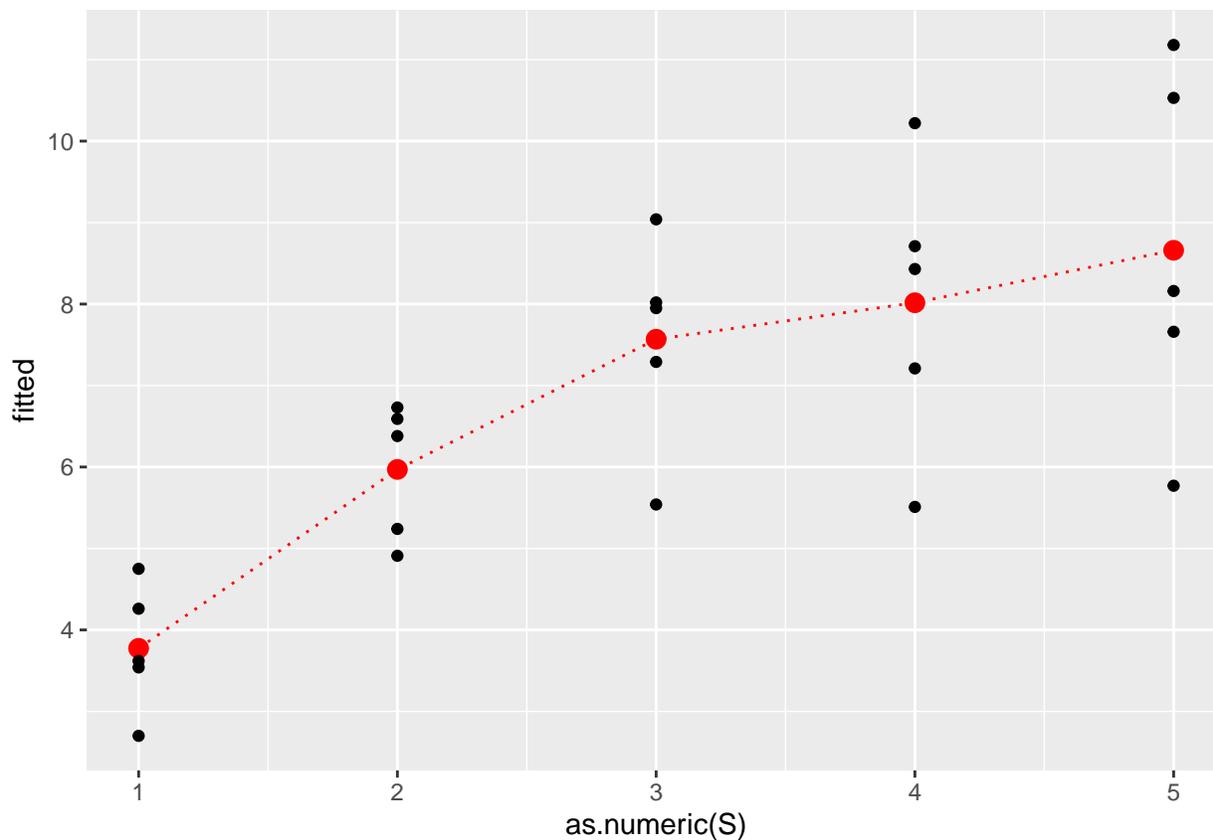
```
##
## Call:
## lm(formula = yield ~ P + S, data = dat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.0024 -0.4064 -0.0744  0.4396  1.7516
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.8624     0.4777   3.899 0.001277 **
## P20            1.3180     0.5035   2.617 0.018665 *
## P40            2.2200     0.5035   4.409 0.000439 ***
## P80            2.6800     0.5035   5.322 6.87e-05 ***
## P160           3.3400     0.5035   6.633 5.75e-06 ***
## S11.25         2.1960     0.5035   4.361 0.000485 ***
## S22.5          3.7940     0.5035   7.535 1.20e-06 ***
## S45            4.2420     0.5035   8.424 2.82e-07 ***
## S90            4.8860     0.5035   9.703 4.17e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7962 on 16 degrees of freedom
## Multiple R-squared:  0.916, Adjusted R-squared:  0.8739
## F-statistic: 21.8 on 8 and 16 DF, p-value: 3.255e-07
```

Partial Dependence Plot

```
dat$fitted=fitted(fit)

dat2 = dat %>%
  group_by(S) %>%
  summarise(fitted=mean(fitted))

ggplot(dat2, aes(x=as.numeric(S), y=fitted)) + geom_line(color="red", linetype="dotted") + geom_point(c
  geom_point(data=dat, aes(x=as.numeric(S), y=yield))
```



on peut faire pareil pour P .

Plans factoriels en randomisation totale avec répétitions

Ici, chaque traitement (combinaison des niveaux des facteurs) est répété plusieurs fois. On va donc pouvoir estimer les effets d'interaction.

La génération d'un tel plan peut se faire avec la fonction `design.crd` du package `agricolae`. On donnera à l'argument `trt` toutes les combinaisons des deux facteurs :

```
A = c("a", "b", "c", "d")
B = 1:3
trt = expand.grid(A = A, B=B)
trt$Traitement = paste0(trt$A, "_", trt$B)
trt
```

```
##   A B Traitement
## 1 a 1      a_1
## 2 b 1      b_1
## 3 c 1      c_1
## 4 d 1      d_1
## 5 a 2      a_2
## 6 b 2      b_2
## 7 c 2      c_2
## 8 d 2      d_2
## 9 a 3      a_3
```

```
## 10 b 3      b_3
## 11 c 3      c_3
## 12 d 3      d_3
```

```
library(agricolae)
```

```
# La fonction design.crd permet de générer le plan en randomisation totale. Le résultat renvoyé est une
```

```
Traitement = trt$Traitement
```

```
design_crd <- design.crd(trt = Traitement, r = 3, serie = 0)$book
```

```
# on change le nom "plot" en "Parcelle"
```

```
names(design_crd)[1] <- "Parcelle"
```

```
# on change "r" en "Répétition"
```

```
names(design_crd)[2] <- "Répétition"
```

```
design_crd
```

```
##   Parcelle Répétition Traitement
## 1         1           1         c_1
## 2         2           1         a_3
## 3         3           1         b_1
## 4         4           1         a_2
## 5         5           1         d_3
## 6         6           1         a_1
## 7         7           1         c_2
## 8         8           1         b_3
## 9         9           2         b_3
## 10        10          1         b_2
## 11        11          2         c_1
## 12        12          2         a_1
## 13        13          1         d_1
## 14        14          1         c_3
## 15        15          1         d_2
## 16        16          2         c_3
## 17        17          2         d_1
## 18        18          3         c_1
## 19        19          2         d_2
## 20        20          2         b_1
## 21        21          3         b_3
## 22        22          2         c_2
## 23        23          3         b_1
## 24        24          2         a_2
## 25        25          2         d_3
## 26        26          3         d_3
## 27        27          3         a_1
## 28        28          2         b_2
## 29        29          3         b_2
## 30        30          3         c_3
## 31        31          3         c_2
## 32        32          2         a_3
## 33        33          3         d_1
## 34        34          3         a_3
## 35        35          3         a_2
## 36        36          3         d_2
```

Analyse : L'analyse se fait au travers d'un modèle linéaire. Prenons un exemple avec 2 facteurs A et B . On note AB l'effet d'interaction. Le modèle peut alors s'écrire :

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}$$

avec :

- i l'indice des niveaux du facteur A ,
- j l'indice des niveaux du facteur B ,
- k l'indice des répétitions dans chacune des combinaisons des facteurs A et B
- y_{ijk} la variable réponse pour l'observation ijk ,
- μ l'intercept,
- α_i l'effet du i ème niveau du facteur A ,
- β_j l'effet du j ème niveau du facteur B ,
- γ_{ij} l'effet d'interaction du i ème niveau du facteur A et du j ème niveau du facteur B ,
- ε_{ijk} le résidu associé à l'observation ijk . Les résidus sont supposés indépendants et identiquement distribués suivant une loi normale $N(0, \sigma^2)$.

Identifiabilité : Là encore le modèle n'est pas identifiable mathématiquement. Il faut donc imposer des contraintes. On retrouve les deux types de contraintes présentés dans le cas 1 facteur :

- La contrainte de type niveau de référence (par défaut dans R) : le groupe des individus ayant la première modalité du facteur A et la première modalité du facteur B est le groupe de référence

- $\alpha_1 = 0$,
- $\beta_1 = 0$,
- Pour tout i , $\gamma_{i1} = 0$,
- Pour tout j , $\gamma_{1j} = 0$.

- La contrainte de type somme : Si nous avons un plan équilibré, la moyenne empirique générale est la référence. Sinon la moyenne des moyennes empiriques des différents groupes est la référence

- $\sum_i \alpha_i = 0$,
- $\sum_j \beta_j = 0$,
- Pour tout i , $\sum_j \gamma_{ij} = 0$,
- Pour tout j , $\sum_i \gamma_{ij} = 0$.

Test des effets L'analyse de la variance s'applique aussi dans le cas où on a plusieurs facteurs. La somme des carrés totaux se décompose alors comme une somme des carrés expliqués par A , des carrés expliqués par B , des carrés expliqués par l'interaction de A et de B et des carrés résiduels :

$$SCT = SCE_A + SCE_B + SCE_{AB} + SCR$$

On commence toujours par tester l'interaction :

- H_0 : Pour tout i, j , $\gamma_{ij} = 0$ (pas d'interaction, modèle additif)
- H_1 : Il existe au moins une interaction non nulle (interaction)

On construit alors la table ANOVA suivante :

Variation	SC	d.d.l.	Carrés moyens	Valeur F_{obs}	$P_{H_0}(F > F_{obs})$
Facteur A	SCE_A	$I - 1$	$CMA = \frac{SCE_A}{I-1}$	$F_A = \frac{CMA}{CMR}$	p-value A
Facteur B	SCE_B	$J - 1$	$CMB = \frac{SCE_B}{J-1}$	$F_B = \frac{CMB}{CMR}$	p-value B
Inter $A \times B$	SCE_{AB}	$(I - 1)(J - 1)$	$CMAB = \frac{SCE_{AB}}{(I-1)(J-1)}$	$F_{AB} = \frac{CMAB}{CMR}$	p-value $A \times B$
Erreur	SCR	$n - IJ$	$CMR = \frac{SCR}{n-IJ}$		
Totale	SCT	$n - 1$			

La statistique de test (F-value) est alors de ratio entre la part de variance captée par l'interaction et la variance résiduelle. Elle suit une loi de Fisher. L'interprétation du test est la même que pour l'ANOVA à 1 facteur.

Interprétation de la p-value :

- Si la p-value est inférieur à 5%, alors on rejette H_0 . Autrement dit il y a un effet d'interaction entre A et B . Dans ce cas on considérera aussi que les effets principaux de A et de B ont un effet significatif sur la variable à expliquer. On peut alors passer au diagnostic du modèle pour vérifier les hypothèses puis faire les comparaisons de moyennes ajustées.
- Si la p-value est supérieur à 5%, alors on ne rejette pas H_0 . On ne peut rien dire sur l'effet d'interaction. On enlève alors l'interaction du modèle (modèle additif ne contenant que les effets principaux) et on continue les tests des effets principaux.

Le modèle additif :

$$Y_{ijk} = \mu + \alpha_i + \beta_j + \varepsilon_{ijk}$$

On va donc tester l'effet principal de A :

- H_0 : Pour tout i , $\alpha_i = 0$ (pas d'effet de A)
- H_1 : Il existe au moins un α_i non nul (effet de A)

Et l'effet principal de B :

- H_0 : Pour tout j , $\beta_j = 0$ (pas d'effet de B)
- H_1 : Il existe au moins un β_j non nul (effet de B)

On peut faire ces deux tests en même temps mais il faut recalculer la table ANOVA (la somme des carrés résiduels augmente lorsqu'on enlève l'interaction) :

Variation	SC	d.d.l.	Carrés moyens	Valeur F_{obs}	$P_{H_0}(F > F_{obs})$
Facteur A	SCE_A	$I - 1$	$CMA = \frac{SCE_A}{I-1}$	$F_A = \frac{CMA}{CMR}$	p-value A
Facteur B	SCE_B	$J - 1$	$CMB = \frac{SCE_B}{J-1}$	$F_B = \frac{CMB}{CMR}$	p-value B
Erreur	SCR	$n - I - J + 1$	$CMR = \frac{SCR}{n-I-J+1}$		
Totale	SCT	$n - 1$			

Pour tester l'effet de A on utilise :

$$F_A = \frac{CMA}{CMR} \underset{H_0}{\sim} \mathcal{F}_{(I-1), (n-I-J+1)}$$

De même, pour tester l'effet de B on utilise :

$$F_B = \frac{CMB}{CMR} \underset{H_0}{\sim} \mathcal{F}_{(J-1), (n-I-J+1)}$$

Interprétation de la p-value pour A :

- Si la p-value est inférieur à 5%, alors on rejette H_0 . Le facteur A a un effet significatif sur la variable réponse.
- Si la p-value est supérieur à 5%, alors on ne rejette pas H_0 . On ne peut rien dire sur l'effet principal de A .

Il en est de même pour l'interprétation de la p-value de B .

Exemple : Les données haricots

Ces données sont issues d'une expérimentation au champ visant à tester l'effet de la fertilisation azotée sur la productivité de haricots à grains. Effectivement, plusieurs références font état de la faible capacité fixatrice du haricot qui ne lui permettrait pas de couvrir intégralement ces besoins en azote. On compare la production de matière sèche totale, mesurée à maturité sur des haricots fertilisés (N) ou non fertilisés (0N), et ce pour quatre génotypes.

Ici, on ne tiendra pas compte des blocs qu'on traitera dans la section suivantes.

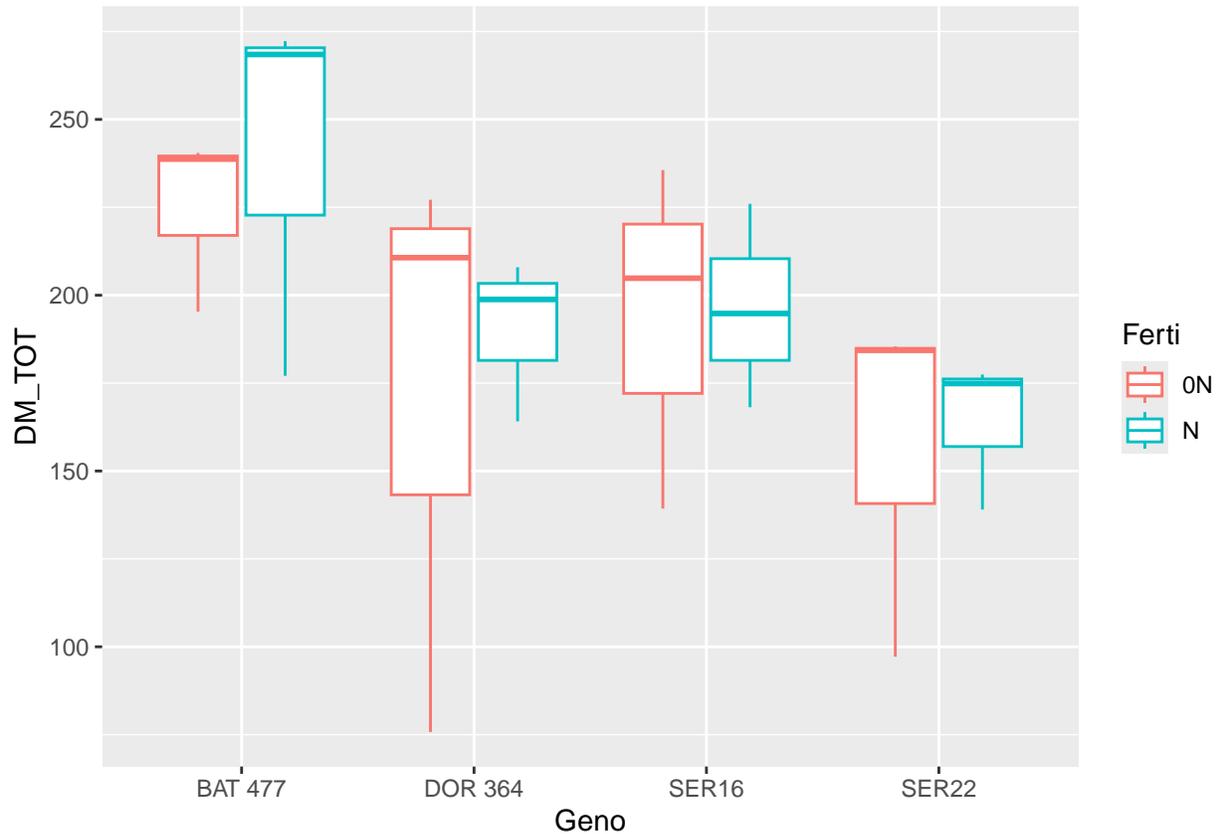
```
haricots <- read.table("data/haricots.txt", header=TRUE, sep="\t", dec=".")

haricots = haricots %>% filter(DAS == 98) %>%
  select(DM_TOT, Bloc, Geno, Treatment) %>%
  rename(Ferti = Treatment)

str(haricots)
```

```
## 'data.frame': 24 obs. of 4 variables:
## $ DM_TOT: num 168 139 177 199 211 ...
## $ Bloc : int 1 1 1 1 2 2 2 2 3 3 ...
## $ Geno : chr "SER16" "SER22" "BAT 477" "DOR 364" ...
## $ Ferti : chr "N" "N" "N" "N" ...
```

```
ggplot(haricots, aes(x=Geno, y = DM_TOT, color = Ferti)) + geom_boxplot()
```



Visuellement, il ne semble pas y avoir d'interaction. On va le vérifier par l'analyse statistique.

```
fit = lm(DM_TOT ~Geno+Ferti + Geno:Ferti, data=haricots)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: DM_TOT
##          Df Sum Sq Mean Sq F value Pr(>F)
## Geno      3  16685   5561.7   2.5753 0.09009 .
## Ferti     1    750    750.1   0.3473 0.56385
## Geno:Ferti 3    223    74.2   0.0344 0.99112
## Residuals 16  34554   2159.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L'interaction n'est pas significative.

```
fit = lm(DM_TOT ~Geno+Ferti, data=haricots)
anova(fit)
```

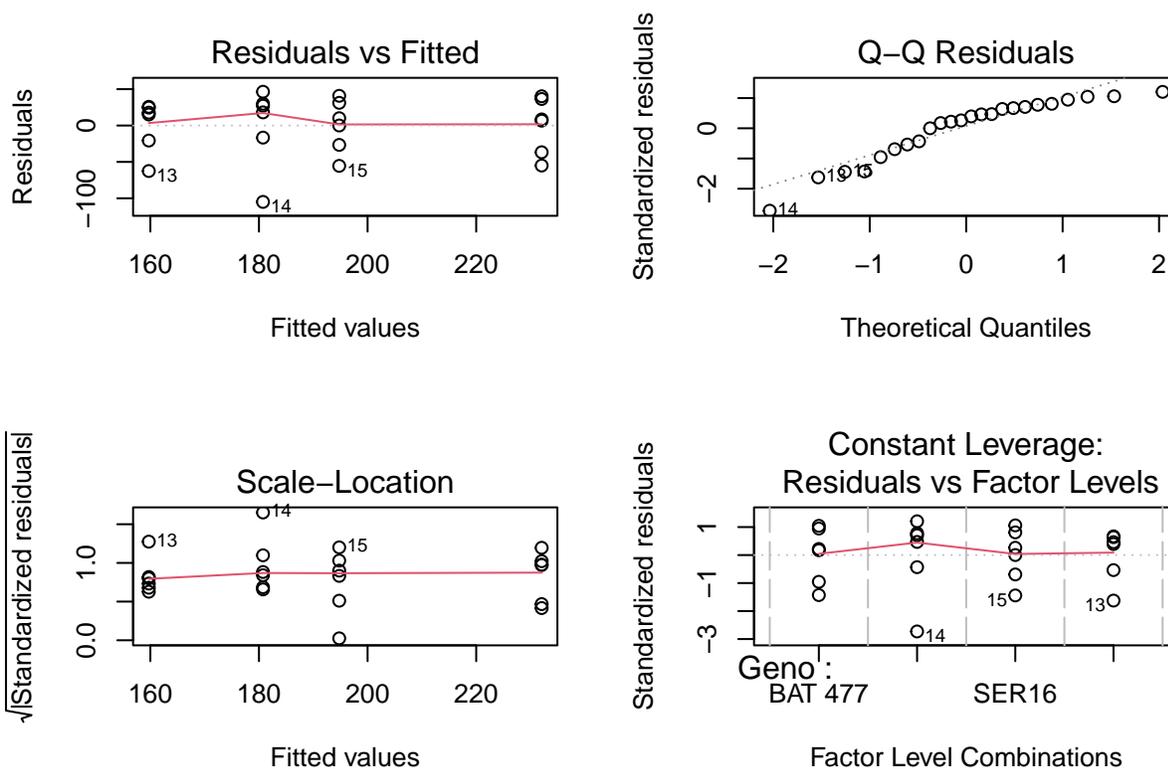
```
## Analysis of Variance Table
##
## Response: DM_TOT
##           Df Sum Sq Mean Sq F value Pr(>F)
## Geno       3  16685  5561.7   3.0386 0.05428 .
## Ferti      1    750   750.1   0.4098 0.52970
## Residuals 19  34777  1830.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

La fertilisation n'est pas significative non plus

```
fit = lm(DM_TOT ~Geno, data=haricots)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: DM_TOT
##           Df Sum Sq Mean Sq F value Pr(>F)
## Geno       3  16685  5561.7   3.131 0.04849 *
## Residuals 20  35527  1776.3
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow = c(2, 2))
plot(fit)
```



Plan factoriel en blocs à un facteur de contrôle d'hétérogénéité

Plan en bloc complet en randomisation totale (RCB design) C'est la généralisation du plan en blocs complets vu précédemment au cas où l'on aurait plusieurs facteurs. Là encore, on utilisera ce type de plan lorsque des sources d'hétérogénéité sont présentes sur le terrain. On constitue alors des groupes de parcelles supposées homogènes qui formeront les blocs.

Génération :

```
A = c("a", "b")
B = 1:3
trt = expand.grid(A = A, B=B)
trt$Traitement = paste0(trt$A, "_", trt$B)
trt
```

```
##   A B Traitement
## 1 a 1      a_1
## 2 b 1      b_1
## 3 a 2      a_2
## 4 b 2      b_2
## 5 a 3      a_3
## 6 b 3      b_3
```

```
Traitement = trt$Traitement
design.rcbd(Traitement, r=3, serie = 1, seed = 0, kinds = "Super-Duper")
```

```
## $parameters
## $parameters$design
## [1] "rcbd"
##
## $parameters$trt
## [1] "a_1" "b_1" "a_2" "b_2" "a_3" "b_3"
##
## $parameters$r
## [1] 3
##
## $parameters$serie
## [1] 1
##
## $parameters$seed
## [1] -1914657327
##
## $parameters$kinds
## [1] "Super-Duper"
##
## $parameters[[7]]
## [1] TRUE
##
## $sketch
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,] "b_1" "b_2" "a_3" "a_2" "b_3" "a_1"
## [2,] "a_2" "a_3" "b_2" "b_1" "a_1" "b_3"
## [3,] "a_3" "a_1" "b_2" "a_2" "b_1" "b_3"
##
## $book
##      plots block Traitement
## 1      11      1      b_1
## 2      12      1      b_2
## 3      13      1      a_3
## 4      14      1      a_2
## 5      15      1      b_3
## 6      16      1      a_1
## 7      21      2      a_2
## 8      22      2      a_3
## 9      23      2      b_2
## 10     24      2      b_1
## 11     25      2      a_1
## 12     26      2      b_3
## 13     31      3      a_3
## 14     32      3      a_1
## 15     33      3      b_2
## 16     34      3      a_2
## 17     35      3      b_1
## 18     36      3      b_3
```

Analyse : L'analyse se fait au travers d'un modèle linéaire. Prenons un exemple avec 2 facteurs A et B . On note AB l'effet d'interaction. Soit U l'effet bloc. Le modèle peut alors s'écrire :

$$Y_{i,j,k} = \mu + A_i + B_j + AB_{i,j} + U_k + \varepsilon_{i,j,k}$$

avec :

- i l'indice des niveaux du facteur A
- j l'indice des niveaux du facteur B
- k est l'indice des blocs
- les $\varepsilon_{i,j,k}$ sont supposés iid suivants une loi normale $N(0, \sigma_e^2)$ avec σ_e^2 la variance résiduelle.

L'effet bloc peut être pris en aléatoire si le nombre de blocs est suffisant. Ici, on le considèrera en effet fixe.

Ce modèle peut être mis en oeuvre avec R à l'aide des fonctions `aov` ou `lm` si U est en effet fixe ou alors à l'aide des packages `nlme` ou `lme4` si U est pris en effet aléatoire.

On peut ensuite tester les effets d'interaction principaux. **IMPORTANT :** on commence toujours par tester l'effet d'interaction. S'il est significatif alors on garde les effets principaux. Même s'ils ne semblent pas significatifs. L'idée est que si l'interaction est significative alors les deux facteurs ont un effet sur Y donc on garde les effets principaux.

Exemple : Les données haricots

Ces données sont issues d'une expérimentation au champ visant à tester l'effet de la fertilisation azotée sur la productivité de haricots à grains. Effectivement, plusieurs références font état de la faible capacité fixatrice du haricot qui ne lui permettrait pas de couvrir intégralement ces besoins en azote. On compare la production de matière sèche totale, mesurée à maturité sur des haricots fertilisés (N) ou non fertilisés (0N), et ce pour quatre génotypes.

```
haricots <- read.table("data/haricots.txt", header=TRUE, sep="\t", dec=".")

haricots = haricots %>% filter(DAS == 98) %>%
  select(DM_TOT, Bloc, Geno, Treatment) %>%
  rename(Ferti = Treatment)

str(haricots)
```

```
## 'data.frame': 24 obs. of 4 variables:
## $ DM_TOT: num 168 139 177 199 211 ...
## $ Bloc : int 1 1 1 1 2 2 2 2 3 3 ...
## $ Geno : chr "SER16" "SER22" "BAT 477" "DOR 364" ...
## $ Ferti : chr "N" "N" "N" "N" ...
```

```
fit = lm(DM_TOT ~Geno+Ferti + Geno:Ferti + Bloc, data=haricots)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: DM_TOT
##          Df Sum Sq Mean Sq F value Pr(>F)
## Geno      3 16685.2  5561.7   4.6482 0.017238 *
## Ferti     1   750.1   750.1   0.6269 0.440832
## Bloc      1 16605.9 16605.9 13.8783 0.002031 **
```

```
## Geno:Ferti 3 222.7 74.2 0.0620 0.979040
## Residuals 15 17948.0 1196.5
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

L'interaction n'est pas significative.

```
fit = lm(DM_TOT ~Geno+Ferti + Bloc, data=haricots)
anova(fit)
```

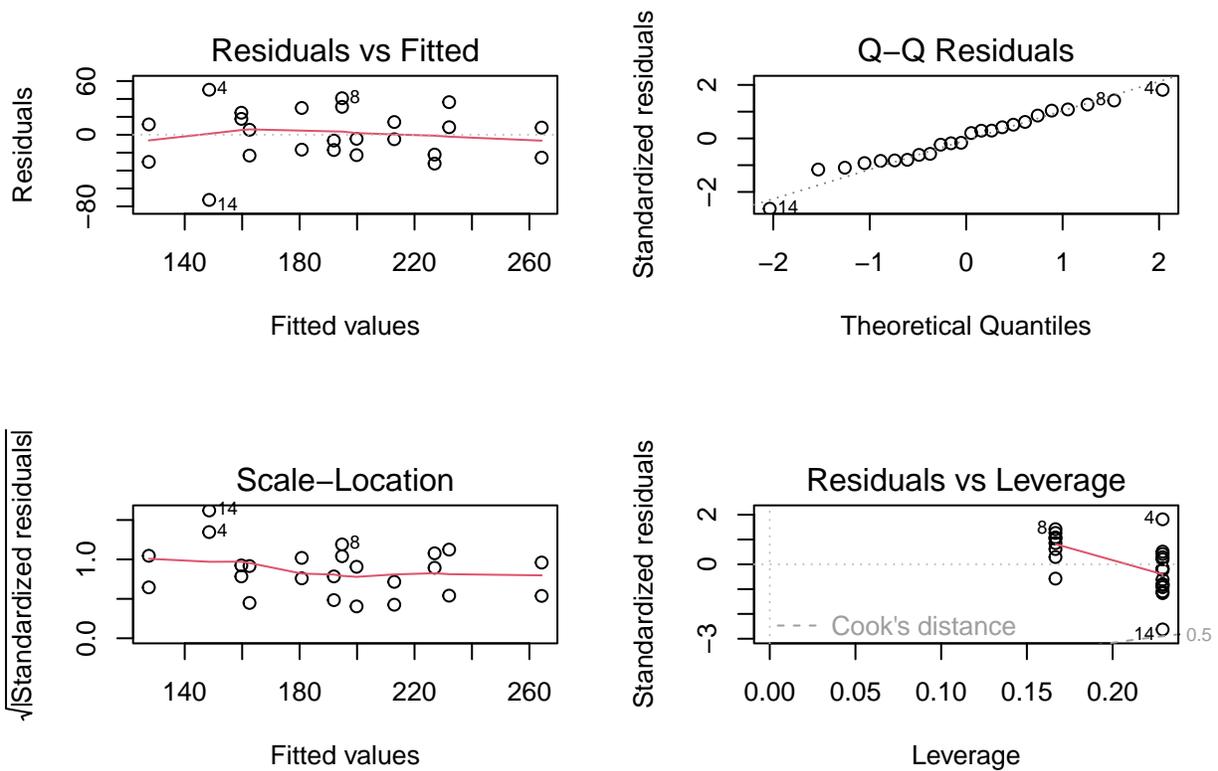
```
## Analysis of Variance Table
##
## Response: DM_TOT
##      Df Sum Sq Mean Sq F value    Pr(>F)
## Geno   3 16685.2  5561.7   5.5095 0.0072938 **
## Ferti   1   750.1   750.1   0.7431 0.4000159
## Bloc    1 16605.9 16605.9 16.4499 0.0007416 ***
## Residuals 18 18170.7 1009.5
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

La fertilisation n'est pas significative non plus

```
fit = lm(DM_TOT ~Geno+ Bloc, data=haricots)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: DM_TOT
##      Df Sum Sq Mean Sq F value    Pr(>F)
## Geno   3 16685  5561.7   5.585 0.0064063 **
## Bloc    1 16606 16605.9 16.675 0.0006331 ***
## Residuals 19 18921   995.8
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
par(mfrow = c(2, 2))
plot(fit)
```



Plans factoriels avec plusieurs unités expérimentales emboîtées ou croisées

Split-plot : des effets principaux sont confondus avec des blocs Le plan en split-plot est un plan en blocs complets avec 2 facteurs étudiés et un facteur de contrôle d'hétérogénéité.

Ce plan est utilisé lorsque chaque modalité d'un des deux facteurs étudiés ne peut être appliquée aux unités expérimentales dues à des contraintes techniques et sera donc appliquée à des sous-blocs, grandes parcelles (plots), contenant plusieurs unités expérimentales. Pour ce facteur en question, que l'on appellera le premier facteur, l'effet de chacun de ces niveaux est confondu avec l'hétérogénéité du terrain au niveau des sous-blocs (plots), alors que, pour le deuxième facteur, les différences au niveau des sous-blocs sont éliminées. Le split-plot est une technique qui "sacrifie" le premier facteur (perte d'une grande partie de l'information sur le premier facteur).

Dans ce dispositif, chaque bloc (1 à r) est tout d'abord découpé en I sous-blocs, appelés grandes parcelles (plots) sur lesquelles seront appliquées les modalités d'un premier facteur (ex variété). Puis, chaque grande parcelle est elle-même subdivisée en J unités expérimentales, appelées petites parcelles (split-plots) sur lesquelles seront appliquées les modalités d'un second facteur (ex nitrogène). Il y a donc au total rIJ petites parcelles.

La randomisation s'effectue alors de la façon suivante : dans chaque bloc, les I modalités du premier facteur sont attribuées aléatoirement aux I grandes parcelles. Puis dans un second temps, pour chaque grande parcelle, les J modalités du second facteur sont attribuées aléatoirement aux J petites parcelles.

Bloc

B	B	B
Y	X	Z

A	A	A
Z	X	Y

I

A	A	A
X	Y	Z

B	B	B
X	Z	Y

II

A	A	A
X	Y	Z

B	B	B
Y	X	Z

III

B	B	B
Z	X	Y

A	A	A
Z	Y	X

IV

```
library(agricolae)
Trt1 <- c("A", "B") # ex : 2 variétés
Trt2 <- c("X", "Y", "Z") # ex : azote 3 niveaux
nb_blocs <- 4
design_sp <- design.split(trt1 = Trt1,
                        trt2 = Trt2,
                        r = nb_blocs,
                        seed = 42)$book
```

```
design_sp <- design_sp %>%
  mutate(Row = block %>% as.integer) %>%
  group_by(Row) %>% mutate(Col = 1:n()) %>% ungroup()

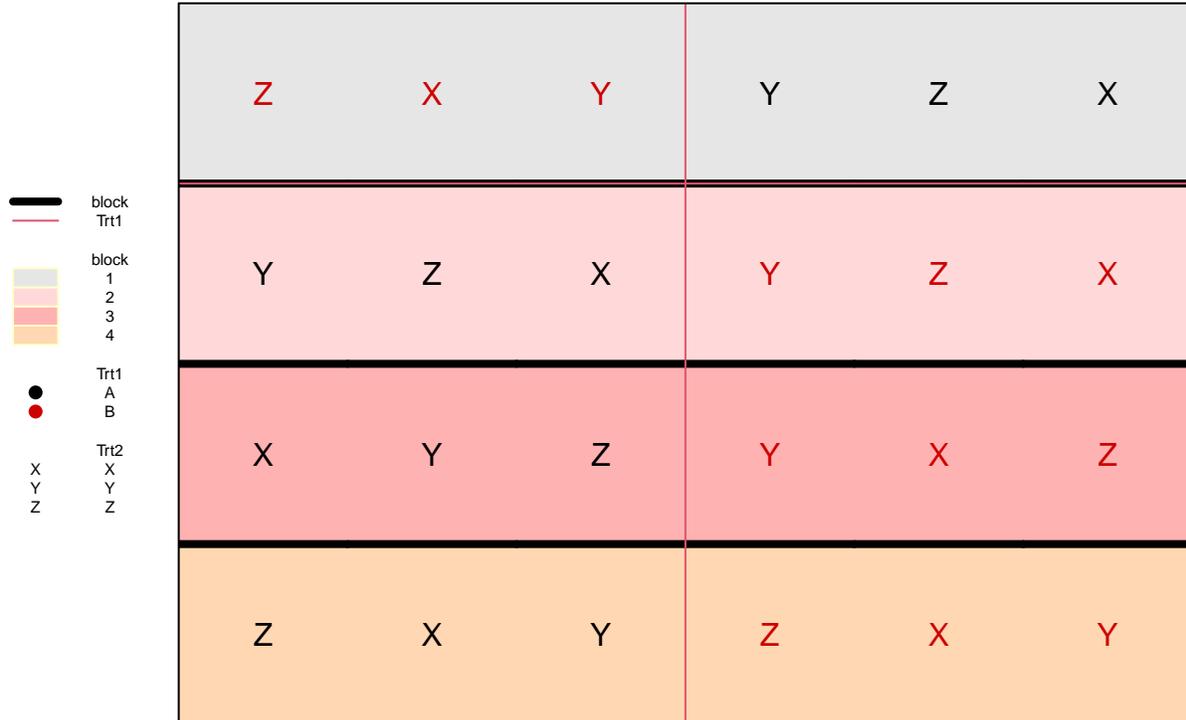
(design_sp)
```

```
## # A tibble: 24 x 7
##   plots  splots block Trt1  Trt2   Row   Col
##   <dbl> <fct>  <fct> <fct> <chr> <int> <int>
## 1  101 1      1      B      Z     1     1
## 2  101 2      1      B      X     1     2
## 3  101 3      1      B      Y     1     3
## 4  102 1      1      A      Y     1     4
## 5  102 2      1      A      Z     1     5
## 6  102 3      1      A      X     1     6
## 7  103 1      2      A      Y     2     1
## 8  103 2      2      A      Z     2     2
## 9  103 3      2      A      X     2     3
## 10 104 1      2      B      Y     2     4
## # i 14 more rows
```

Représentation graphique : les blocs sont en lignes, les traitements en colonnes

```
# Plot field layout
desplot(block ~ Col + Row, flip = TRUE,
  out1 = block, out1.gpar = list(col = 1, lty = 1, lwd = 4),
  out2 = Trt1, out2.gpar = list(col = 2, lty = 1), # optionnel
  text = Trt2, cex = 1, shorten = "no",
  col=Trt1,
  data = design_sp,
  main = "split plot design",
  show.key = T, key.cex = 0.5)
```

split plot design

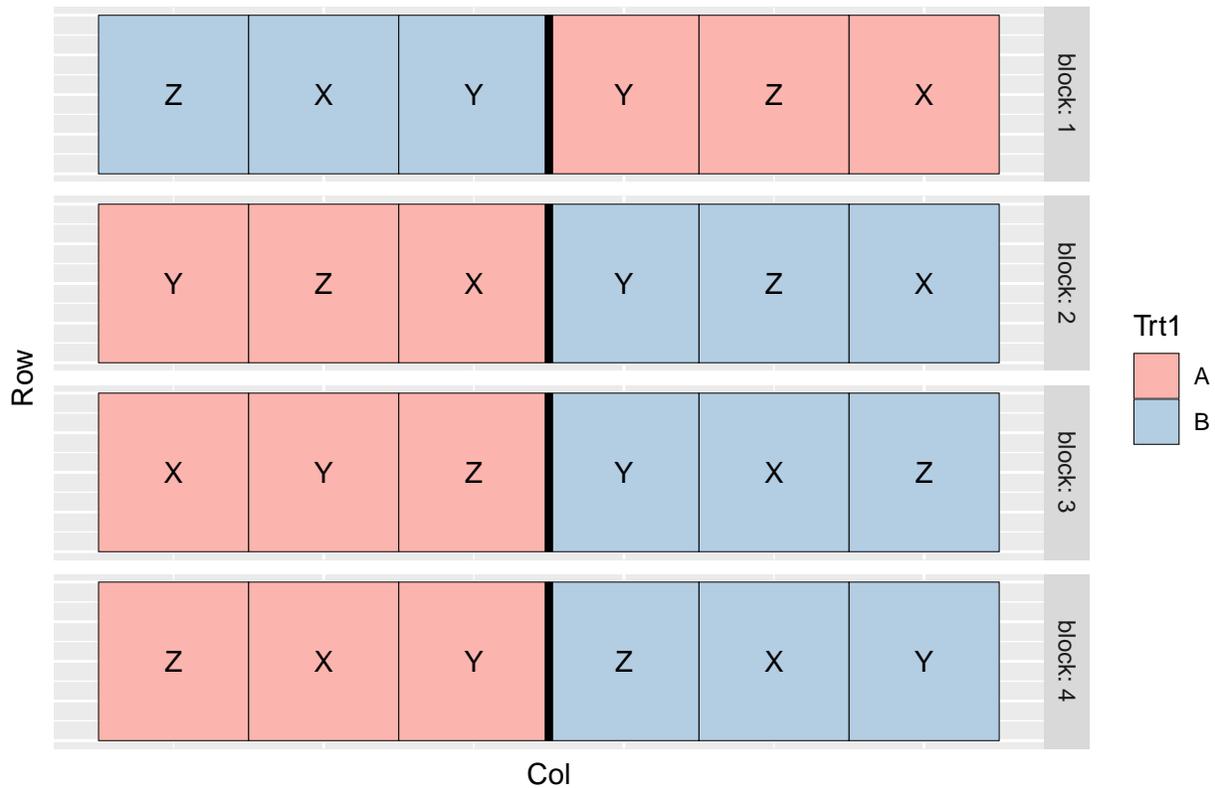


Avec ggplot

```
design_sp2 = design_sp
design_sp2$Row=1

ggplot(design_sp2, aes(x=Col, y=Row)) +
  facet_grid(block~., labeller = label_both) +
  geom_tile(aes(fill=Trt1), color = "black") +
  scale_fill_brewer(palette="Pastel1") +
  geom_tileborder(aes(group=1, grp=Trt1), lwd=1.5)+
  geom_text(aes(label = Trt2), color="black", size = 4) +
  # scale_color_brewer(palette="RdGy")+
  ggtitle("Split plot design") +
  theme(axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank())
```

Split plot design



Analyse :

L'analyse d'un tel dispositif est un peu particulière, car l'effet du facteur appliqué sur les parcelles principales (notons le A) ne peut pas être traité de la même manière qu'un effet d'un facteur appliqué au niveau des parcelles élémentaires (notons le B).

L'effet du facteur B pourra s'observer entre les différentes parcelles élémentaires alors que l'effet du facteur A pourra s'observer entre les différentes parcelles principales. Ainsi, pour tester l'effet d'un facteur du modèle, le principe est que les variabilités expliquées par les différents effets du modèle ne sont pas comparées aux mêmes variabilités résiduelles, en fonction de l'échelle d'application du facteur. Pour cela, on peut soit faire deux ANOVA à différentes échelles. L'une au niveau des parcelles principales pour tester l'effet du facteur A (on moyenne les observations à l'échelle des parcelles principales) et l'autre à l'échelle des parcelles élémentaires pour tester l'effet du facteur B et l'interaction $A \times B$. Soit faire un modèle mixte en prenant en effet aléatoire le facteur parcelle principale.

L'idée est toujours la même. Pour tester l'effet d'un facteur, on compare la variance expliquée par le facteur (inter-niveaux du facteur) à la variance non-expliquée par le facteur (variance intra-niveaux du facteur \rightarrow variance résiduelle). Mais ici les facteurs ne sont pas appliqués à la même échelle et donc les variances doivent être calculées à l'échelle d'application du facteur :

- La variabilité expliquée par A (variabilité inter-niveaux de A) est comparée à la variabilité intra-niveaux de A (variabilité résiduelle dans le cas d'une ANOVA où les sub-plots sont considérés comme étant les observations).
- La variabilité expliquée par B (variabilité inter-niveaux de B) est comparée à la variabilité intra-niveaux de B (variabilité résiduelle dans le cas d'une ANOVA où les parcelles élémentaires sont considérées comme étant les observations).

Généralement, lors de la première analyse (test de A), la variance résiduelle est plus élevée et les degrés

de libertés sont plus faibles que celle de la deuxième analyse (test de B et des interactions). Ainsi, on aura en général plus de puissance pour tester l'effet du facteur B que pour tester l'effet du facteur A . On parle parfois de facteur “sacrifié” pour les effets appliqués aux niveaux des parcelles principales.

L'analyse du split-plot consiste donc à décomposer la variabilité résiduelle à différentes échelles.

On présente ici l'analyse au travers d'un LMM. Les parcelles principales sont obligatoirement prise en effet aléatoire. Les blocs peuvent être pris en effet fixe ou aléatoire.

Le modèle est alors de la forme :

$$y_{ijk} = \underbrace{\mu + \alpha_i + \beta_j + \gamma_{ij}}_{\text{effets fixes}} + \underbrace{u_k + v_{ik} + \varepsilon_{ijk}}_{\text{effets aléatoires}}$$

avec

- i l'indice des modalités du premier facteur (trt1),
- j l'indice des modalités du deuxième facteur (trt2),
- k l'indice du bloc,
- y_{ijk} l'observation de l'individu ijk ,
- μ l'intercept,
- α_i l'effet de la i ème modalité du premier facteur (trt1),
- β_j l'effet de la j ème modalité du deuxième facteur (trt2),
- γ_{ij} l'effet d'interaction entre la i ème modalité du premier facteur (trt1) et de la j ème modalité du deuxième facteur (trt2),
- u_k l'effet aléatoire du k ème bloc avec $u \sim N(0, \sigma_u^2 I)$ avec σ_u^2 la variance de cet effet aléatoire et I une matrice diagonale (de taille égale au nombre blocs),
- v_{ik} l'effet aléatoire de la parcelle principale ik avec $v \sim N(0, \sigma_v^2 I)$ avec σ_v^2 la variance de cet effet aléatoire et I une matrice diagonale (de taille égale au nombre parcelles primères),
- ε_{ijk} l'erreur associé à l'individu ijk supposée suivre une loi normale $N(0, \sigma^2)$ avec σ^2 la variance résiduelle. Tous les ε_{ijk} sont supposés iid.

S'il y a trop peu de blocs, alors il sera préférable de prendre l'effet bloc u_k en effet fixe.

Les parcelles principales doivent être présent en effet aléatoire.

L'estimation de ce modèle peut se faire sous R avec les bibliothèques `nlme` et `lmer` entre autres.

Exemple : `yates.oat`

This dataset is available in the R package `agridat`. It gives the yield of oats from a split-plot field trial using three varieties and four levels of manurial treatment. The experiment was laid out in 6 blocks of 3 main plots, each split into 4 sub-plots. The varieties were applied to the main plots and the manurial (nitrogen) treatments to the sub-plots. It is a data frame with 72 observations on the following 6 variables:

- `nitro` nitrogen treatment in hundredweight per acre
- `block` block factor, 6 levels
- `gen` genotype factor, 3 levels
- `x` x ordinate
- `y` y ordinate
- `yield` yield in 1/4 lbs per sub-plot, each 1/80 acre.

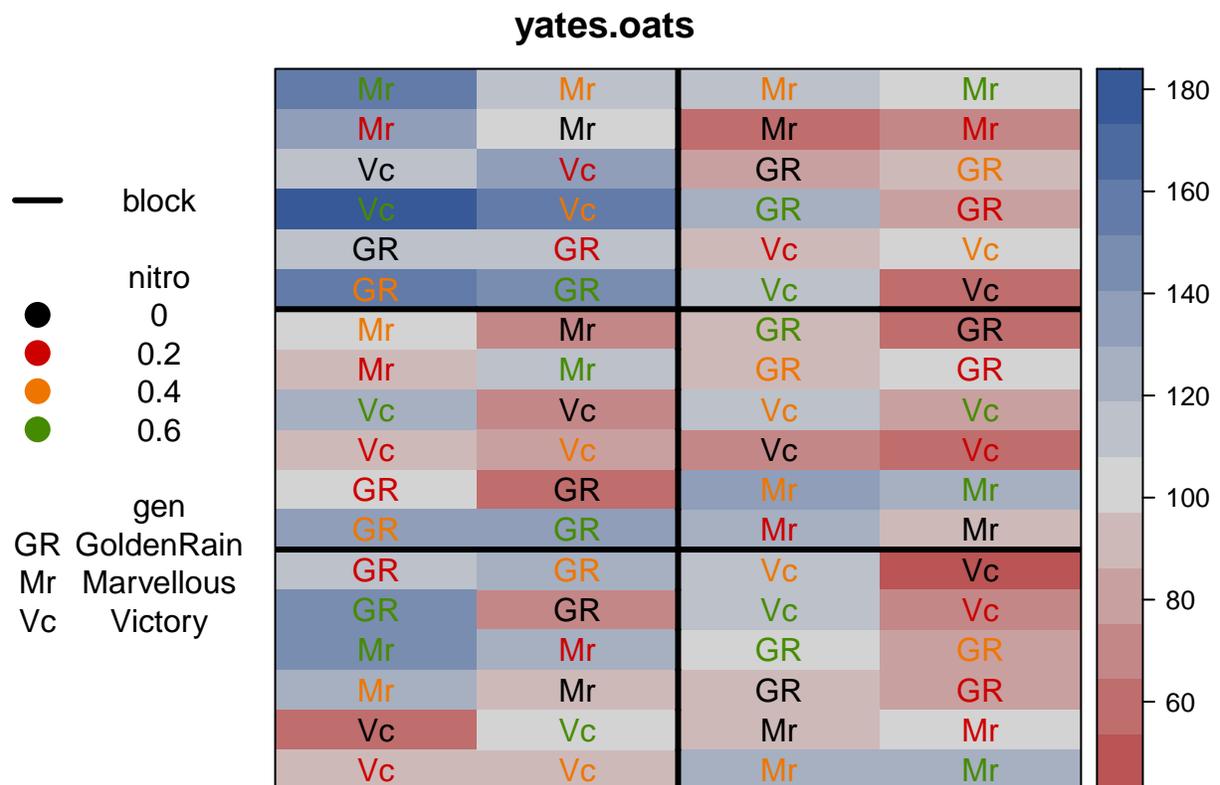
Référence:

- Yates, Frank (1935) Complex experiments, Journal of the Royal Statistical Society Suppl. 2, 181–247.
- Venables, W. N. and Ripley, B. D. (2002). Modern Applied Statistics with S. Fourth edition. Springer.

```
library(agridat)
str(yates.oats)
```

```
## 'data.frame': 72 obs. of 8 variables:
## $ row : int 16 12 3 14 8 5 15 11 3 14 ...
## $ col : int 3 4 3 1 2 2 4 4 4 2 ...
## $ yield: int 80 60 89 117 64 70 82 102 82 114 ...
## $ nitro: num 0 0 0 0 0 0 0.2 0.2 0.2 0.2 ...
## $ gen : Factor w/ 3 levels "GoldenRain","Marvellous",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ block: Factor w/ 6 levels "B1","B2","B3",...: 1 2 3 4 5 6 1 2 3 4 ...
## $ grain: num 20 15 22.2 29.2 16 ...
## $ straw: num 28 25 40.5 28.8 32 ...
```

```
yates.oats$nitro <- as.factor(yates.oats$nitro)
desplot(yield ~ col*row, data=yates.oats, out1=block, text=gen, col=nitro, cex=1, main="yates.oats")
```



Ici il y a 6 blocs, on va prendre le facteur bloc en effet fixe.

Le facteur parcelle principale est pris en aléatoire (obligatoire dans un split-plot). On commence par définir ce facteur. L'identification des parcelles principales peut se faire en regardant le niveaux de génotype combiné au bloc :

```
yates.oats$PP = paste0(yates.oats$block, "_", yates.oats$gen)
```

```
library(nlme)
m1 <- lme(yield ~ nitro + gen + nitro:gen + block, random=list(~1|PP), data=yates.oats, method = "REML")
```

Pour faire la sélection des effets fixe, on fait un ANOVA en utilisant l'estimation par maximum de vraisemblance "ML"

```
m1 <- lme(yield ~ nitro + gen + nitro:gen + block, random=list(~1|PP), data=yates.oats, method = "ML")
anova(m1)
```

```
##           numDF denDF   F-value p-value
## (Intercept)     1    45 1779.7401 <.0001
## nitro           3    45  34.5452 <.0001
## gen            2    10   2.0423 0.1804
## block          5    10   7.2601 0.0041
## nitro:gen       6    45   0.2776 0.9446
```

On ne garde pas l'interaction

```
m1 <- lme(yield ~ nitro + gen + block, random=list(~1|PP), data=yates.oats, method = "ML")
anova(m1)
```

```
##           numDF denDF   F-value p-value
## (Intercept)     1    51 1973.8935 <.0001
## nitro           3    51  36.8268 <.0001
## gen            2    10   2.2651 0.1544
## block          5    10   8.0521 0.0028
```

On ne garde pas le génotype. Remarque, les plans split-plot offre une faible précision pour tester l'effet du facteur appliqué sur les parcelles principales, en l'occurrence le génotype ici.

```
m1 <- lme(yield ~ nitro + block, random=list(~1|PP), data=yates.oats, method = "ML")
anova(m1)
```

```
##           numDF denDF   F-value p-value
## (Intercept)     1    51 1571.7072 <.0001
## nitro           3    51  38.0342 <.0001
## block          5    12   6.4114 0.004
```

```
m1 <- lme(yield ~ nitro + block, random=list(~1|PP), data=yates.oats, method = "REML")
summary(m1)
```

```
## Linear mixed-effects model fit by REML
## Data: yates.oats
##      AIC      BIC    logLik
## 560.3467 583.9212 -269.1734
##
## Random effects:
## Formula: ~1 | PP
##      (Intercept) Residual
## StdDev:    11.03872 12.74986
```

```

##
## Fixed effects: yield ~ nitro + block
##           Value Std.Error DF   t-value p-value
## (Intercept) 66.33333  7.806256 51  8.497458  0.0000
## nitro0.2    19.50000  4.249952 51  4.588287  0.0000
## nitro0.4    34.83333  4.249952 51  8.196171  0.0000
## nitro0.6    44.00000  4.249952 51 10.353058  0.0000
## blockB2     5.00000 10.408107 12  0.480395  0.6396
## blockB3     5.33333 10.408107 12  0.512421  0.6177
## blockB4    44.41667 10.408107 12  4.267507  0.0011
## blockB5     7.25000 10.408107 12  0.696572  0.4993
## blockB6    16.33333 10.408107 12  1.569289  0.1426
## Correlation:
##           (Intr) ntr0.2 ntr0.4 ntr0.6 blckB2 blckB3 blckB4 blckB5
## nitro0.2 -0.272
## nitro0.4 -0.272  0.500
## nitro0.6 -0.272  0.500  0.500
## blockB2 -0.667  0.000  0.000  0.000
## blockB3 -0.667  0.000  0.000  0.000  0.500
## blockB4 -0.667  0.000  0.000  0.000  0.500  0.500
## blockB5 -0.667  0.000  0.000  0.000  0.500  0.500  0.500
## blockB6 -0.667  0.000  0.000  0.000  0.500  0.500  0.500  0.500
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -1.83503238 -0.56416699  0.06460638  0.61725265  1.71239003
##
## Number of Observations: 72
## Number of Groups: 18

```

```

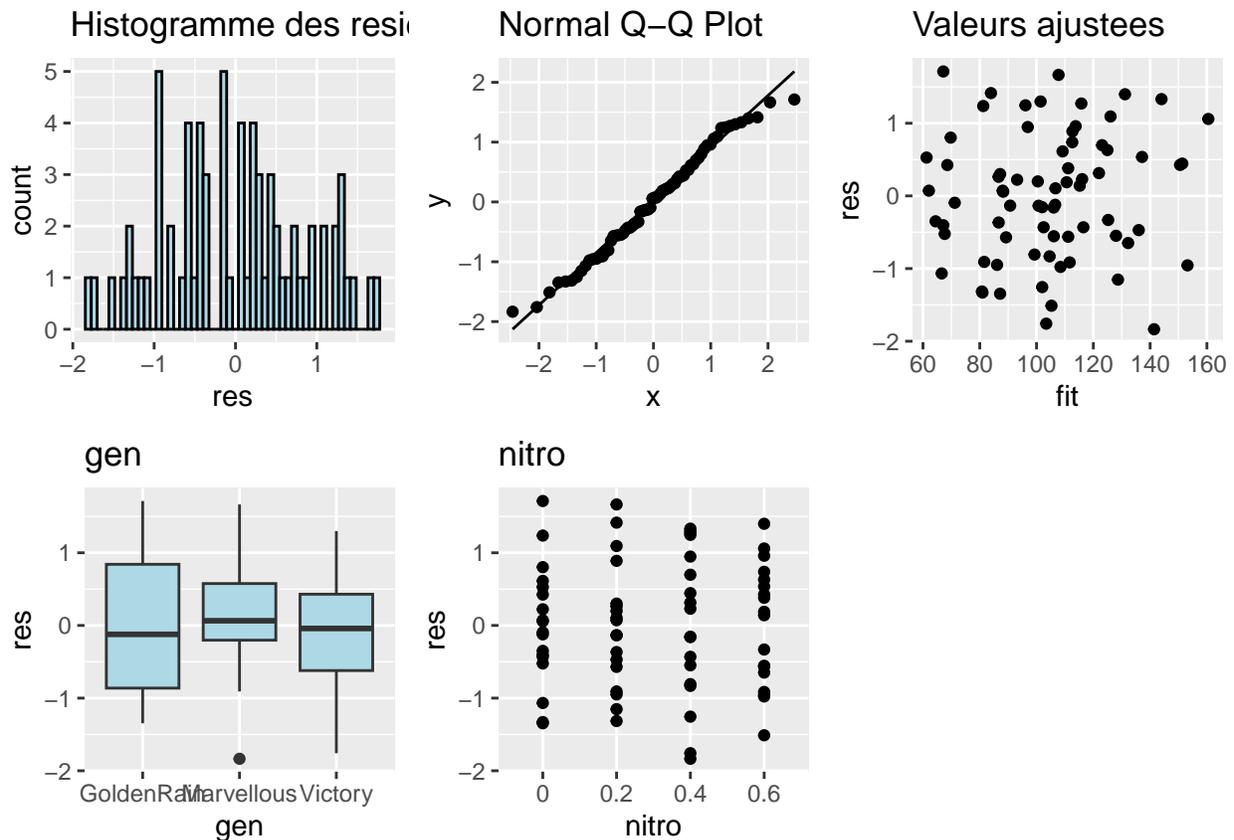
res <- residuals(m1, type="normalized")
fitted <- fitted(m1)

resfit <- bind_cols(res = res,
                    fit = fitted,
                    gen = yates.oats$gen,
                    nitro = yates.oats$nitro)

grid.arrange(nrow = 2, ncol = 3,
             ggplot(resfit, aes(x = res)) +
               geom_histogram(bins = 50,
                              fill = "lightblue", color = "black")+
               labs(title = "Histogramme des residus"),
             ggplot(resfit, aes(sample = res)) +
               stat_qq() +
               stat_qq_line()+
               labs(title = "Normal Q-Q Plot"),
             ggplot(resfit, aes(y = res, x = fit)) +
               geom_point()+
               labs(title = "Valeurs ajustees"),
             ggplot(resfit, aes(y = res, x = gen)) +
               geom_boxplot(fill = "lightblue") +
               labs(title = "gen"),
             ggplot(resfit, aes(y = res, x = nitro)) +

```

```
geom_point()+
labs(title = "nitro"))
```



```
ks.test(res, "pnorm", 0, sd(res))
```

```
##
## Exact one-sample Kolmogorov-Smirnov test
##
## data: res
## D = 0.05089, p-value = 0.9876
## alternative hypothesis: two-sided
```

Plan criss-cross (en bande croisées) : Autres appellations : split-block ou strip-plot

Ce dispositif est utile pour croiser deux facteurs dont aucun ne peut être appliqué aux parcelles élémentaires, mais seulement à des parcelles plus grandes. À l'intérieur de chaque bloc complet, une matrice de parcelles élémentaires est dessinée ; le premier facteur sera appliqué aux lignes, et l'autre aux colonnes. La randomisation se fait en deux étapes, indépendamment sur les lignes et les colonnes, et détermine 3 termes d'erreur : sur les lignes, sur les colonnes, et sur les parcelles élémentaires.

Y	X	Z		Bloc
BY	BX	BZ	B	I
AY	AX	AZ	A	

Z	X	Y		Bloc
AZ	AX	AY	A	II
BZ	BX	BY	B	

```
library(agricolae)
Trt1 <- c("A", "B")      # ex : 2 variétés
Trt2 <- c("X", "Y", "Z") # ex : azote 3 niveaux
nb_blocs <- 2
design_cc <- design.strip(Trt1, Trt2, nb_blocs, seed=45, kinds = "Super-Duper")$book
design_cc$block <- paste0("Block ", design_cc$block)
design_cc$Row = rep(rep(1:length(Trt1), each=length(Trt2)), nb_blocs)
design_cc$Col = rep(rep(1:length(Trt2), length(Trt1)), nb_blocs)

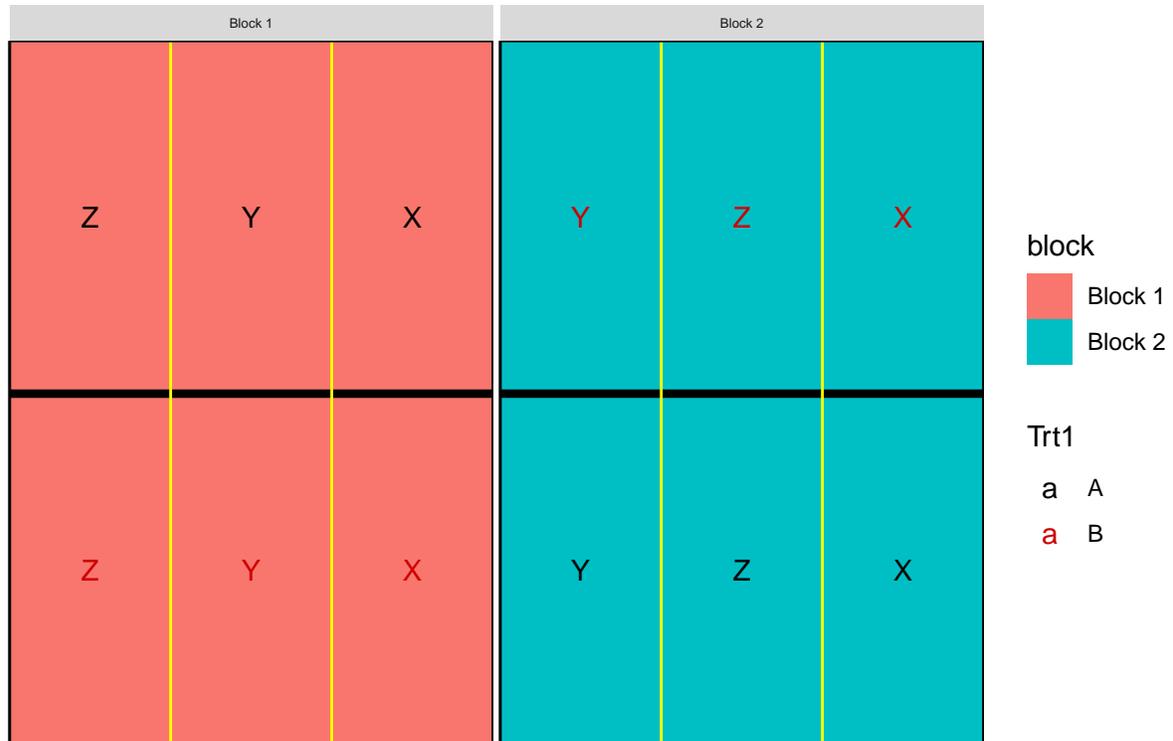
head(design_cc)
```

```
##   plots  block Trt1 Trt2 Row Col
## 1   101 Block 1  A   Z   1   1
## 2   102 Block 1  A   Y   1   2
## 3   103 Block 1  A   X   1   3
## 4   104 Block 1  B   Z   2   1
## 5   105 Block 1  B   Y   2   2
## 6   106 Block 1  B   X   2   3
```

```
desplot(block ~ Col + Row | block, flip = TRUE,
        out1 = Row, out1.gpar = list(col = "grey", lty = 1),
        out2 = Col, out2.gpar = list(col = "grey", lty = 1),
        text = Trt2, cex = 1, shorten = "no", col=Trt1,
        data = design_cc,
```

```
main = "Criss cross design",
show.key = T, key.cex = 1, gg=TRUE)
```

Criss cross design



Modèle statistique et Analyse :

On présente ici l'analyse au travers d'un LMM. Les parcelles principales sont obligatoirement prise en effet aléatoire. Les blocs peuvent être pris en effet fixe ou aléatoire.

Le modèle est alors de la forme :

$$y_{ijklk} = \underbrace{\mu + \alpha_i + \beta_j + \gamma_{ij} + b_k}_{\text{effets fixes}} + \underbrace{L_{lk} + C_{ck} + \varepsilon_{ijklk}}_{\text{effets aléatoires}}$$

avec

- i l'indice des modalités du premier facteur (trt1),
- j l'indice des modalités du deuxième facteur (trt2),
- k l'indice du bloc,
- l l'indice des lignes,
- c l'indice des colonnes,
- y_{ijklk} l'observation de l'individu ijk ,
- μ l'intercept,
- α_i l'effet de la i ème modalité du premier facteur (trt1),
- β_j l'effet de la j ème modalité du deuxième facteur (trt2),
- γ_{ij} l'effet d'interaction entre la i ème modalité du premier facteur (trt1) et de la j ème modalité du deuxième facteur (trt2),
- b_k l'effet du k ème bloc

- L_{lk} l'effet aléatoire de la ligne l du k ème bloc avec $L \sim N(0, \sigma_L^2 I)$ avec σ_L^2 la variance de cet effet aléatoire et I une matrice diagonale (de taille égale au nombre de lignes),
- C_{ck} l'effet aléatoire de la colonne c du k ème bloc avec $C \sim N(0, \sigma_C^2 I)$ avec σ_C^2 la variance de cet effet aléatoire et I une matrice diagonale (de taille égale au nombre de colonnes),
- ε_{ijklk} l'erreur associé à l'individu $ijklk$ supposée suivre une loi normale $N(0, \sigma^2)$ avec σ^2 la variance résiduelle. Tous les ε_{ijklk} sont supposés iid.

L'estimation de ce modèle peut se faire sous R avec les bibliothèques `nmls` et `lmer` entre autres.

Exemple : Strip-plot experiment of rice

A strip-plot experiment with three reps, variety as the horizontal strip and nitrogen fertilizer as the vertical strip.

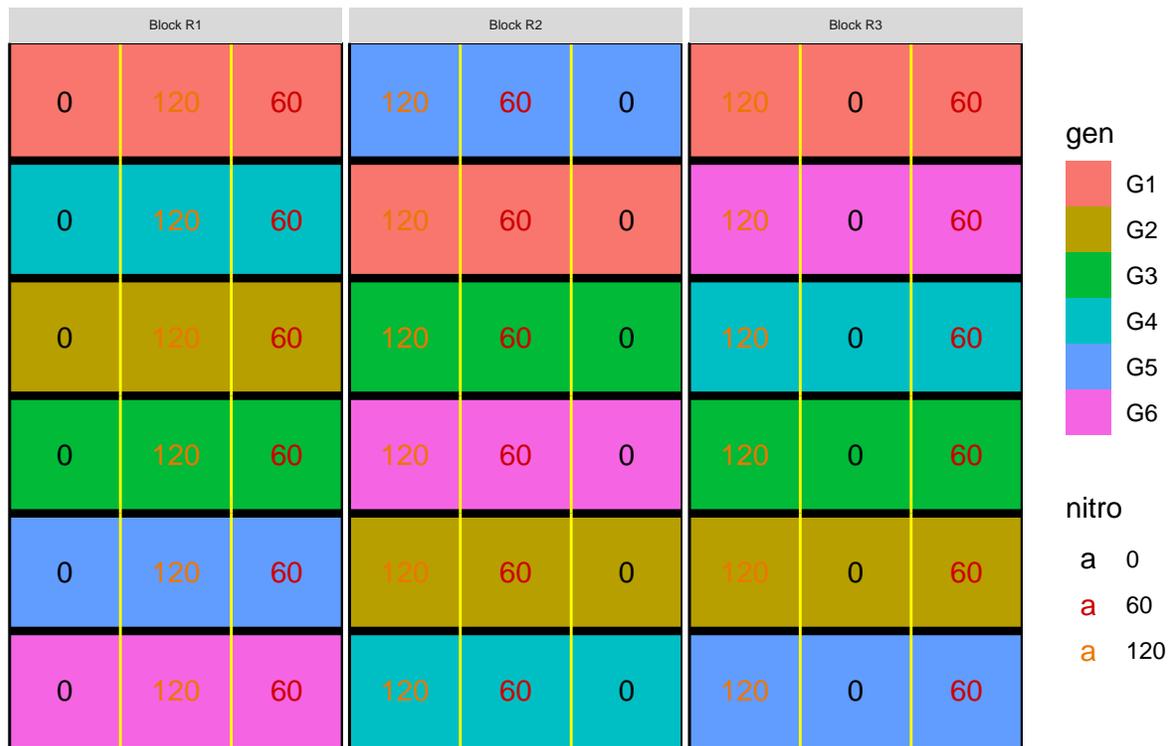
Gomez, K.A. and Gomez, A.A.. 1984, Statistical Procedures for Agricultural Research. Wiley-Interscience. Page 110.

Jan Gertheiss (2014). ANOVA for Factors With Ordered Levels. J Agric Biological Environmental Stat, 19, 258-277.

```
library(agridat)
# data(gomez.stripplot)
dat <- gomez.stripplot
dat$nitro <- factor(dat$nitro)
dat$block <- paste0("Block ", dat$rep)
```

```
desplot(gen ~ col + row | block, flip = TRUE,
        out1 = row, out1.gpar = list(col = "grey", lty = 1),
        out2 = col, out2.gpar = list(col = "grey", lty = 1),
        text = nitro, cex = 1, shorten = "no", col=nitro,
        data = dat,
        main = "Criss cross design",
        show.key = T, key.cex = 1, gg=TRUE)
```

Criss cross design



```
library(lme4)
m1 <- lmer(yield ~ nitro + gen + nitro:gen + block +
           (1|block:gen) +
           (1|block:nitro),
           data=dat, REML=TRUE)
summary(m1)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: yield ~ nitro + gen + nitro:gen + block + (1 | block:gen) + (1 |
##   block:nitro)
##   Data: dat
##
## REML criterion at convergence: 575.7
##
## Scaled residuals:
##   Min      1Q   Median      3Q      Max
## -1.57028 -0.51573  0.04101  0.51292  1.42868
##
## Random effects:
##   Groups      Name          Variance Std.Dev.
## block:gen   (Intercept) 360200   600.2
## block:nitro (Intercept)  55346   235.3
## Residual                    411648   641.6
## Number of obs: 54, groups: block:gen, 18; block:nitro, 9
```

```

##
## Fixed effects:
##           Estimate Std. Error      df t value Pr(>|t|)
## (Intercept) 2987.278   585.917   18.962  5.098 6.42e-05 ***
## nitro60     1560.333   557.969   22.425  2.796  0.0104 *
## nitro120    3976.333   557.969   22.425  7.126 3.40e-07 ***
## genG2       1362.667   717.332   20.898  1.900  0.0714 .
## genG3        678.000   717.332   20.898  0.945  0.3554
## genG4        487.333   717.332   20.898  0.679  0.5044
## genG5        530.000   717.332   20.898  0.739  0.4682
## genG6       -364.333   717.332   20.898 -0.508  0.6168
## blockBlock R2  874.333   450.225    9.009  1.942  0.0840 .
## blockBlock R3  878.833   450.225    9.009  1.952  0.0827 .
## nitro60:genG2  219.000   740.854   20.000  0.296  0.7706
## nitro120:genG2 -1699.333  740.854   20.000 -2.294  0.0328 *
## nitro60:genG3  312.333   740.854   20.000  0.422  0.6778
## nitro120:genG3 -357.667  740.854   20.000 -0.483  0.6345
## nitro60:genG4  -65.667   740.854   20.000 -0.089  0.9303
## nitro120:genG4 -941.000  740.854   20.000 -1.270  0.2186
## nitro60:genG5  -28.667   740.854   20.000 -0.039  0.9695
## nitro120:genG5 -2066.000  740.854   20.000 -2.789  0.0113 *
## nitro60:genG6 -1053.333  740.854   20.000 -1.422  0.1705
## nitro120:genG6 -4691.667  740.854   20.000 -6.333 3.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 20 > 12.
## Use print(x, correlation=TRUE) or
##   vcov(x)           if you need it

```

Plan à 2 facteurs étudier avec contrôle d'hétérogénéité sur les lignes et les colonnes : carrés gréco-latins ou latins orthogonaux

Ce type de plan permet de prendre en compte 2 facteurs tout en considérant deux contrôles d'hétérogénéité sur les lignes et les colonnes. Ainsi avec J^2 unités expérimentales, le carré gréco-latin permet l'étude des effets simples de 4 facteurs à j niveaux (dont 2 de contrôles).

Ce plan peut être vu comme la superposition de deux plans latins.

```

library (agricolae)

A <- c("1", "2", "3", "4")
B <- c("a", "b", "c", "d")
design_graeco <- design.graeco(trt1 = A, trt2 = B, serie=1, seed=42, kinds = "Super-Duper", randomizat.

design_graeco

##   plots row col A B
## 1    11   1   1 4 a
## 2    12   1   2 1 b

```

```
## 3    13    1    3 3 d
## 4    14    1    4 2 c
## 5    21    2    1 1 d
## 6    22    2    2 4 c
## 7    23    2    3 2 a
## 8    24    2    4 3 b
## 9    31    3    1 3 c
## 10   32    3    2 2 d
## 11   33    3    3 4 b
## 12   34    3    4 1 a
## 13   41    4    1 2 b
## 14   42    4    2 3 a
## 15   43    4    3 1 c
## 16   44    4    4 4 d
```

On peut maintenant représenter graphiquement notre plan d'expérimentation :

```
# Add Row and Col
design_graeco <- design_graeco %>%
  mutate(row = row %>% as.integer,
         col = col %>% as.integer,
         trts = as.factor(paste0(A, "_", B)) )

# Plot field layout
desplot(trts ~ col + row, flip = TRUE,
        text = trts, cex = 1,
        shorten = FALSE,
        data = design_graeco,
        main = "Graeco - Latin Square Design",
        # key.cex = 1,
        show.key=FALSE
)
```

Graeco – Latin Square Design

4_a	1_b	3_d	2_c
1_d	4_c	2_a	3_b
3_c	2_d	4_b	1_a
2_b	3_a	1_c	4_d

Décomposition des degrés de liberté :

- Totale : j^2
- Facteur A : $j - 1$
- Facteur B : $j - 1$
- Lignes : $j - 1$
- Colonnes : $j - 1$
- Résiduelle : $(j - 1)(j - 3)$ (généralement faible)

Avantage :

- Permet d'étudier les effets de 4 facteurs (étudiés ou contrôlés) avec peu d'unités expérimentales.

Inconvénients :

- Egalité du nombre de niveaux des 4 facteurs.
- Peu puissant du fait du faible nombre de degrés de liberté sur la résiduelle (ex avec $j = 4$, le degré de liberté est de 3). Attention, si $j = 3$ alors le degré de liberté est de 0 !

Modèle statistique :

Ici, les blocs colonnes et lignes sont pris en effets fixes. Le modèle est alors une ANOVA :

$$y_{i,j,k,l} = \mu + \alpha_i + \beta_j + L_k + C_l + \varepsilon_{k,l}, \quad \varepsilon_{k,l} \sim N(0, \sigma^2)$$

avec :

- $y_{i,j,k,l}$: l'observation associée au traitement i ce trouvant sur la ligne j et colonne k
- μ l'intercept (moyenne générale)
- α_i l'effet du i ème niveau du facteur A (contrainte d'identifiabilité : $\alpha_1 = 0$ ou $\sum_i \alpha_i = 0$)
- β_j l'effet du j ème niveau du facteur B (contrainte d'identifiabilité : $\beta_1 = 0$ ou $\sum_j \beta_j = 0$)
- L_k l'effet de la ligne k (contrainte d'identifiabilité : $L_1 = 0$ ou $\sum_k L_k = 0$)
- C_l l'effet de la colonne l (contrainte d'identifiabilité : $C_1 = 0$ ou $\sum_l C_l = 0$)
- $\varepsilon_{k,l}$ l'erreur résiduelle associée à la placette de la ligne k et colonne l

On pourra alors tester la pertinence des effets des facteurs A et B .

Exemple : agridat::devries.pine

Graeco-Latin Square experiment in pine

Experiment conducted on Caribbean Pine at Coebiti in Surinam (Long 55 28 30 W, Lat 5 18 5 N). Land was cleared in Jan 1965 and planted May 1965. Each experimental plot was 60m x 60m. Roads 10 m wide run between the rows. Each block is thus 180m wide and 200m deep. Data were collected only on 40m x 40m plots in the center of each experimental unit. Plots were thinned in 1972 and 1975. The two treatment factors (spacing, thinning) were assigned in a Graeco-Latin Square design.

Spacing: A=2.5, B=3, C=3.5. Thinning: Z=low, M=medium, S=heavy.

Field width: 4 blocks x 180 m = 720 m

Field length: 1 block x 200 m = 200 m.

Source : P.G. De Vries, J.W. Hildebrand, N.R. De Graaf. (1978). Analysis of 11 years growth of caribbean pine in a replicated Graeco-Latin square spacing-thinning experiment in Surinam. Page 46, 51. <https://edepot.wur.nl/287590>

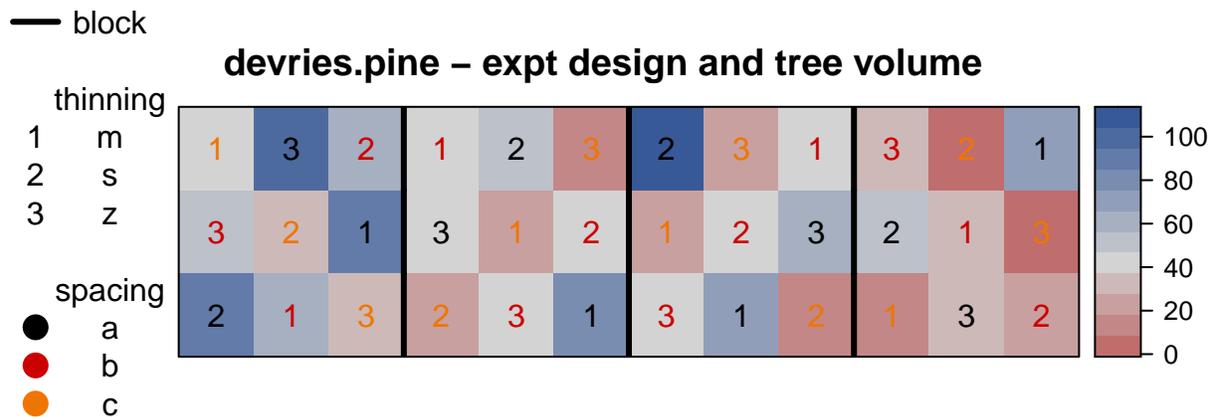
```
agridat::devries.pine
```

```
##      block row col spacing thinning volume growth
## 1      B1   1   1       a         s   91.0   24.4
## 2      B1   2   1       b         z   53.7   28.9
## 3      B1   3   1       c         m   37.8   25.7
## 4      B1   1   2       b         m   59.4   27.9
## 5      B1   2   2       c         s   36.9   27.3
## 6      B1   3   2       a         z   95.3   37.6
## 7      B1   1   3       c         z   34.6   24.3
## 8      B1   2   3       a         m   89.1   36.7
## 9      B1   3   3       b         s   57.6   27.5
## 10     B2   1   4       c         s   18.6   19.9
## 11     B2   2   4       a         z   43.8   30.1
## 12     B2   3   4       b         m   43.0   33.6
## 13     B2   1   5       b         z   45.4   27.5
## 14     B2   2   5       c         m   18.3   25.1
## 15     B2   3   5       a         s   52.7   26.7
## 16     B2   1   6       a         m   77.5   31.9
## 17     B2   2   6       b         s   45.0   28.3
## 18     B2   3   6       c         z   11.0   21.2
## 19     B3   1   7       b         z   42.1   28.2
## 20     B3   2   7       c         m   22.7   24.5
## 21     B3   3   7       a         s  104.9   26.8
## 22     B3   1   8       a         m   74.7   27.0
## 23     B3   2   8       b         s   41.7   25.4
## 24     B3   3   8       c         z   21.9   24.5
```

```
## 25 B3 1 9 c s 8.9 20.9
## 26 B3 2 9 a z 63.7 30.8
## 27 B3 3 9 b m 41.0 28.4
## 28 B4 1 10 c m 9.9 22.6
## 29 B4 2 10 a s 49.9 27.1
## 30 B4 3 10 b z 34.9 29.9
## 31 B4 1 11 a z 36.7 31.8
## 32 B4 2 11 b m 29.1 28.2
## 33 B4 3 11 c s 6.0 24.1
## 34 B4 1 12 b s 22.7 26.7
## 35 B4 2 12 c z 0.6 22.7
## 36 B4 3 12 a m 68.3 34.9
```

```
library(agridat)
data(devries.pine)
dat <- devries.pine

library(desplot)
desplot(dat, volume ~ col*row,
        main="devries.pine - expt design and tree volume",
        col=spacing, num=thinning, cex=1, out1=block, aspect=200/720)
```



Dans cet essai, il y a 4 blocs (répétitions). On va donc prendre en compte un effet bloc et les effets lignes et colonnes seront propres à chaque bloc (effet des lignes et des colonnes différents d'un bloc à un autre).

```
m1 <- lm(volume ~ spacing + thinning + block + block:factor(row) +
          block:factor(col), data=dat)
anova(m1)
```

```
## Analysis of Variance Table
##
## Response: volume
##              Df  Sum Sq Mean Sq F value    Pr(>F)
## spacing        2 16063.7  8031.9  94.7790 4.453e-08 ***
## thinning        2   320.3   160.1   1.8896  0.1935
## block           3  5191.8  1730.6  20.4218 5.249e-05 ***
## block:factor(row) 8   945.0   118.1   1.3939  0.2913
## block:factor(col) 8   784.7    98.1   1.1575  0.3953
## Residuals      12  1016.9    84.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Références

Livres :

- Cochran and Cox. (1992). Experimental design. Second edition. Wiley Classics Library Edition published
- Dieter Rasch, Jurgen Pilz, Rob Verdooren and Albrecht Gebhardt (2011). Optimal Experimental Design with R. Taylor and Francis Group, LLC CRC Press is an imprint of Taylor and Francis Group, an Informa business.
- Fisher, R. A., Fisher, R. A., Genetiker, S., Fisher, R. A., Genetician, S., Britain, G., . . . & Généticien, S. (1966). The design of experiments (Vol. 21). Edinburgh: Oliver and Boyd.
- McCulloch, C. E., & Searle, S. R. (2004). Generalized, linear, and mixed models. John Wiley & Sons.
- Gomez, K. A., & Gomez, A. A. (1984). Statistical procedures for agricultural research. John wiley & sons.
- Philippeau, G. (1977). Théorie des plans d'expérience: application à l'agronomie. ITCF.
- Pinheiro, J., & Bates, D. (2000). Mixed-effects models in S and S-PLUS. Springer science & business media.
- Rencher, A. C., & Schaalje, G. B. (2008). Linear models in statistics. John Wiley & Sons.
- Robert O. Kuehl. 2nd ed., Duxbury, (2000). Design of Experiments.
- Schwarz, C. J. (2011). Sampling, regression, experimental design and analysis for environmental scientists, biologists, and resource managers. Department of Statistics and Actuarial Science, Simon Fraser University, 57.

Articles :

- Piepho, H. P., Büchse, A., & Emrich, K. (2003). A hitchhiker's guide to mixed models for randomized experiments. Journal of Agronomy and Crop Science, 189(5), 310-322.
- Piepho, H. P., & Edmondson, R. N. (2018). A tutorial on the statistical analysis of factorial experiments with qualitative and quantitative treatment factor levels. Journal of Agronomy and Crop Science, 204(5), 429-455.