

# Randomisation

Mettre votre nom

27-10-2021

# Randomisation

Quelques types courants de randomisation

Bonnes pratiques

Quelques designs expérimentaux

Quelques limites

# Randomisation

# Randomisation de l'assignation de traitement

Commencez avec un groupe d'unités que vous pouvez observer directement : comme les personnes qui se sont présentées à un laboratoire, les villages d'une région, les personnes prêtes à participer à une enquête.

- ▶ Chaque observation a une probabilité connue d'assignation au traitement comprise entre 0 et 1.
  - ▶ Aucune unité de l'échantillon expérimental n'est assignée au traitement ou au contrôle avec certitude.
- ▶ Les unités peuvent varier dans leur probabilité d'assignation au traitement.
  - ▶ Par exemple, la probabilité peut varier selon le groupe : les femmes peuvent avoir une probabilité de 75% d'être assignées au traitement.
  - ▶ Les probabilités d'assignation peuvent même varier d'une unité à l'autre **tant que vous connaissez la probabilité pour chaque unité**, bien que cela compliquerait votre analyse.

# Assignment aléatoire $\neq$ échantillonnage aléatoire

- ▶ Échantillonnage aléatoire (à partir d'une population) : sélectionner des sujets d'une population dans votre échantillon avec probabilité connue. Vous ne pouvez pas observer directement l'ensemble de la population, donc vous tirez un échantillon.
- ▶ Randomisation (du traitement) : assigner des sujets d'un groupe existant de sujets avec une probabilité connue à des conditions expérimentales.
  - ▶ Vous devez être en mesure de voir les résultats et d'assigner un traitement à chaque sujet du groupe
  - ▶ Cette randomisation du traitement peut être combinée avec tout type d'échantillon (échantillon aléatoire, échantillon de convenance, etc.) ou même avec l'ensemble de la population (tous les hôpitaux, tous les juges).

# Assignment aléatoire $\neq$ échantillonnage aléatoire

- ▶ La manière dont vous recrutez votre groupe initial (ou pool expérimental) est importante :
  - ▶ un grand groupe initial signifie une grande expérience et plus de puissance statistique pour détecter de petits effets.
  - ▶ un groupe initial qui est un échantillon aléatoire d'une population connue vous aide à faire valoir que vos effets pourraient être identiques ou similaires si vous faisiez cette expérience avec un autre échantillon de cette population.
  - ▶ un pool expérimental constitué d'un échantillon aléatoire pourrait également aider à soutenir l'argument selon lequel l'effet devrait être similaire si vous étendiez l'intervention à l'ensemble de la population — en fonction des facteurs déterminant un équilibre global.

# La randomisation est étroitement liée à votre design de recherche

- ▶ Voir module sur [le design de recherche](#)
- ▶ N'oubliez pas que vous devez définir et justifier votre condition de contrôle :
  - ▶ intervention absente
  - ▶ un placebo

# Niveau de traitements et résultats

- ▶ Le traitement peut être assigné au niveau de l'individu, du groupe, de l'institution, de la communauté, de la période temporelle. . .
- ▶ Vous pouvez être limité dans le niveau auquel vous pouvez *assigner* le traitement et *mesurer* les résultats.
  - ▶ Exemple : Traitement au niveau de la classe, mais résultats au niveau de l'élève.
  - ▶ Exemple : Traitement au niveau du district, mais résultats au niveau communautaire.
- ▶ Le niveau d'assignation du traitement et le niveau de mesure des résultats affectent ce que votre étude peut démontrer.



## Quelques types courants de randomisation

# Randomisation simple (tirage au sort)

- ▶ Pour chaque unité, lancez une pièce pour voir si elle sera traitée. Ensuite, vous mesurez les résultats au même niveau que la pièce.
- ▶ Les pièces ne doivent pas nécessairement être équitables (50-50), mais vous devez connaître la probabilité d'assignation du traitement.
- ▶ Vous ne pouvez pas garantir un nombre précis d'unités traitées et d'unités de contrôle.
- ▶ Exemple : Si vous avez 6 unités et que vous lancez une pièce équitable pour chacune, vous avez environ 3% de chances d'assigner **toutes** les unités au traitement ou d'assigner **toutes** les unités au contrôle.

# Exemple de code pour la randomisation simple I

```
# définir une "graine" aléatoire pour rendre cela reproductible  
set.seed(12345)
```

```
# définir une taille d'échantillon  
N <- 200
```

```
# Générer l'assignation aléatoire simple  
# (Remarquez que dans une expérience, nous avons un seul  
# essai, donc taille=1)  
# Notre objet avec N personnes au total s'appelle simple.ra  
simple.ra <- rbinom(n = N, size = 1, prob = .5)
```

```
# 112 personnes se sont retrouvées dans le groupe de traitement  
sum(simple.ra)
```

```
[1] 112
```

# Exemple de code pour la randomisation simple II

```
# vous pouvez également utiliser le package randomizr
library(randomizr)

# pour la reproductibilité
set.seed(23456)

# L'assignation aléatoire simple utilise la fonction simple_ra
# Notre objet avec N personnes au total est appelé traitement
treatment <- simple_ra(
  N = N, # taille totale de l'échantillon
  prob = .5 # probabilité de recevoir un traitement
)
sum(treatment)
```

[1] 96

# Ramdomisation complète (tirage d'une urne)

- ▶ Un nombre fixe de  $m$  sur  $N$  d'unités est assigné au traitement.
- ▶ La probabilité qu'une unité soit assignée au traitement est de  $m/N$ .
- ▶ C'est comme avoir une urne avec  $N$  boules, dont  $m$  sont marquées comme traitement et  $N - m$  sont marquées comme contrôle. La loterie publique utilise cette méthode.

# Exemple de code pour la randomisation complète I

```
# définir la taille d'échantillon N  
N <- 200  
# définir le nombre d'unités traitées m  
m <- 100  
  
# crée un vecteur de m 1's et N-m 0's  
complete.ra <- c(rep(1, m), rep(0, N - m))  
  
# Et ajoute hasard en utilisant sample()  
# La valeur par défaut est un tirage sans remplacement  
  
set.seed(12345) # pour la reproductibilité  
complete.ra <- sample(complete.ra)  
  
sum(complete.ra)
```

```
[1] 100
```

# Exemple de code pour la randomisation complète II

```
# vous pouvez également utiliser le package randomizr
library(randomizr)

# pour la reproductibilité
set.seed(23456)

# Assignment aléatoire complète:
treatment <- complete_ra(
  N = 200, # taille totale de l'échantillon
  m = 100
) # quantité assignée au traitement

sum(treatment)
```

```
[1] 100
```

```
# Regardez ce qui se passe si vous ne spécifiez pas m !
```

# Randomisation par bloc (ou stratifiée) I

- ▶ Nous créons des blocs d'unités et randomisons séparément dans chaque bloc. Nous faisons des mini-expériences dans chaque bloc.
  - ▶ Exemple : bloc = district, unités = communautés. Nous randomisons le traitement au niveau communautaire **au sein du district** et mesurons également les résultats au niveau communautaire.
- ▶ Les blocs qui représentent un sous-groupe significatif peuvent vous aider à appréhender comment les effets peuvent différer selon le sous-groupe.
  - ▶ En contrôlant le nombre de sujets par sous-groupe, vous vous assurez d'avoir suffisamment de sujets dans chaque groupe.
  - ▶ Particulièrement utile lorsque vous avez un groupe rare — par hasard, vous pourriez en avoir très peu dans le traitement ou le contrôle même avec une assignation aléatoire (ou vous pourriez avoir un certain déséquilibre).



# Randomisation par bloc (ou stratifiée) II

- Les blocs homogènes sur un résultat donné augmentent la précision de l'estimation de ce résultat par rapport à l'expérience sans blocs. (Nous aborderons cela plus en détail dans la section analyse de puissance statistique).

# Randomisation par grappe (cluster) I

- ▶ Une grappe est un **groupe d'unités**. Dans une étude randomisée par grappe, toutes les unités de la grappe sont assignées au même statut de traitement.
- ▶ Utilisez la randomisation par grappe si l'intervention doit fonctionner au niveau de la grappe.
  - ▶ Par exemple, si l'intervention concerne les cours de récréation des écoles, alors l'école est l'unité d'assignation même si la santé des élèves peut être un résultat mesuré au niveau individuel des élèves.
- ▶ Un nombre réduit de grappes nuit à votre capacité à détecter les effets du traitement et conduit à des  $p$ -valeurs, des intervalles de confiance (ou même des estimations) trompeurs. Cela dépendra de la corrélation intra-grappe (intra cluster correlation, ICC ou  $\rho$ ).

## Randomisation par grappe (cluster) II

- ▶ Un  $\rho$  plus élevé est pire:
  - ▶ si  $\rho = 0$  alors le village n'a pas d'importance pour le comportement des individus.
  - ▶ si  $\rho = 1$  alors chaque personne dans le village donnerait exactement la même réponse. Avoir une autre personne de ce village ne vous donne pas d'informations supplémentaires puisque son résultat est identique à la personne que vous aviez déjà.
- ▶ Pour le même nombre d'unités, avoir **plus de grappes** avec moins d'unités par grappe peut aider.
- ▶ Compromis entre les effets de débordement et la puissance statistique.
- ▶ Si vous ne souhaitez pas une expérience avec 10 unités, alors vous ne devriez pas vous contenter d'une expérience avec 10 groupes de 100 unités. La taille effective de l'échantillon de cette expérience aléatoire par grappe est comprise entre 10 et  $10 \times 100 = 1000$ , mais plus proche de 10 plus  $\rho$  est élevé.

# Vous pouvez combiner blocs et grappes

- ▶ Vous pouvez avoir des grappes dans des blocs.
  - ▶ Exemple : bloc = district, grappe = communautés, unités = individus. Vous mesurez les résultats au niveau individuel.
  - ▶ Exemple : bloc = province, grappe = district, unités = communautés. Vous mesurez les résultats au niveau communautaire.
- ▶ Vous ne pouvez pas avoir de blocs dans les grappes.
- ▶ Pour la randomisation des blocs et grappes, vous pouvez utiliser `block_ra` et `cluster_ra` dans le package `randomizr` de R.
- ▶ Pour les designs plus compliqués, vous pourriez trouver `DeclareDesign` utile. (<https://declaredesign.org>)

## Bonnes pratiques

# Bonnes pratiques : la répliquabilité

- ▶ Guide des méthodes EGAP sur la randomisation (<https://egap.org/resource/10-things-to-know-about-randomization/>)
- ▶ Définissez une “graine” aléatoire et sauvegardez votre code et votre assignation aléatoire
- ▶ Vérifier
- ▶ Parfois transparence accrue => répliquabilité

# Bonnes pratiques : l'équilibre

- ▶ Vérifiez l'équilibre global avec un test D2 (D-square test) en utilisant xBalance dans le package RIttools (**Hansen and Bowers (2008)**) (inférence de randomisation d'un grand échantillon):

```
xBalance(treatment ~ x1 + x2, data = dat, report = "chisquare")
```

- ▶ Voir aussi le package coin independence\_test pour la version basée sur la permutation
- ▶ Utiliser un test F pour une régression entre l'assignation de traitement à gauche et les covariables à droite :

```
anova(lm(treatment ~ 1, data = dat),  
      lm(treatment ~ x1 + x2 + x3, data = dat),  
      test = "F"  
)
```

## Bonnes pratiques : l'équilibre

- ▶ L'assignation aléatoire nous donne, en espérance, l'**équilibre global** sur les nombreuses covariables. Cela ne garantit pas que toutes les covariables au traitement seront 0. En fait, dans une petite expérience, les amplitudes de déséquilibre peuvent être élevées même si la randomisation s'est parfaitement déroulée.
- ▶ Vous regardez les tests t des covariables un par un. Par hasard, vous pourriez obtenir des différences statistiquement significatives sur une variable. Si vous vérifiez l'équilibre sur 100 variables, vous rejetterez l'hypothèse nulle d'absence d'effet pour 5 d'entre elles, même s'il n'y a réellement aucune relation.



## Quelques designs expérimentaux

# Accès

- ▶ Sélectionnez de manière aléatoire un groupe de traitement par le biais d'une loterie ou d'un mécanisme équivalent, qui randomise l'**accès** au programme.
- ▶ Utile quand on n'a pas assez de moyens pour traiter tout le monde.
- ▶ Parfois, certaines unités (peuples, communautés) doivent avoir accès à un programme.
  - ▶ Par exemple : une organisation partenaire ne veut pas risquer qu'une communauté vulnérable *ne bénéficie pas* d'un programme (elle veut la garantie qu'elle sera toujours traitée).
  - ▶ Vous pouvez exclure ces unités de l'expérience et procéder à une assignation aléatoire parmi les unités restantes qui ont une probabilité d'assignation strictement comprise entre 0 et 1 (non inclus).

# Accès différé (Accès graduel ou liste d'attente)

- ▶ Randomiser le *timing* d'accès au programme.
- ▶ Souvent, vous n'avez pas la capacité de mettre en œuvre le traitement dans de nombreux endroits à la fois.
- ▶ Lorsqu'une intervention peut ou doit être déployée par étapes, vous pouvez randomiser l'ordre de traitement des unités.
- ▶ Votre groupe de contrôle sont les unités pas encore traitées.
- ▶ Attention : la probabilité d'assignation au traitement variera dans le temps car les unités assignées au traitement à des stades antérieurs ne sont pas éligibles pour être assignées à un traitement à des stades ultérieurs.

# Assignment factorielle ou croisée

- ▶ Le design factoriel permet de tester plus qu'un traitement.
- ▶ Vous pouvez analyser un traitement à la fois.
- ▶ Ou leurs combinaisons.
- ▶ Exemple:

	$X_1 = 0$	$X_1 = 1$
$X_2 = 1$	A	C
$X_2 = 0$	B	D

Nous pourrions nous concentrer sur un estimateur comme  $\mathbb{E}[Y(X_1 = 1, X_2 = 1)] - \mathbb{E}[Y(X_1 = 0, X_2 = 0)]$ .

# Incitations

- ▶ Randomisez l'**incitation** à suivre le traitement, comme une invitation ou une subvention pour participer à un programme.
- ▶ Utile lorsque vous ne pouvez pas forcer un sujet à participer.
- ▶ Estimandes :
  - ▶ l'effet moyen du traitement (average treatment effect, ATE) de l'incitation pour votre échantillon expérimental.
  - ▶ l'ATE de participation (et non d'incitation) pour les unités qui participeraient lorsqu'encouragées et ne participeraient pas sans incitation (les conformistes).
- ▶ Analyse des variables instrumentales pour l'ATE des conformistes, avec l'assignation comme instrument. Notez la restriction d'exclusion.

## Quelques limites

## Limites aux études aléatoires

- ▶ Éthique — ce genre de manipulation est-il éthique ? Parfois non.
- ▶ Doit être fait en temps réel, en amont du déploiement de l'intervention.
- ▶ Flexibilité réduite pour une organisation partenaire (problème pour toute évaluation prospective).
- ▶ Limites à la taille du pool expérimental.
- ▶ Coût.
- ▶ La contrainte de puissance statistique — vous avez besoin de beaucoup d'unités (problème pour de nombreuses approches statistiques).
- ▶ Violations des hypothèses clés (effets de débordement ; violation de la deuxième hypothèse clé de l'**inférence causale**).
- ▶ Validité externe (problème pour toute évaluation et les sciences sociales en général).

# Références

Hansen, Ben B., and Jake Bowers. 2008. "Covariate Balance in Simple, Stratified and Clustered Comparative Studies." *Statistical Science* 23 (2): 219–36.