

Méthodologie statistique

M 2018/01

Estimation avec le score de propension
sous 

Simon Quantin
(DMCSI)

Document de travail



Institut National de la Statistique et des Études Économiques

M 2018/01

Estimation avec le score de propension sous 

Simon Quantin (DMCSI)

Ce document a bénéficié des conseils (et encouragements !) précieux de Pauline Givord tout au long de son élaboration. Qu'elle en soit ici grandement remerciée.


Merci également à Elise Coudin qui en a réalisé une relecture attentive le rendant, je l'espère, d'une lecture agréable.

Je reste seul responsable des erreurs, approximations ou omissions pouvant subsister dans ce document.

Estimation avec le score de propension sous

Simon Quantin*

Résumé

Ce document présente les principales méthodes économétriques utilisant le score de propension pour comparer deux groupes en ajustant des effets de compositions observables. Plus précisément, il se veut un complément pratique au document méthodologique de [Givord \(2010\)](#) sur les méthodes d'appariement, de stratification et de pondération par l'inverse de la probabilité de traitement. Après avoir rappelé comment estimer le score de propension, et les indicateurs statistiques classiquement utilisés pour vérifier sa propriété équilibrante, un chapitre dédié à chaque méthode rappelle la démarche correspondante, les principales recommandations pratiques formulées par les nombreux travaux académiques, et expose leur implémentation avec le logiciel .

Mots-clés : Évaluation des politiques publiques, score de propension, appariement, stratification, pondération

Classification JEL : C01, C14, C21, C52

* INSEE. Auteur correspondant

Adresse : 88, avenue Verdier, CS 70058 92541 Montrouge cedex

Téléphone : (+33) 1 87 69 55 26

E-mail : simon.quantin@insee.fr

Table des matières

Introduction

1 Score de propension et propriété équilibrante

- 1.1 L'estimation du score de propension
 - 1.1.1 Le choix des covariables 2
 - 1.1.2 Estimation sous R 2
- 1.2 Vérifier la propriété équilibrante
 - 1.2.1 Indicateurs statistiques 3
 - 1.2.2 Ajustement pour traiter du biais résiduel 4

2 Appariement sur le score de propension

- 2.1 Appariement sur le score de propension
 - 2.1.1 Choisir le nombre de voisins : *Ratio matching* 5
 - 2.1.2 Appariement exact et/ou caliper 6
 - 2.1.3 Avec ou sans remise 6
- 2.2 Estimation après appariement
 - 2.2.1 Estimateur simple après appariement 7
 - 2.2.2 Estimateur apparié corrigé du biais 7
- 2.3 Mise en œuvre sous R
 - 2.3.1 Appariement sur le score de propension 8
 - 2.3.2 Vérifier la propriété équilibrante après appariement 9
 - 2.3.3 Représentations graphiques univariées et support commun 13
 - 2.3.4 Estimation de l'effet du traitement 17

3 Stratification

- 3.1 L'estimation par stratification
 - 3.1.1 Constitution des strates 22
 - 3.1.2 Précision de l'estimation 22
- 3.2 Mise en œuvre sous R
 - 3.2.1 Estimation du score de propension et construction des strates 23
 - 3.2.2 Vérifier la propriété équilibrante au sein de chaque strate 23
 - 3.2.3 Estimation de l'effet du traitement 28

4 Ajustement par pondération

- 4.1 Estimateurs pondérés par l'inverse de la probabilité de traitement
 - 4.1.1 Estimateur pondéré simple et doublement robuste 31
 - 4.1.2 Analyse des pondérations calculées 32
- 4.2 Mise en œuvre sous R

4.2.1	Calcul des pondérations après estimation du score de propension	33
4.2.2	Vérifier la propriété équilibrante après ajustement par pondération	33
4.2.3	Estimation de l'effet du traitement	35

Conclusion

Bibliographie

Introduction

Ce document se veut un complément pratique au document méthodologique de [Givord \(2010\)](#), *Méthodes économétriques pour l'évaluation de politiques publiques*. En effet, il explicite l'implémentation sous [R](#) de trois méthodes utilisant le score de propension : l'appariement sur le score de propension, la stratification et l'ajustement par pondération. Ces méthodes, désormais utilisées dans des secteurs variés (exploitation des données multimodes, *sample matching*, etc.), sont classiques dans la littérature sur l'évaluation des politiques publiques. Elles permettent de comparer les réalisations d'une variable, observées dans deux échantillons distincts, en ajustant des différences de composition en termes de caractéristiques observables entre ces échantillons. Cet ajustement vise à réduire le biais engendré par les effets de sélection liés aux caractéristiques observables des bénéficiaires¹ pour estimer l'effet causal d'un traitement.

A priori différentes, leur mise en œuvre repose toutefois sur 4 étapes similaires ([Givord, 2010](#)) qui structurent pour chacune le travail du chargé d'études :

1. choisir les caractéristiques observables à retenir et estimer le score de propension correspondant,
2. implémenter la méthode choisie,
3. analyser la propriété équilibrante du score de propension en réalisant si besoin à nouveau les étapes 1 et 2,
4. estimer enfin l'effet de la mesure étudiée.

Comme le font remarquer [Imbens \(2014\)](#) et [Stuart \(2010\)](#), les trois premières étapes n'utilisent nullement la variable analysée, l'analyse à proprement parler n'intervenant qu'après. Ces étapes consistent, une fois estimé le score de propension, à réduire l'échantillon étudié en éliminant plusieurs unités avec l'appariement, à le partitionner en strates ou à attribuer une pondération à chaque observation, afin d'améliorer la similitude des distributions des groupes traités et de contrôle vis-à-vis des covariables. La dernière étape estime l'effet moyen sur les traités (*Average Treatment effect on Treated*) ou sur l'ensemble de la population (*Average Treatment Effect*) à partir de la base de données constituée.

Plusieurs packages peuvent être utilisés sous [R](#) pour estimer l'effet d'un traitement avec le score de propension mais tous se structurent globalement autour de cette démarche en dédiant des fonctions séparées à la constitution de la base de données (appariement, stratification ou calcul des pondérations), à la vérification de la propriété équilibrante et au calcul de l'effet de la mesure.

Le document est structuré comme suit. Un premier chapitre revient sur l'estimation du score de propension et rappelle les indicateurs statistiques utilisés pour vérifier sa propriété équilibrante. Ensuite un chapitre, découpé en deux parties, est consacré à chacune des méthodes. À chaque fois, la

1. Mais il n'est cependant pas suffisant pour réduire les biais liés à la présence de caractéristiques inobservables.

première partie présente succinctement la démarche correspondante², le(s) estimateur(s) de l'effet moyen du traitement et celui de leur(s) variance(s). La deuxième partie détaille leur mise en œuvre sous **R**. Pour cela, nous utilisons deux packages spécialement dédiés à l'estimation d'un effet par score de propension³. Le package **Matching** (Sekhon, 2011) permet de mettre en œuvre l'appariement sur score de propension. L'estimation par stratification et par pondération ne nécessitent, eux, pas de packages dédiés. Le package **cobalt** (Greifer, 2016) sera utilisé, quelle que soit la méthode implémentée, pour l'analyse statistique et graphique de la propriété équilibrante du score de propension.

Tous les exemples présentés s'appuient sur la base `lalonde` disponible dans le package **cobalt**. Cette base de données, couramment utilisée pour illustrer les méthodes d'appariement, provient de l'article de Dehejia et Wahba (1999). Elle permet d'estimer l'impact d'un programme de formation (*National Supported Work Demonstration*) sur les revenus des bénéficiaires en 1978. Elle comporte 614 observations et 9 variables, certaines qualitatives (comme la situation maritale), d'autres continues comme les revenus en 1974 et 1975. Nous reproduisons ci-dessous la syntaxe du code **R** pour l'utiliser.

```
library(cobalt)

## Chargement des données : Lalonde Data from cobalt
data(lalonde, package="cobalt")
mybase <- lalonde
## Construction d'indicateurs à partir de la variable
## qualitative RACE
mybase <- splitfactor(data = mybase, var.name = "race",
                      replace = FALSE,
                      drop.level=NULL,
                      drop.first=FALSE)
## Changement d'unité des variables de revenus (en milliers de dollars)
mybase$re74 <- mybase$re74/1000
mybase$re75 <- mybase$re75/1000
## Créations d'indicateurs d'absence de revenus
mybase$u74 <- ifelse(test=(mybase$re74==0), 1, 0)
mybase$u75 <- ifelse(test=(mybase$re75==0), 1, 0)
## Création variable factorielle de groupe
mybase$group <- factor(x=mybase$treat, levels=c(0,1),
                      labels=c("Control", "Treated"))
```

Comme le montre les résultats ci-dessous, les moyennes (ou proportions) des différentes covariables diffèrent entre le groupe traité et non traité. Les deux groupes ne sont donc pas en moyenne directement comparables, ce que l'appariement sur score de propension cherchera à dépasser.

```
var <- names(mybase)[-c(1,4,8,15)]
round(t(sapply(mybase[var], function(x) tapply(x, mybase$treat, mean))), digits = 3)

      0      1
```

2. Pour les aspects théoriques, le lecteur est invité à se référer au document méthodologique de Givord (2010).

3. Toutefois, pour certaines méthodes, des fonctions d'autres packages, non explicitement dédiés à l'estimation à partir du score de propension, pourront être appelées. Celles-ci seront explicitement indiquées dans le texte.

age	28.030	25.816
educ	10.235	10.346
married	0.513	0.189
nodegree	0.597	0.708
re74	5.619	2.096
re78	6984.170	6349.144
race_black	0.203	0.843
race_hispan	0.142	0.059
race_white	0.655	0.097
u74	0.261	0.708
u75	0.312	0.600

Par ailleurs, nous utiliserons aussi une dataframe composée de deux vecteurs, `old` et `new`, permettant d'affecter un libellé aux variables pour les représentations graphiques du package **cobalt**.

```
## Définition de nouveaux noms de variables
## pour les représentations graphiques
new.names <- data.frame(
  old = c("age", "educ", "race_black",
          "race_hispan", "race_white", "married",
          "nodegree", "re74", "re75", "u74",
          "u75", "pscore"),

  new = c("Age", "Years of Education", "Black",
          "Hispanic", "White", "Married", "No Degree Earned",
          "Earnings 1974", "Earnings 1975", "Employed in 1974",
          "Employed in 1975", "Propensity Score"))
```

Pour finir, soulignons que ce document ne discute pas la pertinence des résultats des exemples proposés. En effet, il s'agit surtout de proposer des exemples de syntaxe sous **R** pour chacune des méthodes. Néanmoins les principales recommandations issues des nombreux articles académiques consacrés aux différentes approches sont présentées tout au long du document.



1 Score de propension et propriété équilibrante

Rosenbaum et Rubin (1983) définissent le score de propension comme *la probabilité d'être traitée* conditionnellement aux variables observées, c'est-à-dire $p(X) = P[T = 1 | X]$ où T est une indicatrice permettant d'identifier les personnes effectivement traitées. Dans cet article, ils montrent notamment qu'apparier ou stratifier des unités traitées et non-traitées avec le score de propension permet d'éliminer le biais dû aux différences de caractéristiques *observables*, notées ici X , (mais pas *inobservables*) pour comparer les deux groupes. En effet, le score de propension « équilibre » les distributions des covariables (on parle alors de « propriété équilibrante » ou *balanced function*) : pour chaque valeur du score de propension, la distribution des covariables X dans le groupe traité est identique à celle du groupe de contrôle ($T \perp X | p(X)$) ce qui permet de les comparer en ayant ainsi ajusté des différences de composition. Toutefois, on notera que l'utilisation du score de propension ne garantit pas, en tant que tel, que l'effet du traitement estimé soit un effet *causal*. En effet, pour cela, il faut notamment que le processus de sélection des unités traitées ne s'appuie pas sur des covariables autres que celles observées et qui pourraient influencer l'effet du traitement (hypothèse dite d'indépendance conditionnelle, cf. Givord (2010)).

En pratique, le score de propension n'est souvent pas connu et doit ainsi être estimé en fonction des covariables disponibles. Dans une première partie, nous rappelons comment choisir les covariables nécessaires à son estimation et comment les intégrer dans un modèle logistique¹. La deuxième partie est consacrée à la présentation des indicateurs et représentations graphiques les plus couramment utilisés pour vérifier la propriété équilibrante du score de propension. Tous sont implémentés par le package **cobalt** qui peut être utilisé quelle que soit la méthode d'estimation de l'effet du traitement retenue.

1.1 L'estimation du score de propension

La propriété équilibrante du score de propension assure que les unités traitées et non traitées ayant un score de propension identique présentent des distributions des caractéristiques observables similaires. Cette propriété n'est cependant vérifiée que pour le *vrai* score qui n'est pas observé et doit donc être estimé. La qualité de l'estimation de l'effet moyen du traitement dépend donc étroitement de celle du score de propension.

1. D'autres méthodes (notamment non-paramétriques) peuvent aussi être implémentées sous  comme l'approche par modèles généralisés additifs (*generalized additive models*), arbre de classification (*classification trees*), analyse discriminante ou modèles généralisés augmentés (voir Ridgeway et al., 2006, et le package **twang** sous  pour une mise en œuvre des *generalized boosted model*), ou modèles de réseaux de neurones (*neural network model*). Elles ne sont toutefois pas présentées dans ce document.

Au-delà, la pertinence des méthodes présentées dans ce document s'appuient sur l'*existence* pour chaque unité traitée, d'une unité non traitée ayant un score de propension identique ou proche. Il faut donc aussi que la distribution du score de propension dans les unités traitées « recoupe » largement celle dans les unités du groupe de contrôle; on parle alors de *support commun large*.

En pratique, la spécification du modèle est donc un processus itératif qui s'appuie sur l'obtention d'un support commun suffisamment large et le respect de la propriété équilibrante du score de propension pour les covariables.

1.1.1 Le choix des covariables

L'utilisation du score de propension vise à obtenir, par appariement, stratification ou pondération, deux groupes (l'un traité, l'autre non) ayant des caractéristiques similaires afin de les comparer sur une dimension donnée. Classiquement, il est donc recommandé d'introduire dans la procédure d'estimation toutes les variables corrélées au traitement et à la variable que l'on souhaite étudier (Rubin et Thomas, 1996; Rubin, 2001; Heckman *et al.*, 1998). Cette recommandation n'est évidemment pas simple à suivre, puisqu'il n'est pas certain que le statisticien observe l'ensemble des caractéristiques souhaitées dans les données dont il dispose.

Ceci étant, on notera tout d'abord qu'une variable trop corrélée avec le fait d'être traité peut conduire à observer un support commun réduit, voire complètement disjoint dans le cas d'un processus d'attribution du traitement complètement déterministe en les covariables observées². De plus, les variables retenues ne doivent pas être en retour affectées par le traitement car on ne pourra isoler ces effets de rétroactions indirectes de l'effet direct que l'on cherche à estimer³. Cela peut se faire soit par des variables dont on a de bonnes raisons de penser qu'elles sont indépendantes du traitement, soit par des variables retardées. Enfin, les propriétés des estimateurs de l'effet moyen reposent souvent sur la convergence de l'estimateur du score de propension, et l'on cherchera donc à modéliser le lien entre la probabilité d'être traité et les covariables sous la forme la plus flexible possible, dans la mesure où la taille de l'échantillon le permet.

1.1.2 Estimation sous R

L'estimation (ici paramétrique) du score de propension est un préalable commun aux trois méthodes présentées. Dans ce document, elle s'appuie sur l'estimation d'un modèle de régression logistique, réalisée avec la fonction `glm`.

Code R: Estimation du score de propension

```
ps.logit <- glm(treat ~ re74 + u74 + re75 + u75 + educ + race +
               married + nodegree + age + I(age^2) +
               nodegree:educ + re74:nodegree + u75:educ,
               family = binomial(link="logit"),
               data = mybase)
```

2. Dans ce cas, il conviendra de réfléchir à une autre méthode d'estimation de l'effet moyen, par exemple par les régressions sur discontinuités (cf. Givord, 2010)

3. voir Givord (2010) pour le choix des variables de contrôle respectant l'hypothèse d'indépendance conditionnelle.

```
mybase$pscore <- ps.logit$fitted
```

Dans l'exemple ci-dessus, la probabilité d'être traitée (`treat`) dépend de plusieurs covariables prises individuellement (par exemple `re74`, `u74`), de leur carré (`I(age^2)`), mais aussi de leurs interactions deux à deux (`nodegree:educ`). Le score de propension estimé (`ps.logit$fitted`) est ensuite stocké dans la base initiale (`mybase$pscore`).

1.2 Vérifier la propriété équilibrante

Dans le cas de la méthode par appariement, vérifier la propriété équilibrante du score de propension repose sur une comparaison des distributions des covariables, *dans l'échantillon apparié* des unités traitées et non traitées (éventuellement pondérées dans le cas de l'appariement avec remise). Pour une estimation par stratification, la comparaison est effectuée *au sein de chaque strate*. Enfin, pour un ajustement par les pondérations, celle-ci est menée *sur les données pondérées*. Dans tous les cas, l'attention portée à l'analyse des distributions des covariables est une étape déterminante et plusieurs éléments de diagnostic ont été proposés à cette fin.

1.2.1 Indicateurs statistiques

La différence standardisée de moyennes est l'un des éléments de diagnostic les plus fréquemment utilisés depuis son apparition dans un article de [Rosenbaum et Rubin \(1985\)](#). Elle correspond à la différence des moyennes d'une variable dans le groupe traité et non traité, rapportée à une mesure de dispersion; la variance empirique dans le groupe traité si l'effet moyen du traitement sur les traités est évalué ou la variance empirique *poolée* dans le cas de l'effet moyen du traitement sur toutes les unités⁴.

Formellement, dans le cas de l'effet moyen du traitement sur toutes les unités, pour une variable continue, la différence standardisée de moyennes correspond à :

$$\frac{\bar{X}_t - \bar{X}_c}{\sqrt{\frac{s_t^2 + s_c^2}{2}}} \quad (1.1)$$

où \bar{X}_t et \bar{X}_c désignent les moyennes empiriques de la covariable X respectivement dans le groupe traité et dans le groupe de contrôle, et s_t^2 et s_c^2 les variances empiriques correspondantes.

L'approche par différences standardisées présente plusieurs avantages. En premier lieu, elle ne dépend pas de la taille de l'échantillon et est donc pertinente, dans le cas de l'appariement sur score de propension, pour comparer son apport, en moyenne, sur les différences de caractéristiques observables avant (i.e. sur toutes les unités disponibles) et après (i.e. en se restreignant aux seules unités appariées). Par ailleurs, elle est aussi plus appropriée que les tests de significativité sur la différence de moyennes. En effet, l'appariement réduisant la taille de l'échantillon, les tests perdent de leur puissance et rejettent donc plus souvent l'hypothèse de moyennes différentes dans le groupe traité et non traité ([Imai et al., 2008](#)). Enfin, la différence standardisée permet la comparaison des résultats

4. [Stuart \(2010\)](#) préconise l'utilisation de la variance empirique de l'échantillon non ajusté (avant appariement par exemple) pour calculer la différence standardisée. L'idée est d'éviter de constater une augmentation de la différence standardisée alors même que la différence de moyenne et la variance de l'échantillon diminuent suite à l'appariement par exemple. C'est ce qui est implémenté dans le package **cobalt** présenté dans ce document.

pour des variables présentant des unités de mesure différentes.

Pour les variables qualitatives, on étudiera simplement la différence des proportions. En effet, dans le cas des variables binaires, la variance est une fonction de la proportion de telle sorte que l'on peut observer pour des différences de proportions identiques, des différences standardisées très différentes.

Les covariables sont non équilibrées dès lors que la différence standardisée ou la différence de proportions dans le cas des variables qualitatives est élevée. Il n'existe pas de valeur faisant consensus au-delà de laquelle il faut considérer les écarts comme importants. [Ho et al. \(2007\)](#) suggèrent néanmoins que les différences doivent être d'autant plus faibles que la covariable considérée est fortement corrélée avec la variable de résultat.

La propriété équilibrante du score de propension ne se limite toutefois pas à la seule moyenne et concerne toute la distribution des covariables⁵. Il est donc recommandé de *représenter graphiquement* la distribution des covariables dans le groupe traité et dans le groupe de contrôle. De même, [Imai et al. \(2008\)](#) suggèrent d'étudier aussi les moments d'ordre supérieur des covariables. Comparer les *ratios de variances* (moment d'ordre deux) permet ainsi par exemple d'affiner l'analyse des distributions ([Ho et al., 2007](#)). De même, l'étude des covariables croisées deux à deux permet de s'intéresser à la similarité des covariances des variables dans le groupe traité et non traité.

Comme nous l'avons dit, la spécification du modèle pour estimer le score de propension est un processus itératif qui s'appuie notamment sur l'obtention de distributions des covariables identiques dans le groupe traité et non traité. Ainsi, en pratique, si la distribution de variables (y compris leur corrélation) diffère singulièrement entre le groupe traité et non traité, il est souvent nécessaire de les introduire dans le modèle, si toutefois la taille de l'échantillon permet une telle estimation⁶.

1.2.2 Ajustement pour traiter du biais résiduel

La comparaison entre les deux groupes peut ensuite, en théorie, être effectuée par une simple régression linéaire de la variable étudiée sur l'indicatrice de traitement en utilisant l'échantillon apparié, pondéré ou partitionné. L'intégration de covariables peut toutefois être utile pour améliorer la précision de l'estimation, voire nécessaire pour essayer de réduire le biais de sélection résiduel engendré par des variables non équilibrées. [Rubin \(2001\)](#) préconise ainsi l'utilisation de covariables dans l'estimation de l'effet moyen du traitement dès lors que la valeur absolue de la différence standardisée des moyennes ou le ratio des variances *du score de propension* est supérieur, respectivement, à 0,25 et 2⁷.

Ces conditions supposent implicitement que les covariables suivent une distribution normale, ou au moins que leur distribution peut être résumée par leur espérance et leur variance. Dans le cas contraire, une méthode consiste à comparer graphiquement les distributions du score de propension dans les deux groupes.


5. [Rosenbaum et Rubin \(1985\)](#) ont ainsi montré que pour une covariable non linéairement corrélée à la quantité analysée, il subsistait un biais de sélection dans l'échantillon apparié, même si les moyennes sont identiques entre les deux groupes.

6. Certains auteurs (voir par exemple [Rosenbaum et Rubin, 1984](#); [Dehejia et Wahba, 2002](#); [Imbens et Rubin, 2015](#)) ont aussi proposé des algorithmes permettant de sélectionner les covariables et leurs interactions les plus pertinentes. Ceux-ci ne sont cependant pas implémentés sous [R](#) à ce jour.


7. Ces valeurs ne sont qu'indicatives. Elles s'appuient sur des simulations pour estimer le biais résultant de la non-linéarité de la relation entre le revenu et les covariables qui est problématique en présence de trop grandes différences en les covariables entre le groupe traité et le groupe de contrôle.

2 Appariement sur le score de propension

La méthode la plus intuitive - et donc fréquemment mise en œuvre - pour étudier l'effet d'un traitement sur une dimension donnée, consiste à comparer le résultat observé pour une unité traitée à celui d'une ou plusieurs unités non traitées mais « semblables » (Rubin, 1973). Précisément, il s'agit d'apparier chaque unité avec le(s) unité(s) de l'autre groupe présentant le(s) score(s) de propension le(s) plus proche(s)¹. Une fois l'appariement effectué, un estimateur de l'effet du traitement est obtenu en comparant les différences de résultats entre unités appariées.

Les packages **Matching** et **MatchIt** sous  sont les plus couramment utilisés pour effectuer un appariement sur score de propension avec le(s) plus proche(s) voisin(s). Dans ce chapitre, nous nous concentrerons cependant sur l'utilisation du package **Matching** suivant les recommandations d'Abadie et Imbens (2006).

2.1 Appariement sur le score de propension (*Nearest Neighbor Matching*)

Bien qu'intuitive, l'appariement sur le score de propension soulève plusieurs interrogations pratiques. À combien d'unités de l'autre groupe doit-on apparier chaque individu? Une même unité peut-elle être utilisée plusieurs fois? etc. La réponse à ces questions dépend de la taille de l'échantillon disponible, et peut impacter l'estimation de l'effet et sa précision. Il n'existe pas de règles guidant le chargé d'études, mais nous rappelons dans cette section les effets induits par le nombre de voisins, l'utilisation réitérée d'une même observation du groupe de contrôle avec plusieurs unités traitées ou l'appariement exact sur certaines covariables qualitatives couplées au score de propension. Ce choix s'explique en partie par la possibilité offerte sous  de préciser ces différents éléments.

2.1.1 Choisir le nombre de voisins : *Ratio matching*

Lorsqu'il existe un nombre important d'observations, il est parfois possible de pouvoir sélectionner plusieurs individus dont la « proximité » avec l'individu traité est similaire. Le nombre de « voisins » à retenir repose alors sur un compromis classique biais-variance. Augmenter le nombre de voisins permet de réduire la variance en accroissant la taille de l'échantillon apparié mais peut occasionner un biais dans l'estimation puisque le 2^e, 3^e etc. individus retenus peuvent de fait être de plus en plus éloignés de l'individu traité.

1. L'appariement peut chercher à minimiser la distance en valeur absolue pour chaque unité (*greedy* algorithme) ou une mesure de distance globale (voir l'algorithme d'*optimal matching* proposé par Rosenbaum, 2002). Rosenbaum justifie son utilisation lorsque le nombre d'unités du groupe de contrôle est élevé car il permet d'obtenir plus aisément une meilleure qualité de l'appariement au regard des diagnostics usuellement utilisés.

2.1.2 Appariement exact et/ou caliper

L'appariement par score de propension est parfois couplé à l'utilisation d'un appariement exact sur certaines caractéristiques qualitatives (voir par exemple [Dehejia et Wahba, 1999](#); [Rubin, 2001](#)) et/ou à l'utilisation d'une borne supérieure (*caliper*) à la distance entre les observations pour ne pas retenir de voisins trop éloignés. Un individu du groupe de contrôle est sélectionné si sa distance à l'individu traité est inférieure au *caliper*. Si cette méthode conduit à réduire le nombre d'appariement de « faible qualité », il faut noter qu'elle peut conduire à des difficultés d'interprétation si le nombre d'individus non appariés est important, puisque l'effet du traitement n'est identifié que sur la sous-population des unités appariées.

2.1.3 Avec ou sans remise

Une autre interrogation porte sur l'opportunité d'utiliser un même individu du groupe de contrôle pour différents individus du groupe traité. L'appariement avec remise permet d'améliorer la qualité de l'appariement si un individu du groupe de contrôle ressemble à plusieurs individus du groupe traité et peut être une nécessité si le nombre d'individus du groupe de contrôle est restreint ([Dehejia et Wahba, 1999](#)). De plus, l'ordre de traitement des unités n'est alors plus primordial, contrairement à l'appariement sans remise pour lequel le groupe des unités de contrôle potentielles pour chaque unité traitée se réduit à chaque appariement. Cependant le calcul de la précision dans une telle démarche est plus complexe puisque les observations utilisant un même individu du groupe de contrôle ne sont plus indépendantes.

2.2 Estimation après appariement

L'effet du traitement est usuellement défini pour un individu i par la différence entre le résultat qu'il aurait en ayant suivi le traitement (noté $Y_i(1)$) et celui qu'il aurait en l'absence de traitement (noté $Y_i(0)$)². Deux effets moyens du traitement peuvent alors être estimés. L'*effet moyen du traitement sur les traités* (*Average Treatment effect on the Treated* ou ATT) :

$$ATT = E[Y_i(1) - Y_i(0) | T_i = 1]$$

et l'*effet moyen du traitement* (*Average Treatment Effect* ou ATE) qui repose néanmoins sur des hypothèses plus fortes que l'ATT (voir [Givord, 2010](#)) :

$$ATE = E[Y_i(1) - Y_i(0)]$$

Dans le cas de l'appariement sur le score de propension, une fois constitué l'échantillon apparié d'unités traitées et de contrôle, et après avoir vérifié que les distributions des covariables dans l'échantillon apparié sont similaires, l'effet du traitement est estimé à partir de la différence moyenne des résultats entre l'unité considérée et celles auxquelles elle est appariée. Plusieurs estimateurs de cette différence peuvent être envisagés. Toutefois, ces estimateurs ne seront des estimateurs d'un effet *causal* que si l'hypothèse d'indépendance conditionnelle, notamment, est valide, c'est-à-dire que $Y_i(1), Y_i(0) \perp T | X$. Elle précise que conditionnellement aux *seules* caractéristiques observables, le processus de sélection est indépendant des revenus potentiels des personnes (cf. [Givord \(2010\)](#)).

2. Puisqu'une seule de ces deux situations est effectivement observée, ces résultats sont qualifiés de résultats potentiels.

2.2.1 Estimateur simple après appariement

Formellement, dans le cas d'un appariement avec les M unités les plus proches, l'estimateur simple de l'effet moyen du traitement sur les traités (ATT) pour les N_T observations traitées est donné par :

$$\hat{\tau}_M^{\text{ATT}} = \frac{1}{N_T} \sum_{i=1}^{N_T} \left(Y_i - \frac{1}{M} \sum_{j \in J_m(i)} Y_j \right)$$

où $J_m(i)$ correspond aux unités appariées à l'unité i . Ainsi, pour chaque unité appariée, un contrefactuel pour i est construit par la moyenne des valeurs de Y des unités qui lui sont appariées.

L'estimateur ATE correspondant s'appuie sur un appariement supplémentaire des unités de contrôle avec les unités traitées. Formellement, il est donc donné par :

$$\hat{\tau}_M^{\text{ATE}} = \frac{1}{N} \sum_{i=1}^N (\hat{Y}_i(1) - \hat{Y}_i(0))$$

où,

$$\hat{Y}_i(0) = \begin{cases} Y_i & \text{si } T_i = 0 \\ \frac{1}{M} \sum_{j \in J_m(i)} Y_j & \text{si } T_i = 1 \end{cases}$$

et

$$\hat{Y}_i(1) = \begin{cases} \frac{1}{M} \sum_{j \in J_m(i)} Y_j & \text{si } T_i = 0 \\ Y_i & \text{si } T_i = 1 \end{cases}$$

Dans le cas d'un appariement sur le *seul* score de propension (comme variable continue), [Abadie et Imbens \(2006\)](#) ont montré que ces estimateurs sont convergents et suivent une distribution asymptotique normale³. Par ailleurs, ils proposent un estimateur de leur variance, en soulignant que l'estimation par bootstrap d'un intervalle de confiance n'est pas valide.

2.2.2 Estimateur apparié corrigé du biais

En pratique, à distance finie, l'estimateur simple présente un biais dans le cas où l'appariement est effectué avec au moins une variable continue X , puisqu'il sera d'autant plus rare d'obtenir deux unités présentant une valeur strictement égale, que l'échantillon est réduit⁴. Ce biais peut être corrigé en ajustant la valeur du contrefactuel imputée à chaque unité i dans l'expression de l'estimateur simple⁵. Intuitivement, le biais est estimé par la *différence* d'estimations de Y pour l'unité i et chaque unité à laquelle elle est appariée, obtenues à partir d'une estimation convergente de $\mu(x) = E[Y | X = x]$, par exemple avec une régression linéaire.

Précisément, l'estimateur ATT s'écrit alors :

$$\hat{\tau}_{M,bcm}^{\text{ATT}} = \frac{1}{N_T} \sum_{i=1}^{N_T} \left(Y_i - \frac{1}{M} \sum_{j \in J_i} (Y_j + \hat{\mu}_0(X_i) - \hat{\mu}_0(X_j)) \right)$$

3. Ces propriétés s'entendent au sens où le nombre d'observations N augmente, sans que cela soit le cas du nombre de voisins retenus M , comme cela se traduit dans la pratique où l'appariement se fait souvent avec un nombre restreint d'unités proches.

4. Le biais diminue lorsque la taille de l'échantillon grandit mais à une « vitesse » d'autant plus faible que le nombre de covariables utilisées est élevé.

5. soit en ajustant $\frac{1}{M} \sum_{j \in J_m(i)} Y_j$

On notera qu'il ne s'agit pas d'une simple régression linéaire sur l'échantillon apparié et qu'une fonction de régression $\mu_0(x)$ est estimée sur le groupe non traité⁶.

De même, l'estimateur ATE correspond à :

$$\hat{\tau}_{M,bcm}^{ATE} = \frac{1}{N} \sum_{i=1}^N (\tilde{Y}_i(1) - \tilde{Y}_i(0))$$

où

$$\tilde{Y}_i(0) = \begin{cases} Y_i & \text{si } T_i = 0 \\ \frac{1}{M} \sum_{j \in J_i} (Y_j + \hat{\mu}_0(X_i) - \hat{\mu}_0(X_j)) & \text{si } T_i = 1 \end{cases}$$

et

$$\tilde{Y}_i(1) = \begin{cases} \frac{1}{M} \sum_{j \in J_i} (Y_j + \hat{\mu}_1(X_i) - \hat{\mu}_1(X_j)) & \text{si } T_i = 0 \\ Y_i & \text{si } T_i = 1 \end{cases}$$

où $\mu_1(x)$ et $\mu_0(x)$, sont estimées, respectivement sur le groupe traité et le groupe non traité.

Ces estimateurs, proposés par [Abadie et Imbens \(2011\)](#), sont appelés *estimateurs appariés corrigés du biais*. Ils sont, par ailleurs, convergents quel que soit le nombre de variables *continues* introduites lors de l'appariement, contrairement à l'estimateur simple. Les auteurs proposent, de plus, un estimateur convergent de leur variance et soulignent qu'ils suivent asymptotiquement une loi normale. Les estimateurs appariés corrigés du biais sont implémentés par le package **Matching** présenté dans la partie suivante⁷.

2.3 Mise en œuvre sous R

2.3.1 Appariement sur le score de propension

Une fois le score de propension estimé (cf. section 1.1.2), la fonction `Match` du package **Matching** permet d'effectuer un appariement avec ou sans remise, de préciser le nombre de « voisins », mais aussi de réaliser un appariement exact sur certaines variables et de calculer l'effet du traitement (« ATT » ou « ATE »). Elle attribue enfin une pondération aux unités qui interviennent plusieurs fois dans l'appariement.

Syntaxe complète de la fonction Match

```
Match(Y=NULL, Tr, X, Z = X, V = rep(1, length(Y)),
      estimand = "ATT", M = 1, BiasAdjust = FALSE,
      exact = NULL, caliper = NULL, replace=TRUE, ties=TRUE,
      CommonSupport=FALSE, Weight = 1, Weight.matrix = NULL, weights = NULL,
      Var.calc = 0, sample = FALSE, restrict=NULL, match.out = NULL,
      distance.tolerance = 1e-05, tolerance=sqrt(.Machine$double.eps),
      version="standard")
```

6. Par ailleurs, cette correction est nulle dans le cas d'un appariement exact, par exemple avec des covariables uniquement qualitatives (et un échantillon suffisamment large).

7. Les estimateurs de la variance proposés par [Abadie et Imbens \(2006, 2011\)](#) ne sont pas implémentés dans le package **MatchIt**. Il faut pour les calculer faire appel au package **MatchItSE** ([Henke, 2016](#)).

L'ordre d'appariement des unités par la fonction `Match` est aléatoire. Ceci est sans impact si l'option `replace=TRUE` est renseignée, car l'appariement est alors effectué avec remise. L'argument `M=k` renseigne le nombre d'unités appariées à chaque observation. Si plusieurs unités sont situées à la même distance d'une observation, l'option `ties = TRUE` (par défaut) apparie toutes les unités à l'observation et une pondération leur est appliquée pour refléter la multiplicité des appariements possibles (Abadie et Imbens, 2006). Si `ties = FALSE`, une unité est choisie aléatoirement. L'argument de `caliper` correspond à un coefficient multiplicatif de l'écart-type du score de propension. Ainsi, `caliper=c` désigne un caliper de $c\sigma_p$ où σ_p correspond à l'écart-type du score de propension. Usuellement, un caliper de 0.25 ou 0.5 est recommandé (Rubin et Thomas, 1996)⁸. Enfin, l'appariement sur score de propension peut être couplé avec un appariement exact sur certaines variables avec `exact=c("var1", "var2")`. Dans ce cas, il est recommandé que l'appariement exact ne soit pas demandé pour des variables continues⁹. Dans l'exemple ci-dessous, l'appariement sur le score de propension est couplé à un appariement exact sur la variable `race`. On notera que rien n'exclut en effet d'utiliser cette caractéristique qui intervient pourtant déjà dans l'estimation du score de propension.

Code : Appariement k:1 sur score de propension et matching exact

```
library(Matching)
Y <- mybase$re78      # variable de revenu
Tr <- mybase$treat    # variable de traitement


## 2:1 matching avec remise et appariement exact sur black et hisp

Match.out <- Match(Y = Y, Tr = Tr,
  estimand="ATT",
  M=2,                # nombre de voisins
  replace = TRUE,     # avec remise
  ties = TRUE,        # prise en compte
                    # des voisins similaires

  caliper = 0.15,     # caliper à 0.15 ecart-type
                    # du score de propension

  ## déclaration des variables utilisées pour
  ## l'appariement exact ou non
  X = cbind(mybase$pscore, mybase$race),
  exact = c("FALSE", "TRUE"))
```

2.3.2 Vérifier la propriété équilibrante après appariement

Sous , quelle que soit la méthode utilisée (appariement, stratification, ou ajustement par les pondérations), la fonction `bal.tab` du package `cobalt` peut être utilisée pour calculer les différences de moyennes standardisées et les ratios de variances des covariables afin de vérifier la propriété équilibrante du score de propension. Sa syntaxe est rappelée ci-dessous :

8. Si cette option n'est pas renseignée (`caliper=NULL`) alors aucun caliper n'est appliqué.

9. et donc notamment pour le score de propension

```
## Syntaxe complète de la fonction bal.tab
bal.tab(M, formula, data, weights = NULL, distance = NULL,
        subclass = NULL, method, int = FALSE,
        add1 = NULL, continuous = c("std", "raw"),
        binary = c("raw", "std"), s.d.denom, m.threshold = NULL,
        v.threshold = NULL, r.threshold = NULL, un = FALSE,
        disp.means = FALSE, disp.v.ratio = FALSE,
        disp.subclass = FALSE, cluster = NULL,
        which.cluster = NULL, cluster.summary = TRUE,
        estimand = NULL, quick = FALSE, ...)
```

La fonction `bal.tab` doit être appliquée à l'objet créé par la fonction `Match` (dans l'exemple `Match.out`) et renseigné dans l'argument `M`. L'argument `formula` précise l'indicatrice de traitement et les variables pour lesquelles on souhaite calculer les différents diagnostics. Son expression est sans lien informatique avec les covariables utilisées dans l'estimation du score de propension; il est ainsi possible d'introduire des variables supplémentaires¹⁰. À ce titre, l'argument `int=TRUE` permet d'effectuer les calculs pour *toutes* les interactions deux à deux des variables, sans avoir à saisir explicitement. Par ailleurs, on peut aussi ajouter la variable de score de propension¹¹ afin d'en calculer la différence de moyennes standardisées et le ratio de variances pour déterminer s'il convient d'ajuster d'un biais résiduel (cf. section 1.2.2). Enfin, si les différences de moyenne standardisées sont affichées par défaut¹², les arguments `disp.means=TRUE` et `disp.v.ratio=TRUE` permettent d'afficher aussi, respectivement, les moyennes dans chaque groupe et le ratio de variances. L'argument `un=TRUE` permet, lui, d'obtenir, pour comparaison, les calculs des différences standardisées et de ratios de variances *avant* l'appariement.

Nous détaillons à présent les calculs effectués par défaut. Tout d'abord, pour les variables qualitatives, c'est la différence de proportions qui est calculée par défaut (`binary="raw"`). Pour les variables continues, la différence de moyennes sera rapportée à la variance empirique du groupe traité, ou à la variance poolée (comme dans l'équation 1.1), en fonction de la valeur de l'argument `estimand` de la fonction `Match` qui aura été lancée auparavant. Par défaut, la fonction `Match` permet d'estimer l'effet moyen du traitement sur les unités traitées (`binary="ATT"`). La différence de moyennes sera alors rapportée à la variance de la variable au sein des seules unités traitées (qui correspond à l'argument `s.d.denom="treated"`). Pour utiliser un autre dénominateur, l'utilisateur doit donc le déclarer explicitement lors de l'appel de la fonction `bal.tab`. Ainsi, pour utiliser la variance poolée, il convient de renseigner `s.d.denom="pooled"`.

Si l'appariement est effectué avec remise (`replace=TRUE`) ou que l'on ne tire pas au hasard les unités du groupe de contrôle situées à la même distance d'une même unité traitée (`ties=TRUE`) comme dans l'exemple 2.3.1, la fonction `Match` affecte une pondération aux observations appariées (dont le nom par défaut est `weights`). Celle-ci est prise en compte, par défaut, dans les calculs et ne nécessite pas d'être précisée dans l'appel de la fonction `bal.tab`.

10. Une autre syntaxe est possible : elle utilise les arguments `treat` et `covs`, pour lesquels on précise respectivement le vecteur contenant l'indicatrice de traitement et la *dataframe* contenant les covariables.

11. c'est pourquoi nous proposons de stocker la variable `pscore` dans la base initiale `mybase`, une fois le score de propension estimé.

12. par la fonction `print.bal.tab`

Code R: Différences standardisées et ratios de variance avant/après appariement

```
balMatch <- bal.tab(M=Match.out,  
  formula = treat ~ re74 + u74 + re75 +  
    u75 + educ + race + married +  
    nodegree + age + pscore,  
  data = mybase,  
  un = TRUE,  
  disp.v.ratio = TRUE)
```

Balance Measures:

	Type	Diff.Un	V.Ratio.Un	Diff.Adj	V.Ratio.Adj
re74	Contin.	-0.7211	1.9300	0.1188	1.3776
u74	Binary	0.4470		-0.0592	
re75	Contin.	-0.2903	1.0457	0.1437	2.0880
u75	Binary	0.2876		-0.0103	
educ	Contin.	0.0550	2.0166	-0.0484	1.4490
race_black	Binary	0.6404		0.0000	
race_hispan	Binary	-0.0827		0.0000	
race_white	Binary	-0.5577		0.0000	
married	Binary	-0.3236		0.0919	
nodegree	Binary	0.1114		0.0417	
age	Contin.	-0.3094	2.2728	0.1519	1.0203
pscore	Contin.	1.9610	1.5637	0.0054	1.0238

Sample sizes:

	Control	Treated
All	429	185
Matched	173	173
Matched (Unweighted)	109	173
Unmatched	320	0
Discarded	0	12

Dans l'affichage des résultats (cf. le tableau Balance Measures), Diff.Un correspond à la différence standardisée de moyennes¹³ et V.ratio.Un au ratio de variances, *avant* appariement. Les résultats correspondants *après* appariement sont affichés dans les colonnes Diff.Adj et V.Ratio.Adj respectivement¹⁴. Dans notre exemple, la plupart des variables présentent, après appariement, de faibles différences standardisées de moyennes alors qu'elles étaient beaucoup plus élevées avant appariement; aucune n'excède 0.2. On notera par ailleurs que celle-ci est bien évidemment nulle pour les catégories de la variable race pour lesquelles un appariement exact a été effectué (race_black, race_hispan et race_white). Pour les variables continues, les ratios de variances avant (V.Ratio.Un) et après appariement (V.Ratio.Adj) sont aussi présentés. L'appariement sur le score de propension a permis d'égaliser la variance de la variable age dans le groupe traité et non traité apparié, et de réduire le ratio de variance pour les variables re74 et educ. En revanche, il s'est accru pour la variable re75.

13. ou la différence de proportions pour les variables qualitatives

14. Si ces calculs sont demandés, les colonnes M.C.Un et M.T.Un correspondent aux moyennes (proportions) des covariables, respectivement, dans le groupe de contrôle (C) et dans le groupe traité (T) *avant* tout traitement (Un pour *unadjusted*).

Cela illustre la difficulté à mener une comparaison parfaite, sur un échantillon de taille limitée; il est difficile d'adapter parfaitement toutes les distributions.

Comme le révèle le tableau `Sample sizes` - qu'il convient de toujours analyser - 12 unités traitées n'ont pas été appariées, à cause de l'appariement exact sur la variable `race` ou du caliper renseigné pour l'appariement sur le score de propension. La comparaison entre les deux groupes ignorera donc une partie des unités du groupe traité.

Lorsque le nombre de variables est important, une représentation graphique est souvent adoptée pour résumer les résultats. Elle est réalisée avec la fonction `love.plot` dont la syntaxe complète est rappelée ici.

```
## Syntaxe complète de la fonction love.plot

love.plot(x, stat = c("mean.diffs", "variance.ratios", "ks.statistics"),
          threshold = NULL, abs = FALSE, var.order = NULL,
          no.missing = TRUE, var.names = NULL,
          drop.distance = FALSE, agg.fun = c("mean",
          "median", "max", "range"), colors = NULL, shapes = NULL,
          line = FALSE, ...)
```

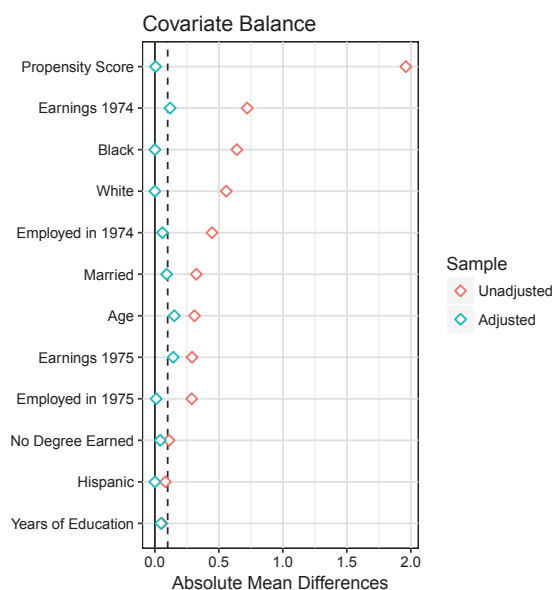
Cette fonction synthétise les différences de moyennes (standardisées), de proportions et les ratios de variances (pour les variables continues) *avant et après* appariement¹⁵. La fonction prend comme argument l'objet créé par `bal.tab` (`x = balMatch`), le type de statistiques (`stat="mean.diffs"` pour les différences de moyennes standardisées et `stat="variance.ratios"` pour les ratios de variance), éventuellement considérées en valeur absolue (`abs = TRUE`), et différentes options graphiques (`var.names` pour renommer les variables dans le graphique, `var.order` pour les ordonner selon l'ampleur de l'écart avant appariement, `colors` pour la couleur des résultats avant et après traitement, `threshold` pour représenter un seuil en deçà duquel on considérerait les variables comme équilibrées, etc.).

Le graphique 2.1 représente, ainsi, les résultats concernant les différences standardisées de moyennes *avant et après* appariement, stocké dans l'objet créé par la fonction `bal.tab` (`x = balMatch` dans l'exemple). Comme en témoignaient déjà les résultats fournis par la fonction `balMatch`, les différences standardisées après appariement sont plus faibles que sur l'échantillon initial.

Code : Différences standardisées avant / après appariement

```
## Graphique des différences de moyennes (standardisées) et de proportions
love.plot(x = balMatch,
          stat = "mean.diffs",
          abs = TRUE,
          ## options graphiques
          var.order = "unadjusted",
          var.names = new.names,
          threshold = 0.2,
          shape = 23)
```

15. ou après stratification ou ajustement par les pondérations, comme nous le verrons par la suite.



Graphique 2.1 – Différences standardisées de moyennes (et différences de proportions) avant et après appariement.

2.3.3 Représentations graphiques univariées et support commun

Au-delà, des différences de moyennes ou des ratios de variances, il est utile de *représenter graphiquement* la distribution des covariables dans le groupe traité et dans le groupe de contrôle, mais aussi de représenter le support commun¹⁶. Pour cela, il faut utiliser la fonction `bal.plot` du package **cobalt**.

Cette fonction est dédiée aux représentations graphiques univariées et s'appelle de manière similaire pour les variables continues ou qualitatives. Dans le cas continu, la fonction représente les estimations non paramétriques des densités de la variable dans le groupe traité et non traité. Pour les variables qualitatives, il s'agit d'histogrammes. Comme pour la fonction `bal.tab`, dans le cas d'un appariement avec remise ou prise en compte des jumeaux, les pondérations générées par la fonction `Match` sont prises en compte directement dans l'estimation des densités.

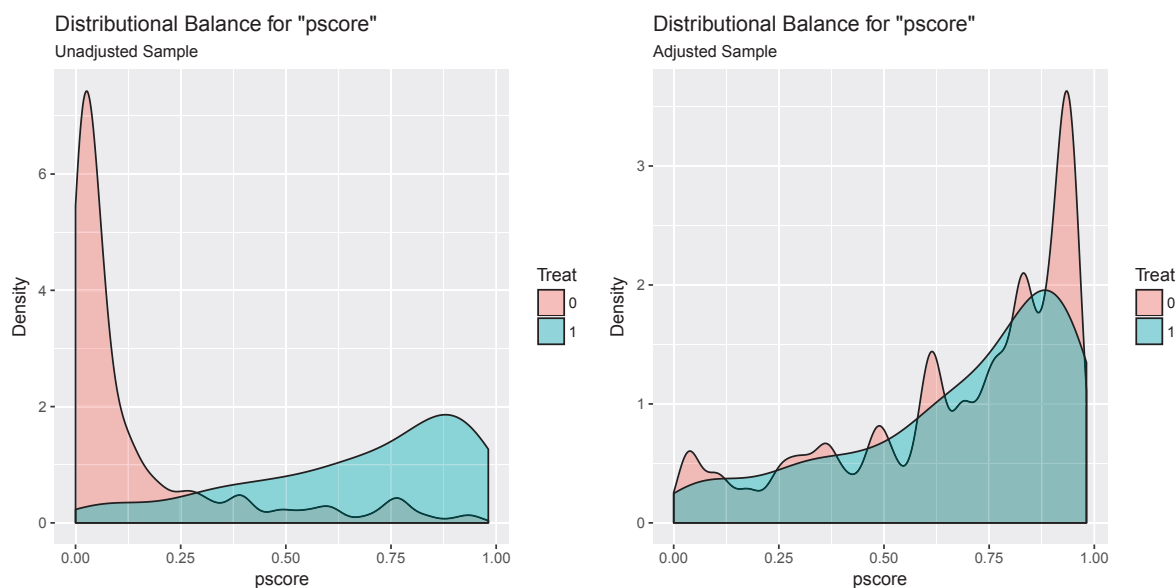
```
## Syntaxe complète de la fonction bal.plot

bal.plot(obj, var.name, ..., un = FALSE, which,
         which.sub = NULL, cluster = NULL, which.cluster = NULL,
         imp = NULL, which.imp = NULL, size.weight = FALSE)
```

La fonction `bal.plot` doit être exécuté *sur l'objet créé* par la fonction `Match` (ici `Match.out`), via l'argument `obj`. Pour comparer la distribution d'une variable *avant* et *après* appariement, il faut renseigner `which="both"`¹⁷ et son nom doit être indiqué dans les arguments `var.name` et `formula`. Enfin, on notera, dans le code présenté ci-dessous, que l'argument de la fonction `formula` reprend

16. c'est-à-dire comparer la distribution du score de propension dans le groupe traité et dans le groupe de contrôle.

17. Pour une simple comparaison après appariement, il faudra écrire `which="adjusted"`.



Graphique 2.2 – Support commun avant et après appariement.

les deux variables utilisées pour l'appariement `pscore` et `race`. Cela est nécessaire pour restreindre l'échantillon aux seules unités traitées *effectivement appariées*¹⁸.

Code : Densité du score de propension avant et après appariement

```
## Densité du score de propension avant et après appariement
## Les deux variables utilisées pour l'appariement sont déclarées dans formula
bal.plot(obj = Match.out,
         formula = treat ~ pscore + race,
         data = mybase,
         var.name = "pscore",
         which = "both")
```

Le Graphique 2.2 représente les distributions du score de propension estimé avant et après appariement (respectivement à gauche et à droite). Avant appariement, le support commun semble assez large, mais le graphique fait apparaître que des unités traitées ayant un fort score de propension peuvent ne pas être appariées à cause du faible nombre d'unités du groupe de contrôle proches. En effet, dans notre exemple, l'utilisation du caliper conduit ainsi à ne pas conserver toutes les unités traitées après l'appariement, comme nous l'avons d'ailleurs constaté précédemment lors de la présentation des résultats de la fonction `bal.tab`. Après appariement, la distribution du score de propension dans le groupe traité et non traité est similaire, ce qui reflète que les unités traitées ont effectivement été appariées avec des unités non traitées présentant un score de propension proche.

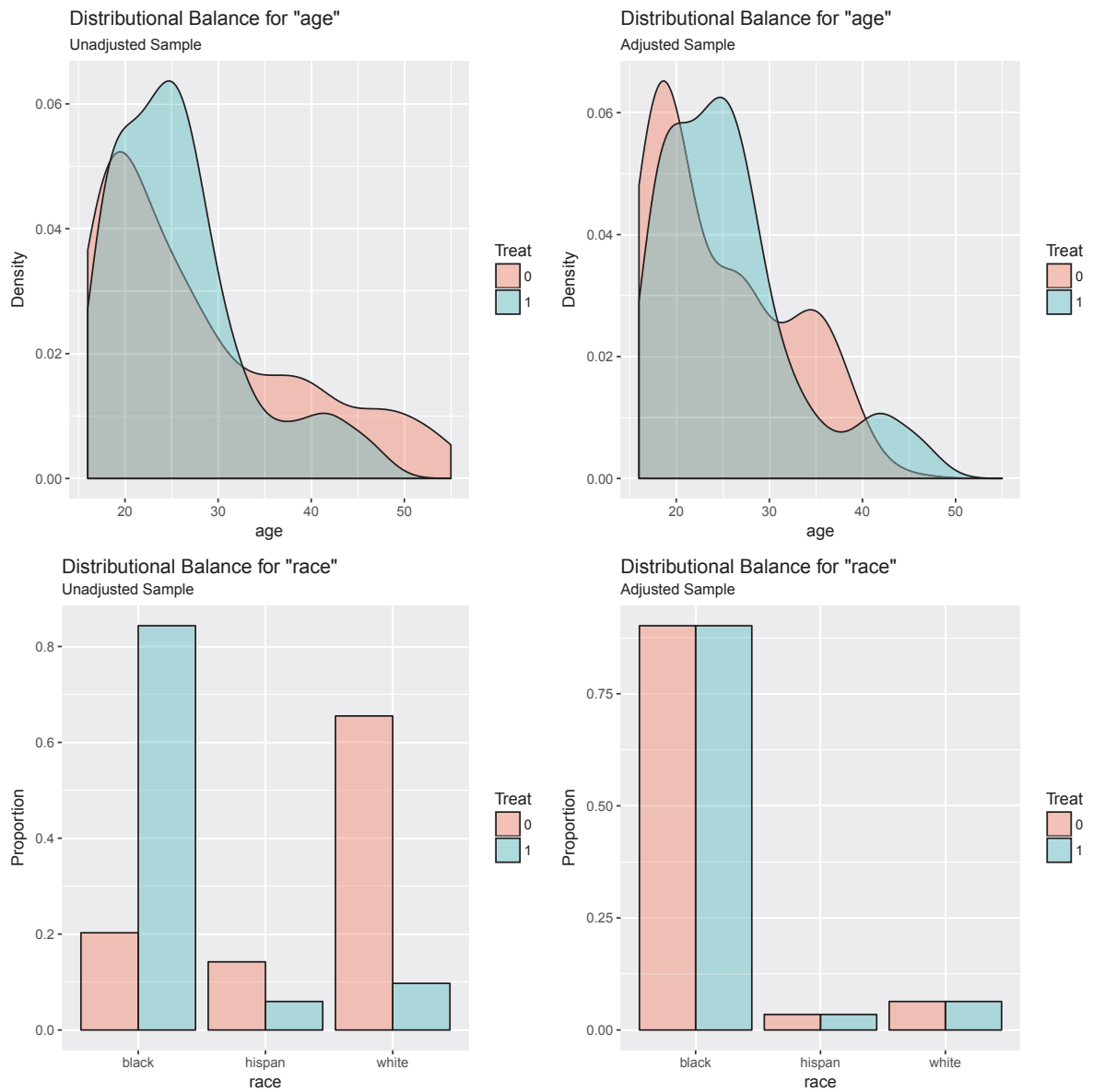
La représentation des distributions des covariables dans les deux groupes, avant et après appariement, complète les résultats obtenus sur les (différences standardisées de) moyennes et (ratios de) variances, qui ne sauraient à elles-seules résumer l'ensemble de la distribution. L'appariement sur le

18. Le cas échéant, par exemple `formula = treat ~ age`, occasionne un message d'erreur, puisque 12 unités traitées ne sont pas appariées dans notre exemple.

score de propension permet ainsi d'obtenir des distributions similaires pour l'âge dans l'échantillon apparié (cf. Graphique 2.3). Elles sont même identiques après appariement pour la variable race, puisque dans notre exemple, l'appariement sur cette caractéristique est exact.

Code : Densité de covariables avant et après appariement

```
## Densité de l'âge avant et après appariement
## La variable age doit aussi être intégrée dans l'option formula
bal.plot(obj = Match.out,
         formula = treat ~ age + pscore + race,
         data = mybase,
         var.name = "pscore",
         which = "both")
```



Graphique 2.3 – Densité de l'âge et de la proportion de personnes mariées.

2.3.4 Estimation de l'effet du traitement

La fonction `Match` calcule l'estimation de l'effet moyen du traitement sur les traités (ATT) mais aussi sur la population totale (ATE). Par défaut, l'estimateur calculé n'est pas corrigé du biais et correspond donc à une simple différence de moyennes sur l'échantillon apparié (cf. section 2.2.1). Les résultats obtenus sont fournis par la fonction `summary.Match()`. L'option `full=TRUE` permet d'obtenir l'estimateur classique de la variance et celui proposé par [Abadie et Imbens \(2006\)](#) pour tenir compte (asymptotiquement) de la variabilité induite par la procédure d'appariement elle-même¹⁹.

Code : Estimateur simple après appariement

```
## 2:1 matching avec remise et appariement exact sur race
Match.out <- Match(Y = Y, Tr = Tr,
                  X = cbind(mybase$pscore,mybase$race),
                  estimand = "ATT",
                  exact = c("FALSE","TRUE"),
                  M=2,
                  caliper = 0.15,
                  sample=TRUE)
```

```
summary(Match.out, full=TRUE)
```

```
Estimate... 1128.2
AI SE..... 1268.3
T-stat..... 0.88956
p.val..... 0.3737

Est noAdj.. 1128.2
SE..... 726.42
T-stat..... 1.5532
p.val..... 0.12038

Original number of observations..... 614
Original number of treated obs..... 185
Matched number of observations..... 173
Matched number of observations (unweighted). 374

Number of obs dropped by 'exact' or 'caliper' 12
```

Dans notre exemple, l'effet moyen de la formation (traitement) sur les revenus en 1978 des personnes bénéficiaires du programme de formation est estimé à 1128,2 dollars. Comme nous n'avons pas demandé d'ajustement du biais (`BiasAdjust=FALSE`), l'estimateur calculé (`Estimate`) est égal à celui non ajusté (`Est noAdj`). L'estimateur de la variance de l'estimateur est `AI SE`. Il est plus élevé que l'estimateur classique (`SE`) de la variance de l'écart entre deux moyennes qui ne prend pas en

19. On notera, toutefois, que les estimateurs proposés ne corrigent pas de la variabilité induite par l'estimation du score de propension.

compte la variabilité induite par le choix des voisins. L'effet estimé n'est cependant pas significatif quel que soit l'estimateur de l'écart-type retenu.

L'utilisation de variables pour calculer l'estimateur apparié corrigé du biais sur l'échantillon apparié (cf. section 2.2.1) doit être déclarée lors de l'appel de la fonction `Match` avec l'option `BiasAdjust=TRUE`. Par défaut, les covariables retenues sont les mêmes que celles utilisées par l'appariement. Elles peuvent néanmoins différer via l'argument `Z`. Lors de la présentation des résultats, l'estimation de l'effet moyen obtenue avec correction du biais est appelée `Estimate`, celle de l'estimateur simple après appariement `Est noAdj`.

Code : Estimateur apparié corrigé du biais

```
## 2:1 matching avec remise et appariement exact sur race
Match.out <- Match(Y = Y, Tr = Tr,
  X = cbind(mybase$pscore,mybase$race),
  exact = c("FALSE","TRUE"),
  M=2,
  caliper = 0.15,
  ## Correction du biais résiduel
  BiasAdjust = TRUE,
  ## Covariables utilisées pour corriger du biais
  Z = cbind(mybase$re74,mybase$re75,mybase$age),
  sample=TRUE)
```

```
summary(Match.out, full=TRUE)
```

```
Estimate... 1070.7
AI SE..... 1250.7
T-stat..... 0.85611
p.val..... 0.39194

Est noAdj.. 1128.2
SE..... 726.42
T-stat..... 1.5532
p.val..... 0.12038

Original number of observations..... 614
Original number of treated obs..... 185
Matched number of observations..... 173
Matched number of observations (unweighted). 374

Number of obs dropped by 'exact' or 'caliper' 12
```

Dans notre exemple, nous introduisons comme covariables pour corriger du biais, les revenus en 1974 (`re74`), en 1975 (`re75`) et l'âge (`age`). L'effet moyen de la formation sur les revenus des bénéficiaires estimé serait alors de 1070,7 dollars. Les deux estimateurs, corrigé et non corrigé du biais sont

donc proches. Cela n'est pas surprenant puisque les distributions des covariables dans le groupe traité et parmi les non-traités présentaient peu de différences une fois l'échantillon apparié.

3 Stratification

L'appariement en sélectionnant le(s) plus proche(s) voisin(s) conduit souvent, de fait, à ne pas utiliser l'information disponible pour plusieurs individus du groupe de contrôle quand bien même, par exemple, ceux-ci présenteraient un score de propension situé dans l'intervalle des valeurs du score de propension du groupe traité. La méthode de stratification utilise au contraire l'ensemble des observations disponibles en partitionnant les individus traités et du groupe de contrôle suivant les valeurs estimées du score de propension. Au sein de chaque strate, les individus ont des valeurs similaires de score de propension, et ce faisant des distributions de covariables proches par la propriété équilibrante de celui-ci. L'estimation de l'effet moyen total agrège les estimations préalables dans chaque strate (voir partie 3.1). Celles-ci sont réalisées par régression linéaire et ne nécessitent pas de package particulier (cf. partie 3.2).

3.1 L'estimation par stratification

Rosenbaum et Rubin (1983) détaillent l'estimation de l'effet du traitement qui suit une partition de l'échantillon en M strates. L'estimateur de l'effet moyen total du traitement sur la population correspond à la somme pondérée des effets moyens $\hat{\tau}_m$, au sein de chaque strate, les pondérations appliquées différant selon que l'on estime l'effet moyen du traitement sur la population ou sur les seules unités traitées.

Ainsi, l'effet moyen du traitement sur les traités correspond à :

$$\hat{\tau}_{Block}^{ATT} = \sum_{m=1}^M \hat{\tau}_m \frac{N_{1m}}{N_T}$$

où N_T désigne le nombre total d'unités traitées et N_{1m} , le nombre d'unités traitées de la strate m .

De même, un estimateur de l'effet moyen du traitement sur la population est donné par :

$$\hat{\tau}_{Block}^{ATE} = \sum_{m=1}^M \hat{\tau}_m \frac{N_{1m} + N_{0m}}{N}$$

où N_{0m} désigne le nombre d'unités non traitées de la strate m et N le nombre total d'unités.

L'estimation de l'effet du traitement pour chaque strate peut être obtenue par une simple régression linéaire *au sein des unités qui la composent*, soit :

$$Y_i = \alpha_m + \tau_m T_i + \epsilon_i$$

En pratique, il est souhaitable d'ajuster (si le nombre d'observations le permet) ces régressions linéaires en introduisant les covariables déjà présentes dans le score de propension afin de réduire

l'éventuel biais résiduel mais aussi d'améliorer la précision des estimations (Lunceford et Davidian, 2004; Imbens et Rubin, 2015).

$$Y_i = \alpha_m + \beta'_m X_i + \tau_m T_i + \epsilon_i \quad (3.1)$$

3.1.1 Constitution des strates

La validité de la méthode repose sur le respect de la propriété équilibrante au sein de chaque strate. La détermination du nombre de strates et de leurs intervalles respectifs doit donc être guidée par cet objectif. Par ailleurs, l'estimation par régression linéaire nécessite aussi que le nombre d'unités au sein de chaque strate soit suffisamment élevé. Plusieurs approches peuvent être envisagées.

En pratique, les strates sont souvent définies en fonction des quantiles du score de propension pour estimer l'effet moyen sur la population, ou ceux du score de propension *du seul groupe traité* pour estimer l'effet moyen sur les traités (voir par exemple Rosenbaum et Rubin, 1985), ce qui conduit à construire cinq classes de taille identique. Cette méthode sera celle détaillée dans la mise en œuvre pratique.

Mais, il est aussi possible de coupler la définition des strates avec un appariement avec le(s) plus proche(s) voisin(s). Par exemple, chaque individu traité peut être apparié avec un ou plusieurs individus du groupe de contrôle de la même classe. Dans ce cas, tous les individus du groupe de contrôle ne sont pas conservés¹. Enfin, les méthodes de *Full Matching*, elles, sont une extension de la méthode de stratification dans laquelle le nombre de classes est déterminé empiriquement (Rosenbaum, 1991, 2002; Hansen, 2004; Imbens et Rubin, 2015)².

3.1.2 Précision de l'estimation


L'estimateur de l'effet moyen du traitement est une somme pondérée des effets moyens au sein de chaque strate. La variance correspondante est aussi estimée à partir des variances des estimateurs au sein de chaque strate. Ainsi, la variance de l'estimateur de l'effet moyen sur les traités correspond à

$$\hat{V}(\hat{\tau}_{Block}^{ATT}) = \sum_{m=1}^M \hat{V}(\hat{\tau}_m) \left(\frac{N_{1m}}{N_1} \right)^2$$

et celle de l'estimateur de l'effet moyen du traitement sur la population à

$$\hat{V}(\hat{\tau}_{Block}^{ATE}) = \sum_{m=1}^M \hat{V}(\hat{\tau}_m) \left(\frac{N_{1m} + N_{0m}}{N} \right)^2$$

en utilisant les mêmes notations que précédemment.

1. La mise en œuvre de cette méthode peut être réalisée avec le package  **MatchIt** (Ho et al., 2011). Une fois la constitution des strates réalisée, l'estimation de l'effet peut cependant être menée telle que détaillée dans ce document.

2. Un algorithme constitue le nombre de classes nécessaires en cherchant à réduire les différences, au sein de chaque strate, en moyenne, entre les individus traités et les individus du groupe de contrôle pour l'ensemble des covariables, tout en conservant un nombre minimal d'individus. Contrairement à la méthode de stratification usuelle, la taille des classes n'est alors pas nécessairement similaire. Là encore, un exemple de cette approche peut être implémentée avec le package **MatchIt** (voir Sävje et al., 2017, pour une description des différentes techniques de *Full Matching* et des différents packages permettant leur implémentation)

3.2 Mise en œuvre sous R

L'estimation par stratification nécessite une estimation préalable du score de propension et la détermination des strates (cf. section 3.2.1). Pour cela, une attention particulière doit être portée à la vérification des propriétés équilibrantes du score de propension (cf. section 3.2.2) et du nombre d'unités présentes dans chaque strate. Celles-ci enfin constituées, l'estimation de l'effet moyen du traitement au sein de chaque strate est obtenue par régression linéaire (en incluant si nécessaire des covariables pour ajuster du biais résiduel) avant de calculer l'estimateur agrégé avec sa variance (cf. section 3.2.3).

3.2.1 Estimation du score de propension et construction des strates

Une fois le score de propension estimé (cf. section 1.1.2) en faisant attention à disposer d'un support commun suffisamment large pour permettre ensuite une analyse par strate avec un nombre suffisant d'unités, il faut constituer les différentes strates. Est détaillée, ici, la syntaxe nécessaire à la construction de strates à partir des quartiles de la distribution du score de propension *des unités traitées*, à des fins d'estimation de l'ATT. Pour cela, nous utilisons la fonction `findInterval`, qui permet de partitionner une variable continue, associée à la fonction `quantile` pour calculer les quartiles. La variable `subclass` créée contient le numéro de la classe de chaque unité, numérotée de 1 à 4 (et non de 0 à 3, grâce à l'argument `all.inside=TRUE`).

Code R: Constitution des strates à partir des quartiles

```
mybase$subclass <- findInterval(  
  mybase$pscore,  
  ## Calcul des quartiles du score de propension sur  
  ## les seules unités traitées  
  quantile(mybase$pscore[mybase$treat==1], (0:4)/4),  
  all.inside=TRUE)
```

3.2.2 Vérifier la propriété équilibrante au sein de chaque strate

La constitution des strates est, en fait, un processus itératif, car il faut vérifier la propriété équilibrante du score de propension au sein de chaque strate, mais il faut aussi que chacune dispose d'un nombre suffisant d'unités. Pour cela, on utilise la fonction `bal.tab`, déjà présentée dans la section 2.3.2 en introduisant les arguments `method="subclassification"`, `subclass` qui précise le nom de la variable contenant le numéro de classe et `disp.subclass=TRUE` pour afficher les calculs par strates³.

Code R: Différences standardisées et ratio de variances

```
balSubclass <- bal.tab(formula = treat ~ age + race +  
  married + nodegree + re74 + u74 + re75 + u75,  
  data = mybase,
```

3. Par défaut, les différences de moyennes sont rapportées à la variance dans le groupe traité (`s.d.denom="treated"`) et le jeu de pondérations appliquées pour fournir les résultats agrégés (`Balance measures across subclasses`) correspond au poids retenu pour estimer l'effet moyen sur les traités (`estimand="ATT"`, cf. section 3.1).

3. STRATIFICATION

```
estimand = "ATT",
distance = "pscore",
disp.subclass = TRUE,
subclass = "subclass",
method = "subclassification",
un = TRUE)
```

Balance by subclass:

```
- - - Subclass 1 - - -
      Type Diff.Adj
pscore  Distance  1.2720
age      Contin.  -0.2138
race_black  Binary  0.3763
race_hispan Binary  0.0193
race_white  Binary -0.3956
married     Binary -0.1794
nodegree    Binary  0.0236
re74        Contin. -0.5047
u74         Binary  0.2503
re75        Contin. -0.2622
u75         Binary  0.1730

- - - Subclass 2 - - -
      Type Diff.Adj
pscore  Distance  0.1765
age      Contin.  -0.0776
race_black  Binary -0.0178
race_hispan Binary -0.0020
race_white  Binary  0.0198
married     Binary -0.0514
nodegree    Binary -0.1008
re74        Contin.  0.1962
u74         Binary -0.0929
re75        Contin. -0.0653
u75         Binary -0.0198

- - - Subclass 3 - - -
      Type Diff.Adj
pscore  Distance  0.7760
age      Contin.  0.4017
race_black  Binary -0.0217
race_hispan Binary  0.0217
race_white  Binary  0.0000
married     Binary  0.0609
nodegree    Binary  0.1174
re74        Contin.  0.1050
```

```

u74      Binary  -0.0855
re75     Contin.  0.2367
u75      Binary  -0.1029

```

- - - Subclass 4 - - -

```

      Type Diff.Adj
pscore  Distance -0.0167
age      Contin. -0.8314
race_black  Binary  0.0000
race_hispan Binary  0.0000
race_white Binary  0.0000
married  Binary  0.0000
nodegree Binary  0.2872
re74     Contin.  0.1459
u74      Binary  -0.0213
re75     Contin.  0.1013
u75      Binary  0.1223

```

Balance measures across subclasses:

```

      Type Diff.Un M.O.Adj Diff.Adj
pscore  Distance  1.9610  0.6066  0.5489
age      Contin. -0.3094 26.6079 -0.1838
race_black  Binary  0.6404  0.7595  0.0837
race_hispan Binary -0.0827  0.0498  0.0097
race_white Binary -0.5577  0.1907 -0.0934
married  Binary -0.3236  0.2314 -0.0422
nodegree Binary  0.1114  0.6251  0.0830
re74     Contin. -0.7211  2.4285 -0.0135
u74      Binary  0.4470  0.6956  0.0125
re75     Contin. -0.2903  1.4900  0.0032
u75      Binary  0.2876  0.5564  0.0436

```

Sample sizes:

```

      Subclass 1 Subclass 2 Subclass 3 Subclass 4
Control      388         22         15         4
Treated       46         46         46        47
Total        434         68         61        51

```

La présentation des résultats est similaire à celle discutée à la section 2.3.2. Leur pertinence dépend cependant du nombre d'observations présents dans chaque classe. Par ailleurs, la fonction fournit en plus les différences standardisées de moyennes agrégées (c'est-à-dire obtenues sur l'ensemble de l'échantillon stratifié, en pondérant les résultats obtenus par classe - cf. `Balance measures across subclasses`) et une répartition du nombre d'unités traitées et du groupe de contrôle au sein de chaque classe (cf. `Sample sizes`).

Les résultats obtenus révèlent, tout d'abord, que la 4^e classe ne contient que 4 unités du groupe de contrôle : cela résulte des problèmes de support commun que nous avons déjà évoqués à la section 2.3.3. Par contre, le nombre d'unités traitées est presque identique dans chaque classe, puisqu'elles

ont été construites sur les quartiles du score de propension estimé pour les unités traitées.

Des différences importantes subsistent dans les classes 1 et 3 sur certaines covariables. Au niveau agrégé, les différences de moyennes standardisées restent élevées pour les différentes catégories de la variable race et pour le revenu en 1974. Il en est aussi de même pour le score de propension (ce que suggèrerait déjà l'inégale répartition des unités du groupe de contrôle dans les différentes classes), reflet entre autre du faible support commun.

Cette approche peut être complétée par des représentations graphiques de la densité du score de propension et d'autres covariables dans chaque strate.

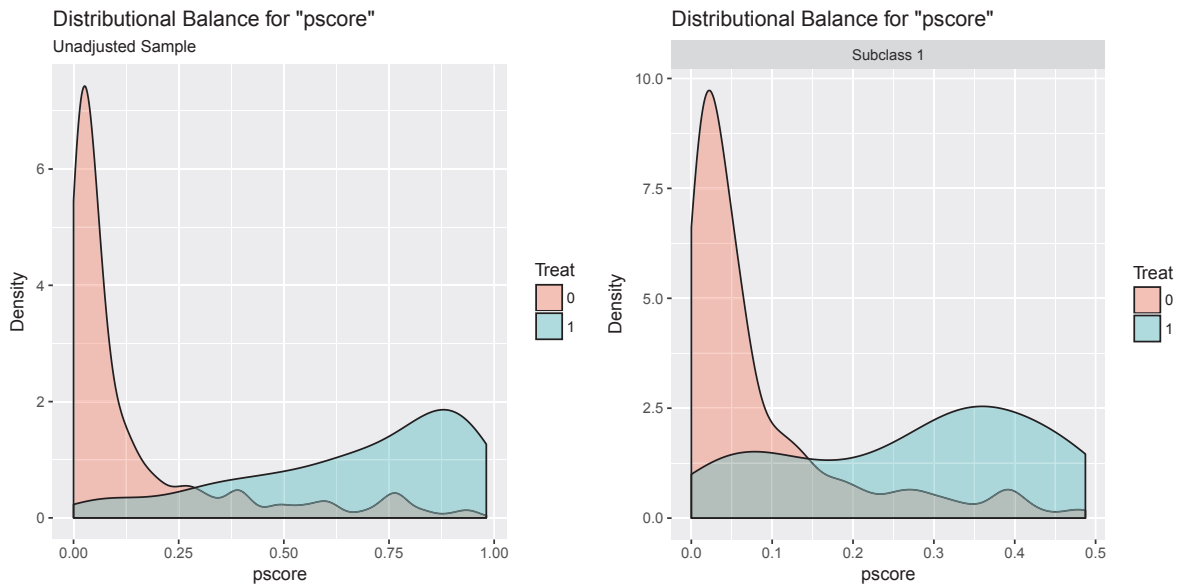
Code  : Densités du score de propension et pourcentage d'Afro-américains dans la première strate

```
bal.plot(obj = treat ~ pscore, data = mybase,
         var.name = "pscore",
         subclass = "subclass", which.sub = 1,
         which="both")

bal.plot(obj = treat ~ race_black, data = mybase,
         var.name="race_black",
         subclass = "subclass", which.sub = 1,
         which="both" )
```

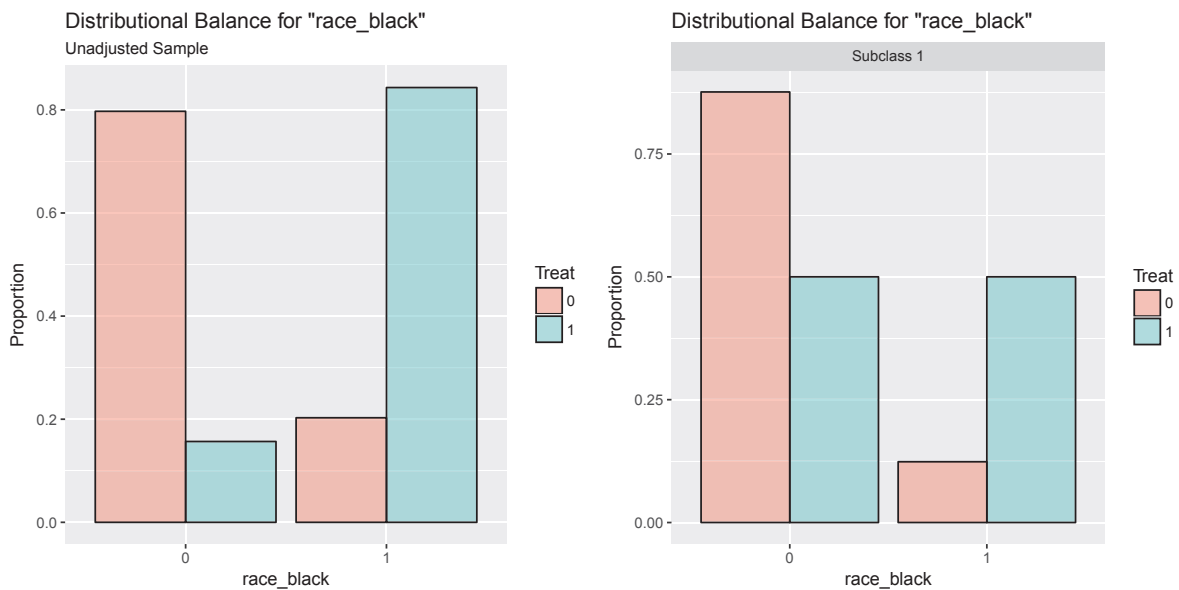
Pour cela, la fonction `bal.plot` doit être appelée avec les options `subclass` et `which.sub`, les autres arguments étant identiques à ceux présentés à la section 2.3.2. On notera que l'argument `which="both"`⁴ conduit à représenter simultanément la densité sur *l'ensemble de l'échantillon* avant stratification à côté de celle dans la *strate demandée* (cf. Graphiques 3.1 et 3.2). Encore une fois, la pertinence de ces représentations dépend toutefois du nombre d'observations dans chaque classe.

4. cf. section 2.3.3



Graphique 3.1 – Densité du score de propension avant stratification et dans la 1^{re} strate.

Le Graphique 3.1 du score de propension avant stratification (à gauche - *Unadjusted sample*) permet de comprendre que, dans la première strate, le support commun entre unités traitées et non traitées est faible (à droite - *Subclass 1*). Dès lors, il n'est pas surprenant de constater qu'il subsiste des différences importantes en les covariables entre les unités traitées et non traitées présentes dans cette strate, comme par exemple en ce qui concerne la proportion d'Afro-américains (cf. Graphiques 3.2).



Graphique 3.2 – Pourcentage d'Afro-américains avant stratification et dans la 1^{re} strate.

3.2.3 Estimation de l'effet du traitement

L'estimation de l'effet moyen du traitement s'appuie sur une analyse au sein de chaque strate, puis du calcul de l'effet moyen agrégé et de sa variance. En pratique sous **R**, cela consiste à estimer l'effet du traitement pour chaque strate (en corrigeant éventuellement du biais résiduel pour certaines covariables) à l'aide, par exemple, de la fonction `sapply`, avant de calculer l'effet moyen global du traitement (`overall.ATT` et son écart-type (`overall.ATTse`)).

Code **R**: Estimations par strate

```
## Pour chaque classe on effectue une régression linéaire
## et on stocke les estimations obtenues

subclass.results <- t(
  sapply(X = split(mybase,mybase$subclass),
        FUN = function(sub.base)
          summary(lm(formula = re78~ treat +
                    race_black + race_hispan +
                    age + married + nodegree +
                    re74 + u74 + re75 + u75,
                    data = sub.base)
                )$coefficients["treat",c(1,2,4)]))

## Calcul de la pondération appliquée à l'effet estimé
## pour chaque strate
weights <- sapply(
  X = split(mybase[which(mybase$treat==1),],
            mybase[which(mybase$treat==1),]$subclass),
  FUN = function(sub.base)
    nrow(sub.base)/nrow(mybase[which(mybase$treat==1),]))

## Concaténation des résultats
subclass.results <- cbind(subclass.results,weights)
```

```
subclass.results

  Estimate Std. Error Pr(>|t|) weights
1   605.6831   1077.063 0.5741776 0.2486486
2  1623.0735   1711.523 0.3469695 0.2486486
3   143.6825   1969.603 0.9421313 0.2486486
4 10652.4859   5577.335 0.0626705 0.2540541
```

Les effets moyens estimés par strate sont très différents. Dans la strate 1, par exemple, l'effet est faible (606 dollars) et non significatif, mais dans cette classe, qui contient 434 observations, les deux groupes traités et non traités présentent des distributions de covariables encore très différentes (cf. section 3.2.2). Malgré leur prise en compte lors de la régression linéaire, l'effet du traitement estimé reste sujet à caution. Dans la classe 4, l'effet moyen obtenu est 10 fois plus élevé que celui estimé par

la méthode d'appariement sur le score de propension (cf. section 2.3.4) et est significatif. Toutefois, comme nous l'avons déjà souligné, cette classe ne comporte que 51 observations avec seulement 4 unités non traitées (à cause du faible support commun) ce qui fragilise l'estimation obtenue.

Code R: Estimation de l'effet moyen agrégé

```
## Calcul de l'estimation de l'effet moyen sur les traités agrégé
overall.ATT <- subclass.results[,"weights"]%*%subclass.results[,"Estimate"]
## et de l'écart-type correspondant
overall.ATTse <- sqrt(subclass.results[,"weights"]^2%*%subclass.results[,"Std. Error"]^2)
## Intervalle de confiance à 95 %
ICinf <- overall.ATT - 1.96 * overall.ATTse
ICsup <- overall.ATT + 1.96 * overall.ATTse
```

```
data.frame(overall.ATT,overall.ATTse, ICinf, ICsup)
```

```
  overall.ATT overall.ATTse    ICinf    ICsup
1    3296.211    1581.267 196.9269 6395.495
```

L'effet moyen agrégé sur les unités traitées estimé correspond à une moyenne pondérée des effets estimés au sein de chaque strate. Puisque les strates sont construites à partir des quartiles du score de propension des unités traitées, l'effet estimé pour chaque strate est pondéré de manière identique (cf. section 3.1). Au total, on remarquera que l'estimateur est ici beaucoup plus élevé, mais également avec une plus grande dispersion. Cette forte imprécision est probablement liée au fait que l'on dispose d'un faible support commun : dans plusieurs strates, les individus des deux groupes comparables sont peu nombreux, ce qui se traduit par une grande imprécision. La différence constatée avec les résultats obtenus par des méthodes alternatives (comme après l'appariement sur le score de propension au chapitre précédent) illustre la nécessité d'être vigilant sur les détails techniques de l'implémentation.

4 Ajustement par pondération

L'ajustement par pondération, comme l'approche par stratification, permet d'utiliser toutes les unités de la base de données contrairement à l'appariement sur score de propension. La démarche consiste à pondérer les unités traitées par l'inverse du score de propension $p(X)$, et les unités de contrôle par $(1 - p(X))$ pour comparer les données ainsi pondérées afin d'estimer l'effet du traitement. En pratique, la démarche est donc similaire à celles explicitées dans les autres chapitres mais sur données pondérées. La première partie rappelle les différents estimateurs pondérés couramment utilisés, ainsi que les poids correspondant (cf. section 4.1). La deuxième partie est consacrée à la mise en œuvre sous \mathbb{R} de l'ajustement par pondération, en insistant notamment sur le calcul par *bootstrap* d'un intervalle de confiance de l'estimation obtenue (cf. section 4.2).

4.1 Estimateurs pondérés par l'inverse de la probabilité de traitement

4.1.1 Estimateur pondéré simple et doublement robuste

Dans la méthode d'ajustement par pondérations, inspirée des travaux d'[Horovitz et Thompson \(1952\)](#), le score de propension est utilisé comme pondération pour éliminer le biais engendré par les différences de caractéristiques observables. Précisément, il s'agit de pondérer chaque unité par l'inverse de leur probabilité d'assignation au groupe auquel elles appartiennent, c'est-à-dire par l'inverse du score de propension, $p(X)$ pour les unités traitées et $(1 - p(X))$ pour les unités du groupe de contrôle.

Cette méthode se justifie par deux propriétés découlant de l'hypothèse d'indépendance conditionnelle aux observables (cf. section 2.2) des résultats potentiels¹ :

$$E\left[\frac{Y_i T_i}{p(X_i)}\right] = E[Y_{1i}] \text{ et } E\left[\frac{Y_i(1 - T_i)}{(1 - p(X_i))}\right] = E[Y_{0i}]$$

Dès lors, l'effet moyen du traitement sur les traités peut s'écrire :

$$ATT = E[Y_i(1) - Y_i(0) | T_i = 1] = E\left[\frac{p(X_i)}{P[T_i = 1]} \left(\frac{T_i Y_i}{p(X_i)} - \frac{(1 - T_i) Y_i}{(1 - p(X_i))} \right)\right]$$

et l'effet moyen du traitement :

$$ATE = E[Y_i(1) - Y_i(0)] = E\left[\frac{T_i Y_i}{p(X_i)} - \frac{(1 - T_i) Y_i}{(1 - p(X_i))}\right]$$

1. Il suffit de remarquer que $E\left[\frac{T_i Y_i}{p(X_i)}\right] = E\left[\frac{T_i Y_i(1)}{p(X_i)}\right] = E\left[E\left[\frac{T_i Y_i(1)}{p(X_i)} | X_i\right]\right] = E\left[\frac{1}{p(X_i)} E[T_i | X_i] \cdot E[Y_i(1) | X_i]\right] = E[Y_i(1)]$, l'avant dernière égalité découlant de l'hypothèse d'indépendance conditionnelle.

Dans le cas de l'effet moyen du traitement sur toute la population, un estimateur « naturel » est donné par :

$$\hat{\tau}_{IPTW}^{ATE} = \frac{1}{N} \sum_{i=1}^N \frac{T_i Y_i}{\hat{p}(X_i)} - \frac{(1 - T_i) Y_i}{(1 - \hat{p}(X_i))}$$

où $\hat{p}(X_i)$ correspond à un estimateur *convergent* de $p(X_i)$ ².

Les pondérations appliquées à chaque unité correspondent ainsi à :

$$w_i^{ATE} = \frac{T_i}{\hat{p}(X_i)} + \frac{1 - T_i}{(1 - \hat{p}(X_i))}$$

Dans le cas, de l'effet moyen du traitement sur les seules unités traitées, ces pondérations sont multipliées par le score de propension $p(X)$. Ainsi,

$$w_i^{ATT} = \hat{p}(X_i) w_i^{ATE} = 1 + \frac{(1 - T_i) \hat{p}(X_i)}{(1 - \hat{p}(X_i))}$$

Il est intéressant de noter que ces estimateurs peuvent être facilement calculés en estimant directement la régression pondérée³ :

$$Y_i = \alpha + \tau T_i + \epsilon_i$$


avec les poids w_i^{ATT} ou w_i^{ATE} respectivement pour estimer l'effet du traitement sur les seules unités traitées ou sur toute la population.

Cette approche permet par ailleurs d'inclure d'autres covariables à cette fonction de régression. L'estimateur obtenu qui combine pondération et régression (suggéré par [Robins et al., 1995](#)) est appelé estimateur *doublement robuste*. En effet, l'estimateur $\hat{\tau}_{IPTW}$ de l'équation (4.1.1) est convergent *si* le modèle estimant le score de propension est correctement spécifié. L'estimateur doublement robuste est lui convergent si la spécification du score de propension *ou* de la fonction de régression (liant le revenu aux covariables) est correcte ([Scharfstein et al., 1999](#)). Il présente aussi une variance plus petite que celle de $\hat{\tau}_{IPTW}$ ([Robins et Ritov, 1997](#); [Hirano et Imbens, 2002](#))⁴.

4.1.2 Analyse des pondérations calculées

Dans le cas de l'ajustement par pondération, la question du support commun peut se voir comme une analyse des pondérations affectées aux différentes unités. Si les covariables diffèrent grandement dans les deux groupes, le score de propension peut être proche de zéro ou de 1 pour certaines valeurs de X_i et le support commun d'autant plus réduit. Dans ce cas, certaines unités auront alors un poids important rendant ainsi l'estimation peu précise. Pour remédier à ce problème, une méthode consiste à remplacer les valeurs de score de propension supérieures à un maximum par la valeur de ce maximum (*trimming*). Cependant, si le score de propension estimé prend des valeurs proches de 0 ou 1, cela peut aussi correspondre à un problème de spécification rendant l'estimateur biaisé. Dans ce cas, il faut effectuer une estimation doublement robuste de l'effet, qui s'appuiera cependant sur l'hypothèse d'une spécification correcte de la fonction liant la variable d'intérêt aux covariables.

2. Cet estimateur est aussi plus efficace que celui construit en remplaçant $\hat{p}(X_i)$ par la *vraie* valeur du score de propension $p(X_i)$ lorsque ce dernier est connu (voir [Hirano et al., 2003](#)).

3. En pratique, il convient de *stabiliser* ces pondérations (*Stabilized Weights*), c'est-à-dire de les normaliser de telle sorte que la somme des poids soit égale à 1 dans chaque groupe, traité et de contrôle. Ce calcul est cependant fait implicitement dans les fonctions  présentées ici pour calculer l'effet souhaité, de telle sorte que l'expression des poids présentée dans ce document est celle que le chargé d'études doit implémenter informatiquement.

4. De même que l'estimateur utilisant les poids stabilisés présente une variance plus faible que celle de l'estimateur dont les poids ne sont pas stabilisés ([Lunceford et Davidian, 2004](#))

4.2 Mise en œuvre sous

L'ajustement par pondérations s'appuie sur une estimation préalable du score de propension et sur le calcul des pondérations correspondant à l'effet du traitement que l'on souhaite estimer (cf. section 4.1). Une fois les pondérations calculées (section 4.2.1), la vérification des propriétés équilibrantes du score de propension repose sur les mêmes diagnostics que pour les autres méthodes (différence standardisée de moyennes, de proportions, ratio de variances, etc.) mais sur les données pondérées (cf. section 4.2.2). Enfin, l'estimation de l'effet du traitement s'effectue directement par régression linéaire en estimant la variance de l'estimateur par *bootstrap* (section 4.2.3) et en incluant éventuellement les covariables déjà intégrées dans le calcul du score de propension.

4.2.1 Calcul des pondérations après estimation du score de propension

Dans le cas de la méthode d'ajustement par pondérations, il est souvent utile d'intégrer les carrés des covariables continues et/ou les interactions entre ces variables pour essayer d'obtenir⁵ une estimation convergente du score de propension. ... tout en conservant, cependant, un support commun large. Le score de propension une fois estimé, les pondérations sont ensuite calculées en fonction de l'effet du traitement recherché (cf. section 4.1)⁶.

Code : Calcul des pondérations

```
## Calcul des pondérations pour estimer l'effet moyen sur les traités
mybase$poids <- with(mybase,
                     treat+(1-treat)*ps.logit$fitted/(1-ps.logit$fitted))
```

4.2.2 Vérifier la propriété équilibrante après ajustement par pondération

Il faut ensuite renseigner dans l'argument `weights` de la fonction `bal.tab`, le vecteur de pondération à appliquer aux différentes observations, ainsi que `method="weighting"`, pour étudier la propriété équilibrante. Les autres arguments sont identiques à ceux présentés dans la section 1.2.1.

Code : Différences standardisées et ratio de variances

```
balIPTW <- bal.tab(formula = treat ~ age + educ + race +
                  married + nodegree + re74 + re75,
                  data = mybase,
                  distance = "pscore",
                  weights = "poids",
                  method = "weighting",
                  disp.v.ratio = TRUE,
                  un = TRUE,
                  s.d.denom = "treated")
```

5. via une modélisation flexible

6. On rappelle que les poids seront normalisés par les fonctions permettant de calculer l'effet du traitement par régression linéaire pondérée.

4. AJUSTEMENT PAR PONDÉRATION

Balance Measures:

	Type	Diff.Un	V.Ratio.Un	Diff.Adj	V.Ratio.Adj
pscore	Distance	1.9610	1.5637	-0.1024	1.0623
age	Contin.	-0.3094	2.2728	-0.0515	1.1509
educ	Contin.	0.0550	2.0166	-0.0532	1.0986
race_black	Binary	0.6404		-0.0314	
race_hispan	Binary	-0.0827		0.0110	
race_white	Binary	-0.5577		0.0205	
married	Binary	-0.3236		0.0320	
nodegree	Binary	0.1114		0.0696	
re74	Contin.	-0.7211	1.9300	0.1057	1.3297
re75	Contin.	-0.2903	1.0457	0.1102	1.5222

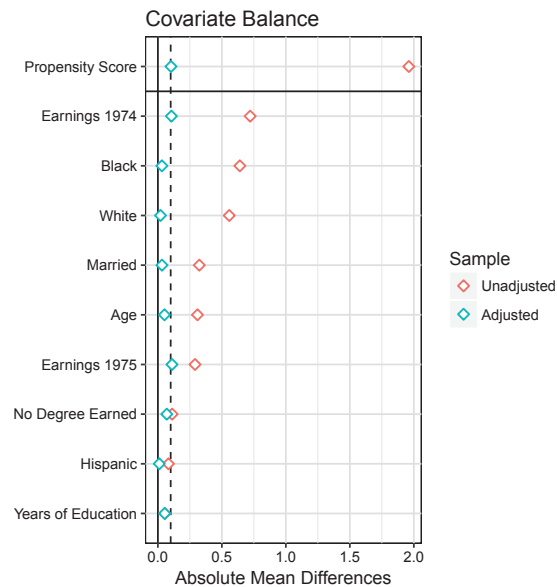
Effective sample sizes:

	Control	Treated
Unadjusted	429.00	185
Adjusted	29.79	185

Il est possible d'obtenir une représentation synthétique des résultats avant et après ajustement par pondérations avec la fonction `love.plot` déjà décrite à la section 1.2.1 (cf. Graphique 4.1).

Code : Différences standardisées avant / après ajustement par pondération

```
## Graphique des différences de moyennes (standardisées) et de proportions
love.plot(x = balIPTW,
  stat = "mean.diffs",
  abs = TRUE,
  ## options graphiques
  var.names = new.names,
  var.order = "unadjusted",
  threshold = 0.1,
  shape = 23)
```



Graphique 4.1 – Différences standardisées de moyennes avant et après ajustement par pondérations.

Au-delà de la moyenne et de la variance, il est aussi aisé de comparer graphiquement les densités des covariables continues avant et après ajustement par les pondérations avec la fonction `bal.plot`, en prenant soin de renseigner encore les arguments `weights` et `method`⁷.

Code : Comparaison des densités après ajustement par des pondérations

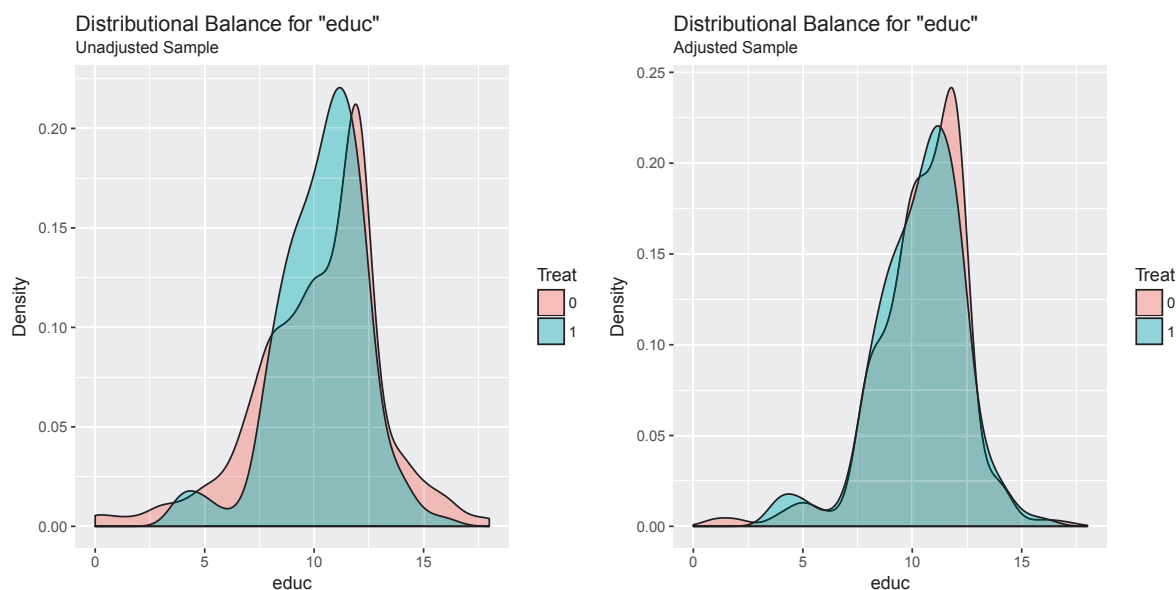
```
## Après ajustement avec des pondérations
bal.plot(obj = treat ~ age + educ + race + married +
         nodegree + re74 + re75,
         data = mybase,
         var.name="educ",
         weights = "poids",
         method = "weighting",
         estimand="ATT",
         which = "both")
```

Dans notre exemple, les unités du groupe de contrôle et du groupe traité présentent ainsi une distribution du nombre d'années d'études identique après ajustement par des pondérations (cf. Graphique 4.2).

4.2.3 Estimation de l'effet du traitement

L'estimation sous du l'effet du traitement s'appuie sur l'analogie avec la régression linéaire présentée dans la partie 4.1. Son implémentation peut donc être effectuée à l'aide de la fonction `glm`

7. pour les autres arguments, se reporter à la section 2.3.2



Graphique 4.2 – Densité du nombre d'années d'études avant et après ajustement par des pondérations.

et l'option `weights`⁸.

L'estimation de la variance de l'estimateur pondéré doit tenir compte du fait que les pondérations appliquées à chaque unité sont elles aussi estimées. Son expression est complexe et l'on privilégie donc la méthode du *bootstrap* pour étudier la significativité de l'effet estimé en construisant un intervalle de confiance. Cette méthode permet par ailleurs de ne pas faire d'hypothèse de normalité sur la distribution de l'estimateur pondéré (Imbens, 2014; Hirano *et al.*, 2003)⁹. Sous R, elle peut être mise en œuvre avec le package **boot**. Le principe est le suivant. On définit une fonction qui réestime, pour chaque échantillon tiré avec remise à partir de la base de données initiales, le score de propension et calcule l'effet du traitement avec les pondérations associées. L'appel de cette fonction (dans l'exemple, il s'agit de la fonction `iptw.boot`) par la fonction `boot` effectue ce calcul pour le nombre souhaité d'échantillons. La fonction `boot.ci` permet ensuite de déterminer plusieurs types d'intervalles de confiance.

Code R : Estimateur avec variance estimée par Bootstrap

```
## Chargement du package
library(boot)

## Définition de la fonction de calcul de l'estimateur souhaité
iptw.boot=function(dat,indices) {
  dat=dat[indices,]
  ## Estimation du score de propension
  ps.out <- glm(formula = treat ~ re74 + u74 + re75 + u75 +
```

8. Il est aussi possible d'utiliser les fonctions `svydesign` et `svyglm` du package **survey**. Les estimations obtenues sont identiques quelque soit la méthode privilégiée.

9. Les estimateurs robustes, de type sandwich, parfois préconisés et souvent utilisés car directement implémentés par les fonctions R ne sont pas à privilégier (voir par exemple Austin, 2016)


```

educ + race_black + race_hispan +
married + nodegree + age + I(age^2) +
nodegree:educ + re74:nodegree + u75:educ,
family = binomial(link=logit),
data = dat)
dat$pscore <- ps.out$fitted
## Calcul des pondérations pour estimer l'effet moyen sur les traités
dat$att.weights <- with(dat, treat + (1-treat)*pscore/(1-pscore))
## Calcul de l'effet moyen
glm.out=glm(re78 ~ treat,weight=att.weights,data=dat)
## Stockage de l'effet moyen estimé et de l'écart-type associé
summary(glm.out)$coefficients[2,1:2]
}

```

```

## Estimation initiale de l'effet du traitement
iptw.boot(mybase,indices=1:nrow(mybase))

```

```

Estimate Std. Error
1260.3767 552.7333

```

Dans notre exemple, l'effet moyen estimé, $\hat{\theta}$, est de 1260.4. Ce résultat est proche de celui obtenu par appariement sur le score de propension (1071 dollars pour l'estimation après correction du biais résiduel - cf. section 2.3.4). Pour déterminer un intervalle de confiance pour cet estimateur, on réitère l'estimation sur 2000 échantillons tirés aléatoirement avec remise à partir des données initiales. Les estimations pour chaque échantillon de l'effet moyen obtenus sont stockées dans l'objet `boot.out`.

```

## 2000 réplifications du calcul de l'estimateur
boot.out <- boot(data = mybase,
                 statistic = iptw.boot,
                 R = 2000)
## Calcul de l'intervalle de confiance
boot.ci(boot.out)

```

```

BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 2000 bootstrap replicates

```

```

CALL :
boot.ci(boot.out = boot.out)

```

```

Intervals :
Level      Normal              Basic              Studentized
95%  (-799, 3210 )  (-811, 3204 )  (-788, 3256 )

```

```

Level      Percentile              BCa
95%  (-683, 3331 )  (-766, 3263 )
Calculations and Intervals on Original Scale

```

La fonction `boot.ci` propose ensuite 4 intervalles de confiance pour l'estimateur $\hat{\theta}$ à partir des estimateurs du même effet obtenus sur les 2000 différents échantillons, $\hat{\theta}^*$. L'intervalle de confiance "normal" fait l'hypothèse que la distribution de $\hat{\theta}$ suit une distribution normale, et est donné par $\hat{\theta} - b \pm z_{\alpha} \hat{v}^{1/2}$. Dans cette formule, $b = \bar{\hat{\theta}}^* - \hat{\theta}$ corrige l'effet moyen estimé sur l'échantillon initial, $\hat{\theta}$ de la différence qu'il présente avec la moyenne des effets estimés sur les 2000 échantillons, $\bar{\hat{\theta}}^*$. Enfin, \hat{v} correspond à l'estimateur classique de la variance des estimations $\hat{\theta}^*$ pour les différents échantillons.

L'intervalle de confiance "basic" ou "Reverse Bootstrap Percentile" correspond à $[\hat{\theta} - (\theta_{1-\alpha/2}^* - \hat{\theta}); \hat{\theta} - (\theta_{\alpha/2}^* - \hat{\theta})]$ où $\theta_{\alpha/2}^*$ et $\theta_{1-\alpha/2}^*$ correspondent aux percentiles de la distribution des effets estimés, $\hat{\theta}^*$, sur les 2000 échantillons. Le principe de construction de cet intervalle de confiance est de considérer que la distribution de $\hat{\theta}^* - \hat{\theta}$ approxime celle du biais b .

L'intervalle de confiance "percentile" est donné par $[\theta_{\alpha/2}^*; \theta_{1-\alpha/2}^*]$, et l'intervalle "BCa" en est une version ajustée pour tenir compte de l'asymétrie éventuelle de la distribution de $\hat{\theta}^*$.

Enfin, l'intervalle de confiance « studentisé » (*studentized*) nécessite de calculer aussi pour chaque échantillon l'écart-type empirique de l'estimateur et correspond à $[\hat{\theta} - \hat{v}^{1/2} t_{1-\alpha/2}^*; \hat{\theta} - \hat{v}^{1/2} t_{\alpha/2}^*]$, où $t_{\alpha/2}^*$ et $t_{1-\alpha/2}^*$ sont les percentiles de la distribution de la statistique de test de Student des 2000 échantillons et v un estimateur de la variance de $\hat{\theta}$.

Quelque soit l'intervalle de confiance, "Normal", "Basic", "Studentized" "Percentile" ou "BCa", il découle des estimations que l'effet moyen sur les traités n'est pas significatif, comme déjà constaté avec l'estimation par appariement sur le score de propension.

Conclusion

L'ensemble des méthodes présentées dans ce document s'appuient sur le score de propension pour tenir compte des différences observables dans la comparaison entre un groupe d'unités traitées et non traitées. En pratique, le score de propension est estimé et l'on vérifie ensuite que les covariables présentent des distributions similaires dans le groupe traité et non traité après appariement, après avoir stratifié l'échantillon étudié selon les valeurs du score de propension estimé ou après pondération par l'inverse de la probabilité de traitement.

La pertinence de ces méthodes dépend souvent de la taille de l'échantillon étudié, mais aussi des covariables disponibles; aucune ne corrige des effets de sélection inobservée. Il est donc recommandé de mettre en œuvre toutes ces méthodes pour vérifier la robustesse des résultats.

Bibliographie

- A. ABADIE et G. W. IMBENS : Large Sample Properties of Matching Estimators for Average Treatment Effects. *Econometrica*, 74(1):235–267, January 2006.
- A. ABADIE et G. W. IMBENS : Bias-Corrected Matching Estimators for Average Treatment Effects. *Journal of Educational and Behavioral Statistics*, 29, 2011.
- Peter C. AUSTIN : Variance estimation when using inverse probability of treatment weighting (iptw) with survival analysis. *Statistics in Medicine*, 35(30):5642–5655, 2016.
- R. H. DEHEJIA et S. WAHBA : Causal Effects in Nonexperimental Studies : Reevaluating the Evaluation of Training Programs. *Journal of American Statistical Association*, 94:1053–1062, 1999.
- R. H. DEHEJIA et S. WAHBA : Propensity Score Matching Methods for Nonexperimental Causal Studies. *Review of Economics and Statistics*, 84:151–161, 2002.
- P. GIVORD : Méthodes économétriques pour l'évaluation des politiques publiques. *Document de travail de l'Insee*, G2010(08), 2010. URL <http://www.insee.fr/fr/themes/document.asp?id=2962>.
- N. GREIFER : *cobalt* : Covariate Balance Tables and Plots, 2016. URL <http://CRAN.R-project.org/package=cobalt>. R package version 1.3.1.
- B. B. HANSEN : Full Matching in an Observational Study of Coaching for the SAT. *Journal of American Statistical Association*, 99:609–618, 2004.
- J. J. HECKMAN, H. ICHIMURA et P. TODD : Matching as an Econometric Evaluation Estimator. *Review of Economics Studies*, 65:261–294, 1998.
- T. HENKE : *MatchItSE* : Calculates SE for Matched Samples from 'MatchIt', 2016. URL <https://CRAN.R-project.org/package=MatchItSE>. R package version 1.0.
- K. HIRANO et G. W. IMBENS : Estimation of Causal Effects using Propensity Score Weighting : An Application to Data on Right Heart Catheterization. *Health Services and Outcomes Research Methodology*, 2:259–278, 2002.
- K. HIRANO, G. W. IMBENS et G. RIDDER : Efficient Estimation of Average Treatment Effects using The Estimated Propensity Score. *Econometrica*, 71:1161–1189, 2003.
- D. E. HO, K. IMAI, G. KING et E. A. STUART : MatchIt : Nonparametric preprocessing for parametric causal inference. *Journal of Statistical Software*, 42(8):1–28, 2011. URL <http://www.jstatsoft.org/v42/i08/>.
- D.E. HO, K. IMAI, G. KING et E.A. STUART : matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. *Political Analysis*, 15:199–236, 2007.

- D. G. HOROVITZ et D. J. THOMPSON : A Generalization of Sampling Without Replacement from a Finite Universe. *Journal of The American Statistical Association*, 47:663–685, 1952.
- K. IMAI, G. KING et E.A. STUART : Misunderstandings among Experimentalists and Observationalists in Causal Inference. *Journal of the Royal Statistical Society : Series A*, 171:481–502, 2008.
- G. W. IMBENS : Matching Methods in Practice : Three Examples. *IZA Discussion Paper*, 8049, March 2014.
- G. W. IMBENS et D. R. RUBIN : *Causal Inference for Statistics, Social and Biomedical Sciences : An introduction*. Cambridge University Press, 2015.
- J. K. LUNCEFORD et M. DAVIDIAN : Stratification and weighting via the propensity score in estimation of causal treatment effects : a comparative study. *Statistics in Medicine*, 23:2937–2960, 2004.
- G. RIDGEWAY, D. MCCAFFREY, A. MORRAL, L. BURGETTE et B. A. GRIFFIN : *Toolkit for Weighting and Analysis of Nonequivalent Groups : a tutorial for the twang package*, april 2006. URL <https://cran.r-project.org/web/packages/twang/index.html>.
- J. M. ROBINS et Y. RITOV : Toward a Curse of Dimensionality Appropriate Asymptotic Theory for semi-parametric Models. *Statistics in Medicine*, 16:285–319, 1997.
- J. M. ROBINS, L. P. ZHAO et A. ROTNITZKY : Estimation of Regression Coefficients when Some Regressors Are not Always Observed. *Journal of the American Statistical Association*, 89(427):846–866, 1995.
- P. R. ROSENBAUM : A Characterization of Optimal Designs for Observational Studies. *Journal of the Royal Statistical Society : Series B*, 53:597–610, 1991.
- P. R. ROSENBAUM : *Observational Studies*. Springer, New York, 2 édition, 2002.
- P. R. ROSENBAUM et D. B. RUBIN : The Central Role of the Propensity Score in Observational Studies for Causal Effects. *Biometrika*, 70:41–55, 1983.
- P. R. ROSENBAUM et D. B. RUBIN : Reducing Bias in Observational Studies using Subclassification on the Propensity Score. *Journal of American Statistical Association*, 79:516–524, 1984.
- P. R. ROSENBAUM et D. B. RUBIN : Constructing a Control Group Using Multivariate Matched Sampling Methods That Incorporate The Propensity Score. *The American Statistician*, 39(1):33–38, February 1985.
- D. B. RUBIN : Matching to Remove Bias In Observational Studies. *Biometrics*, 29:159–184, 1973.
- D. B. RUBIN : Using Propensity Scores to Help Design Observational Studies : Application to the Tobacco Litigation. *Health Services and Outcomes Research Methodology*, 2:169–188, 2001.
- D. B. RUBIN et N. THOMAS : Matching Using Estimated Propensity Scores, Relating Theory to Practice. *Biometrics*, 52:249–264, 1996.
- D. O. SCHARFSTEIN, A. ROTNITZKY et J. M. ROBINS : Adjusting for Nonignorable Drop-Out Using Semiparametric Nonresponse Models. *Journal of the American Statistical Association*, 94(448):1096–1120, 1999.

Jasjeet S. SEKHON : Multivariate and propensity score matching software with automated balance optimization : The Matching package for R. *Journal of Statistical Software*, 42(7):1–52, 2011. URL <http://www.jstatsoft.org/v42/i07/>.

E. A. STUART : Matching Methods for Causal Inference : A Review and a look Forward. *Statistical Science*, 25(1):1–21, 2010.

F. SÄVJE, M. J. HIGGIN et J. S. SEKHON : Generalized Full Matching. arXiv 1703.03882, 2017.

Série des Documents de Travail « Méthodologie Statistique »

- 9601** : Une méthode synthétique, robuste et efficace pour réaliser des estimations locales de population.
G. DECAUDIN, J.-C. LABAT
- 9602** : Estimation de la précision d'un solde dans les enquêtes de conjoncture auprès des entreprises.
N. CARON, P. RAVALET, O. SAUTORY
- 9603** : La procédure **FREQ** de **SAS** - Tests d'indépendance et mesures d'association dans un tableau de contingence.
J. CONFAIS, Y. GRELET, M. LE GUEN
- 9604** : Les principales techniques de correction de la non-réponse et les modèles associés.
N. CARON
- 9605** : L'estimation du taux d'évolution des dépenses d'équipement dans l'enquête de conjoncture : analyse et voies d'amélioration.
P. RAVALET
- 9606** : L'économétrie et l'étude des comportements. Présentation et mise en œuvre de modèles de régression qualitatifs. Les modèles univariés à résidus logistiques ou normaux (**LOGIT**, **PROBIT**).
S. LOLLIVIER, M. MARPSAT, D. VERGER
- 9607** : Enquêtes régionales sur les déplacements des ménages : l'expérience de Rhône-Alpes.
N. CARON, D. LE BLANC
- 9701** : Une bonne petite enquête vaut-elle mieux qu'un mauvais recensement ?
J.-C. DEVILLE
- 9702** : Modèles univariés et modèles de durée sur données individuelles.
S. LOLLIVIER
- 9703** : Comparaison de deux estimateurs par le ratio stratifiés et application aux enquêtes auprès des entreprises.
N. CARON, J.-C. DEVILLE
- 9704** : La faisabilité d'une enquête auprès des ménages.
1. au mois d'août.
2. à un rythme hebdomadaire
C. LAGARENNE, C. THIESSET
- 9705** : Méthodologie de l'enquête sur les déplacements dans l'agglomération toulousaine.
P. GIRARD
- 9801** : Les logiciels de désaisonnalisation **TRAMO** & **SEATS** : philosophie, principes et mise en œuvre sous **SAS**.
K. ATTAL-TOUBERT, D. LADIRAY
- 9802** : Estimation de variance pour des statistiques complexes : technique des résidus et de linéarisation.
J.-C. DEVILLE
- 9803** : Pour essayer d'en finir avec l'individu Kish.
J.-C. DEVILLE
- 9804** : Une nouvelle (encore une !) méthode de tirage à probabilités inégales.
J.-C. DEVILLE
- 9805** : Variance et estimation de variance en cas d'erreurs de mesure non corrélées ou de l'intrusion d'un individu Kish.
J.-C. DEVILLE
- 9806** : Estimation de précision de données issues d'enquêtes : document méthodologique sur le logiciel **POULPE**.
N. CARON, J.-C. DEVILLE, O. SAUTORY
- 9807** : Estimation de données régionales à l'aide de techniques d'analyse multidimensionnelle.
K. ATTAL-TOUBERT, O. SAUTORY
- 9808** : Matrices de mobilité et calcul de la précision associée.
N. CARON, C. CHAMBAZ
- 9809** : Échantillonnage et stratification : une étude empirique des gains de précision.
J. LE GUENNEC
- 9810** : Le Kish : les problèmes de réalisation du tirage et de son extrapolation.
C. BERTHIER, N. CARON, B. NEROS
- 9901** : Perte de précision liée au tirage d'un ou plusieurs individus Kish.
N. CARON
- 9902** : Estimation de variance en présence de données imputées : un exemple à partir de l'enquête Panel Européen.
N. CARON
- 0001** : L'économétrie et l'étude des comportements. Présentation et mise en œuvre de modèles de régression qualitatifs. Les modèles univariés à résidus logistiques ou normaux (**LOGIT**, **PROBIT**) (version actualisée).
S. LOLLIVIER, M. MARPSAT, D. VERGER
- 0002** : Modèles structurels et variables explicatives endogènes.
J.-M. ROBIN
- 0003** : L'enquête 1997-1998 sur le devenir des personnes sorties du RMI - Une présentation de son déroulement.
D. ENEAU, D. GUILLEMOT
- 0004** : Plus d'amis, plus proches ? Essai de comparaison de deux enquêtes peu comparables.
O. GODECHOT
- 0005** : Estimation dans les enquêtes répétées : application à l'Enquête Emploi en Continu.
N. CARON, P. RAVALET
- 0006** : Non-parametric approach to the cost-of-living index.
F. MAGNIEN, J. POUGNARD
- 0101** : Diverses macros **SAS** : Analyse exploratoire des données, Analyse des séries temporelles.
D. LADIRAY
- 0102** : Économétrie linéaire des panels : une introduction.
T. MAGNAC
- 0201** : Application des méthodes de calages à l'enquête EAE-Commerce.
N. CARON
- C 0201** : Comportement face au risque et à l'avenir et accumulation patrimoniale - Bilan d'une expérimentation.
L. ARRONDEL, A. MASSON, D. VERGER
- C 0202** : Enquête Méthodologique Information et Vie Quotidienne - Tome 1 : bilan du test 1, novembre 2002.
J.-A. VALLET, G. BONNET, J.-C. EMIN, J. LEVASSEUR, T. ROCHER, P. VRIGNAUD, X. D'HAULTFOEUILLE, F. MURAT, D. VERGER, P. ZAMORA
- 0203** : General principles for data editing in business surveys and how to optimise it.
P. RIVIERE
- 0301** : Les modèles logit polytomiques non ordonnés : théories et applications.
C. AFSA ESSAFI
- 0401** : Enquête sur le patrimoine des ménages - Synthèse des entretiens monographiques.
V. COHEN, C. DEMMER
- 0402** : La macro **SAS** **CUBE** d'échantillonnage équilibré
S. ROUSSEAU, F. TARDIEU
- 0501** : Correction de la non-réponse et calage de l'enquêtes Santé 2002
N. CARON, S. ROUSSEAU

0502 : Correction de la non-réponse par ré pondération et par imputation
N. CARON

0503 : Introduction à la pratique des indices statistiques - notes de cours
J-P BERTHIER

0601 : La difficile mesure des pratiques dans le domaine du sport et de la culture - bilan d'une opération méthodologique
C. LANDRE, D. VERGER

0801 : Rapport du groupe de réflexion sur la qualité des enquêtes auprès des ménages
D. VERGER

M2013/01 : La régression quantile en pratique
P. GIVORD, X. D'HAULTFOEUILLE

M2014/01 : La microsimulation dynamique : principes généraux et exemples en langage R
D. BLANCHET

M2015/01 : la collecte multimode et le paradigme de l'erreur d'enquête totale
T. RAZAFINDROVONA

M2015/02 : Les méthodes de Pseudo-Panel
M. GUILLERM

M2015/03 : Les méthodes d'estimation de la précision pour les enquêtes ménages de l'Insee tirées dans Octopusse
E. GROS – K.MOUSSALAM

M2016/01 : Le modèle Logit Théorie et application.
C. AFSA

M2016/02 : Les méthodes d'estimation de la précision de l'Enquête Emploi en Continu
E. GROS – K.MOUSSALAM

M2016/03 : Exploitation de l'enquête expérimentale Vols, violence et sécurité.
T. RAZAFINDROVONA

M2016/04 : Savoir compter, savoir coder. Bonnes pratiques du statisticien en programmation.
E. L'HOUE – R. LE SAOUT B. ROUPPERT

M2016/05 : Les modèles multiniveaux

P. GIVORD – M. GUILLERM

M2016/06 : Econométrie spatiale : une introduction pratique
P. GIVORD – R. LE SAOUT

M2016/07 : La gestion de la confidentialité pour les données individuelles
M. BERGEAT

M2016/08 : Exploitation de l'enquête expérimentale Logement internet-papier
T. RAZAFINDROVONA

M2017/01 : Exploitation de l'enquête expérimentale Qualité de vie au travail
T. RAZAFINDROVONA