Université Paris Nanterre
200 av. de la République
92001 Nanterre Cedex
www.parisnanterre.fr

École doctorale 139 :
Connaissance, langage, modélisation
MODAL'X

Membre de l'université Paris Lumières

Antoine Rebecq

Méthodes de sondage pour les données massives

Thèse présentée et soutenue publiquement le 15/02/2019
en vue de l'obtention du doctorat de Mathématiques appliquées et applications des mathématiques de l'Université Paris Nanterre
sous la direction de M. Patrice Bertail (Université Paris Nanterre)

Jury :

Rapporteurs :	Anne Ruiz-Gazen Yves Tillé	Toulouse School of Economics Université de Neuchâtel
Examinateurs :	Émilie Chautru Emmanuel Gros Xavier Mary Éric Matzner-Løber	Mines ParisTech Insee Université Paris Nanterre Université Rennes 2

Abstract (English)

This thesis presents three different parts with ties to survey sampling theory.

In the first part, we present two original results that led to practical applications in surveys conducted at Insee (French official statistics Institute).

The first chapter deals with allocations in stratified sampling. We present a theorem that proves the existence of an optimal compromise between the dispersion of the sampling weights and the allocation yielding optimal precision for a specific variable of interest. Most sampling designs used in official surveys are created with requirements of precision levels for a collection of variables that are linked together. But the survey data is often used to estimate totals of other variables that are not correlated with the ones that were used to create the design in the first place. Expected precision for these variables is thus very low, to the point that it is sometimes impossible to use the estimates in practice. A low dispersion of weights is a method to lower the risks of a useful estimate being of very poor quality.

The second chapter deals with reweighting factors in calibration estimates. When computing weights for survey sampling data, using calibration on margins with distance functions allowing for bounds on the reweighting factors is frequent. One of the main advantages is to lower the risk of influential units appearing, particularly for estimates on sub-populations. Unfortunately, these algorithms do not offer any automatic way to compute bounds. We study an algorithm that computes the minimal bounds so that the calibration estimators exist, and propose an efficient way of resolution. We also study the statistical properties of estimates using these minimal bounds.

The second part studies asymptotic properties of sampling estimates. Obtaining asymptotic guarantees is often hard in practice. We present an original method that establishes weak convergence for the Horvitz-Thompson empirical process indexed by a class of functions for a lot of sampling algorithms used in practice.

In the third and last part, we focus on sampling methods for populations that can be described as networks.

The goal is to explore this field, that could have many applications in practical settings where the graphs are so big that storing and computing statistics on them can be very costly. The number of these cases has been increasing a lot recently, thanks to the rise of successful Internet and social network businesses. We start by presenting useful statistics to describe networks. Then we briefly expose the main probabilistic frameworks to model graphs. Finally, we show different sampling algorithms to select nodes and vertices to create samples and estimates of statistics of interest.

We close this part on two applications. The first one applies sampling algorithms to compute some statistics on Twitter data. The second one is a study using simulations to establish guidelines to design efficient sampling designs for graphs.

Keywords: *Big data, sampling, calibration, empirical process, networks*

Abstract (Français)

Cette thèse présente trois parties liées à la théorie des sondages.

La première partie présente deux résultats originaux de sondages qui ont eu des applications pratiques dans des enquêtes par sondage de l'Insee (Institut National des Statistiques et des Études Économiques - France).

Le premier article présente un théorème autorisant un plan de sondage stratifié constituant un compromis entre la dispersion des poids et l'allocation de précision optimale pour une variable d'intérêt spécifique. La plupart des plans de sondage utilisés pour des enquêtes de statistique officielle sont conçus avec des contraintes de précision minimale sur une collection de variables liées entre elles. Mais les données sont en réalité souvent utilisées pour estimer nombre d'autres totaux ou modèles issus de variables exclues du design. La précision attendue pour ces variables est donc faible - empêchant parfois toute exploitation pratique. Une dispersion des poids faible permet de limiter les risques qu'une estimation dépendant d'une de ces variables ait une très mauvaise précision.

Le second article concerne le facteur de repondération dans les estimateurs par calage. Lors de la construction pratique d'un estimateur sur données de sondage, il est fréquent d'utiliser une fonction de pseudo-distance permettant de borner le facteur multiplicatif - ceci principalement afin de limiter les risques d'apparition d'unités influentes, particulièrement pour l'estimation sur des domaines. Malheureusement, les algorithmes existants requièrent le plus souvent de fixer les bornes manuellement, sans offrir de critère statistique pour aider à la décision. On propose un algorithme efficace capable de calculer les facteurs de poids les plus rapprochés autour de 1 tels qu'une solution au problème de calage existe. On étudie par simulations sur données réelles les propriétés statistiques des estimateurs obtenus.

La seconde partie concerne l'étude des propriétés asymptotique des estimateurs sur données issues de sondage. Celles-ci sont difficiles à étudier en général. On présente une méthode originale qui établit la convergence faible vers un processus gaussien pour le processus empirique d'Horvitz-Thompson indexé par des classes de fonction, pour de nombreux algorithmes de sondage différents utilisés en pratique.

Dans la dernière partie, on s'intéresse à des méthodes de sondage pour des données issues de graphes.

L'objectif est d'explorer un champ peu étudié et qui peut avoir des applications dans des cas pratiques où les graphes sont de taille telles que leur stockage, leur exploration et le calcul de statistiques ou de certains algorithmes peut être très coûteux. Ces cas sont de plus en plus nombreux, particulièrement depuis l'avènement des réseaux sociaux comme acteurs majeurs de l'industrie. On commence par détailler les statistiques d'intérêt usuelles pour des données de réseaux. Puis on présente succintement les modèles probabilistes de population de graphes, et différents algorithmes de sondage pour sélectionner les arêtes et noeuds d'un réseau.

Deux applications concluent cette partie. On présente d'abord l'application d'algorithmes de sondage présentés précédemment pour estimer des statistiques issues des données de Twitter. Enfin, une étude par simulation permet de donner des recommendations pour élaborer un plan de sondage efficace en fonction du modèle de superpopulation et des statistiques d'intérêt.

Mots-clés: *Mégadonnées, sondages, calage, processus empirique, réseaux*

à Alice et Diane

Remerciements

Je tiens à remercier le directeur de cette thèse, Patrice Bertail, de m'avoir fait confiance et guidé avec bienveillance durant ces quatre années. Les méthodes de démonstration techniques en statistique m'étaient complètement inconnues au début de ce travail, et sa pédagogie ainsi que ses idées nouvelles m'ont permis de beaucoup progresser dans ce domaine. Par ailleurs, une bonne partie du travail effectué pendant ces quatre ans se situe à la frontière entre mathématiques et informatique. L'intérêt qu'a montré Patrice pour ces travaux très appliqués a suscité de nombreuses discussions très fructueuses.

Je voudrais remercier mon encadrement à la Direction de la méthodologie et de la coordination statistique et Internationale de l'Insee, Sébastien Faivre, Olivier Sautory, Stéphane Grégoire et Sylvie Lagarde pour m'avoir permis de consacrer du temps pour ma recherche. Ce doctorat n'aurait pas pu aboutir sans cela.

Je voudrais également remercier chaleureusement Yves Tillé et Anne Ruiz-Gazen de l'honneur qu'ils m'ont fait en acceptant d'être les rapporteurs de cette thèse.

Un grand merci à Xavier Mary, Emmanuel Gros, Éric Matzner-Løber et Émilie Chautru d'avoir accepté de faire partie du jury.

L'application des méthodes statistiques à l'industrie est un fil rouge de ce manuscrit, et un domaine qui me tient particulièrement à cœur. Merci à mes collègues d'Ubisoft Montréal, particulièrement à Olivier Delalleau et Jonathan Dumas pour les discussions stimulantes, ainsi qu'à Isabelle Belley-Ferris pour sa confiance.

Enfin, je voudrais remercier mes collègues de la division Sondages de l'Insee, spécialement Thomas Merly-Alpa, Emmanuel Gros (une mention spéciale pour avoir trouvé l'acronyme "Icarus"), Martin Chevalier, Thomas Deroyon, Cyril Favre-Martinoz, Ronan Le Gleut, Pierre-Arnaud Pendoli, pour avoir su créer et maintenir un environnement intellectuellement stimulant autour des sondages, un domaine malheureusement trop méconnu. Les échanges autour des travaux de chacun m'auront permis de beaucoup progresser, et la grande majorité des contributions originales de cette thèse n'auraient pu voir le jour sans eux.

Contents

1 Introduction **1**

I Échantillonnage avec objectifs multiples **17**

2 Optimisation d'une répartition mixte **19**
2.1 Introduction 19
2.2 Programme d'optimisation 23
2.3 Choix du λ 24
2.4 Application pratique 27
2.5 Conclusion 33

3 Calage sur bornes minimales et choix des bornes de calage **37**
3.1 Le calage sur bornes minimales 40
3.1.1 Notations et programme 40
3.1.2 Résolution du programme 42
3.2 Impact du choix des bornes de calage sur la qualité des estimateurs calés 43
3.2.1 Données et protocole utilisés pour l'étude par simulations . . 43
3.2.2 Résultats 44

4 Icarus : un package R pour le calage sur marges et ses variantes **57**
4.1 Introduction 57
4.2 Calage sur marges 58
4.2.1 Notations 58
4.2.2 Exemple 59
4.3 Calage pénalisé 61
4.3.1 Principe 61
4.3.2 Exemple 62
4.4 Calage sur bornes serrées 64
4.5 Tests unitaires et développements futurs 65

II Asymptotics of survey sampling **67**

5 Asymptotics of survey sampling **69**

5.1 Negative association of survey sampling and resampling designs . . . 72
5.1.1 Notations and definitions . . . 72
5.1.2 Negative association of survey sampling designs . . . 74
5.2 Some negatively associated sampling algorithms . . . 76
5.2.1 Rejective sampling . . . 77
5.2.2 Determinantal sampling . . . 79
5.2.3 Pivotal (or Srinivasan) sampling . . . 79
5.2.4 Cluster and multistage sampling designs . . . 80
5.3 A tailor-made CLT for negatively associated sampling plans . . . 84
5.4 Functional central limit theorems for negatively associated survey sampling . . . 91
5.4.1 Framework for empirical processes in survey sampling (single stage) . . . 92
5.4.2 Convergence of the HT-empirical process . . . 94
5.5 Applications and simulations . . . 99
5.5.1 Linearization and asymptotics of non linear functionals . . . 99
5.5.2 Confidence bands for a CDF : a simulation study . . . 101
5.6 Asymptotics of graph sampling . . . 105

III Sampling big networks **109**

6 Graphs: statistics and sampling **111**
6.1 Introduction: Big graphs and sampling . . . 111
6.2 Graphs and statistics . . . 112
6.2.1 Basic definitions and concepts . . . 112
6.2.2 Some statistics and algorithms . . . 115
6.3 Graph sampling . . . 122
6.3.1 Definition . . . 122
6.3.2 A note on graph observability . . . 122
6.3.3 Simple sampling algorithms . . . 125
6.3.4 Extension sampling algorithms . . . 128
6.4 Random graph models . . . 130
6.4.1 The Erdős-Rényi (random graph) model . . . 132
6.4.2 The Barabási-Albert (preferential attachment) model . . . 132
6.4.3 The Watts-Strogatz (small world) model . . . 133
6.4.4 The forest-fire model . . . 134
6.4.5 Other graph models . . . 134

7 Application: the Twitter graph **135**
7.1 Introduction . . . 135
7.1.1 Problem . . . 136
7.1.2 The Twitter graph . . . 138

7.2 Sampling designs . . . 139
7.2.1 Notations . . . 140
7.2.2 Bernoulli sampling . . . 140
7.2.3 Stratified Bernoulli sampling . . . 143
7.2.4 One stage snowball sampling . . . 145
7.2.5 Stratified adaptive cluster sampling . . . 146
7.3 Estimates . . . 148
7.3.1 Horvitz-Thompsom estimator . . . 148
7.3.2 One stage snowball sampling . . . 149
7.3.3 Estimators for adaptive sampling . . . 150
7.3.4 Calibration on margins . . . 153
7.4 Variance and precision estimation . . . 155
7.4.1 Simple designs . . . 155
7.4.2 Snowball sampling . . . 156
7.4.3 Adaptive designs . . . 157
7.4.4 Adjustment for calibration . . . 158
7.5 Results . . . 158
7.5.1 Estimates . . . 158
7.5.2 Size of the extensions . . . 159
7.6 Conclusion and future work . . . 160
7.6.1 Summary . . . 160
7.6.2 Future work . . . 161

8 Practical efficiency of snowball designs **163**
8.1 Introduction . . . 163
8.2 Inefficiency of plugin sampling . . . 164
8.3 Measuring the efficiency of snowball extension . . . 167
8.3.1 Design effect and estimators . . . 167
8.3.2 Variables of interest and population networks . . . 171
8.3.3 Simulations . . . 174
8.4 Proportional-to-size sampling . . . 178
8.5 Conclusion . . . 180

Bibliography **185**

1 Introduction

Ce travail de thèse propose des réflexions autour de l'estimation par sondage.

Dans une première partie, on se concentre sur le cas d'une enquête par sondage en présence de multiples objectifs. On propose l'étude de deux méthodes nouvelles permettant d'assurer une bonne qualité d'estimation sur un ensemble de variables, parfois décorrélées de la variable d'intérêt principale. On montre ensuite des applications de ces travaux en statistique officielle, au sein d'enquêtes menées à l'Insee (Institut National de la Statistique et des Études Économiques).

La deuxième partie utilise une approche originale pour prouver la convergence faible du processus empirique d'Horvitz-Thompson indexé par des classes de fonctions vers un processus gaussien, dans le cas de différents algorithmes de sondage utilisés en pratique, incluant les sondages à plusieurs degrés.

Dans la troisième et dernière partie, on montre que les méthodes de sondage peuvent être utile pour résoudre des problèmes pratiques qui se posent dans l'industrie numérique ("big data"). On retient le cas particulier des graphes ; on montre une application pratique issue de données du réseau social Twitter et on établit des stratégies directrices pour les praticiens travaillant sur des données de réseaux.

Part I - Échantillonnage avec objectifs multiples

Chapitre 2

Dans ce chapitre, on étudie un plan de sondage aléatoire simple stratifié (à un degré). On estime avec les données de sondage le total $T(Y)$ ou la moyenne $\bar{Y}$ d'une variable d'intérêt principale continue y, avec y_k la valeur de y pour le k-ième individu de la population. Les données d'enquêtes sont également utilisées pour estimer une collection d'autres variables, qui sont parfois décorrélées ou anti-corrélées de y. Le but est de modéliser une situation qui se présente régulièrement en statistique officielle, lorsque les données d'enquête sont également utilisées pour estimer une collection d'autres variables, qui peuvent être décorrélées ou anti-corrélées de y.

Le choix d'un plan de sondage permet en général de répondre à un but précis selon la règle classique "un objectif, un échantillon". Cela s'applique dans le cas du sondage stratifié au choix d'une allocation. Si on veut estimer le total $T(Y)$ de la variable d'intérêt avec une précision maximale, on peut utiliser l'allocation de Neyman [Ney34]. De manière plus générale, les allocations répondent à des contraintes relatives à des quantités dérivées de y. Si les données d'enquêtes sont utilisée pour l'estimation de quantités issues d'autres variables, il s'agit de faire en sorte que l'estimation ne soit pas dégradée par le choix du plan de sondage. Et ce même si les variables en question n'ont pas été incluses dans la création du plan. Or, il est connu que la variance de l'estimation du total d'une variable décorélée ou anti-corrélée aux variables utilisées pour le calcul d'une allocation spécifique peut être très élevée.

Pour régler ce problème, on peut utiliser l'allocation proportionnelle, qui est "agnostique" (mais ne fait pas un bon usage de l'information auxiliaire). Cela évite de construire un plan qui sera délétère pour l'estimation de certaines variables, de certains paramètres autres que des totaux ou des moyennes ou pour l'estimation sur des domaines spécifiques.

Plus formellement, notons H le nombre de strates d'une population stratifiée de taille N en strates de taille N_h, et intéressons nous à un plan stratifié de taille fixe n. Le choix d'une allocation consiste à déterminer un vecteur $\mathbf{n} = (n_1, \ldots, n_H)$ vérifiant la contrainte $\sum_{h=1}^{H} n_h = n$. Afin de modéliser les contraintes des enquêtes, nous allons chercher un compromis entre l'allocation proportionnelle et une autre allocation spécifique, répondant à un besoin précis sur une ou plusieurs variables d'intérêt d'une enquête. Nous appelons ceci l'allocation "mixte" :

$$\mathbf{n}_\alpha = \alpha \mathbf{n}_{prop} + (1-\alpha)\mathbf{n}_{specifique} \tag{1.1}$$

où $\mathbf{n}_\alpha$, $\mathbf{n}_{prop}$ et $\mathbf{n}_{specifique}$ sont des vecteurs de taille H et $\alpha \in [0,1]$. L'allocation proportionnelle est définie par :

$$n_h = n\frac{N_h}{N},\ h = 1, \ldots, H$$

L'allocation proportionnelle est choisie ici car elle minimise la dispersion des poids (on discute également ici de la pertinence de cet objectif). Le but est de proposer une

méthode pour déterminer une valeur satisfaisante de α. On cherche ainsi à calculer un paramètre remplissant un certain critère d'optimalité défini en section 2.3.

Dans ce chapitre, on formalise le programme d'optimisation qui correspond à ces contraintes. On donne ensuite un théorème qui définit le critère d'optimalité à résoudre. On propose une méthode de calcul pour l'optimum. Puis on effectue une étude par simulations pour discuter de l'efficacité de la méthode proposée, en comparant avec d'autres méthodes de calcul d'allocations de compromis.

Principales contributions

Dans le chapitre 2, l'écriture du programme d'optimisation consistant à chercher un compromis entre la dispersion des poids et une allocation spécifique est originale.

D'autres allocations de compromis sont classiques dans la théorie des sondages : celles de [Ban88] et [KM09] par exemple permettent d'améliorer la précision sur des domaines, celles de Cochran [Coc63] et Chatterjee [Cha67] proposent une allocation optimale pour plusieurs variables d'intérêt. L'originalité du programme présenté au paragraphe 2.3 tient dans l'écriture du compromis entre un critère de robustesse (dispersion des poids la plus faible possible) et un critère de spécificité (allocation conçue pour un petit nombre de variables d'intérêt principales).

Le programme est le suivant :

$$\min_{\alpha\in[0,1]} \quad \sum_{h=1}^{H} n_{\alpha,h}\left(\frac{N_h}{n_{\alpha,h}} - \frac{N}{n}\right)^2 + \lambda\, Dist(\mathbf{n}_\alpha, \mathbf{n}_{specifique}) \tag{1.2}$$

où $Dist$ est un opérateur de distance entre deux vecteurs de allocations et λ est un réel positif. Ce programme correspond à une généralisation de la méthode ROAUST ([Chi+17], [Chi+10a]) où l'on remplace l'allocation de Neyman par n'importe quelle autre allocation spécifique. En plus de cette généralisation, cette méthode présente par rapport à ROAUST deux ajouts :

- La possibilité de remplacer l'allocation de Neyman par une autre allocation spécifique
- La possibilité de calculer une valeur optimale du paramètre de compromis α ([Chi+10a] propose $\alpha = 0.50$)

Le théorème qui assure l'existence d'une valeur optimale est le suivant :

Theorem 1. *Soit $V(\lambda)$ la fonction de variance de l'estimateur d'Horvitz-Thompson pour l'allocation obtenue pour le α solution du programme de minimisation de l'équation (1.2) pour un λ donné et où l'allocation choisie est celle de Neyman. Alors, il existe un segment $S \subset [0, +\infty[$ tel que :*

- *$\alpha(S) = [0, 1]$, où $\alpha(\lambda)$ associe à λ la solution du programme 1.2.*
- *$V(\lambda)$ est décroissante sur S.*
- *Sa dérivée seconde admet un maximum dans S qu'on appelle **point de torsion**.*

En termes pratiques, cela montre qu'il existe dans ce compromis une zone plate aux alentours de l'allocation spécifique. Parvenir à trouver un plan de sondage qui correspond à cette zone permet donc, en l'échange d'une perte de précision très faible sur la variable d'intérêt principale, de gagner fortement en robustesse (dispersion des poids beaucoup plus faible). La figure 1.1 explique cette logique :

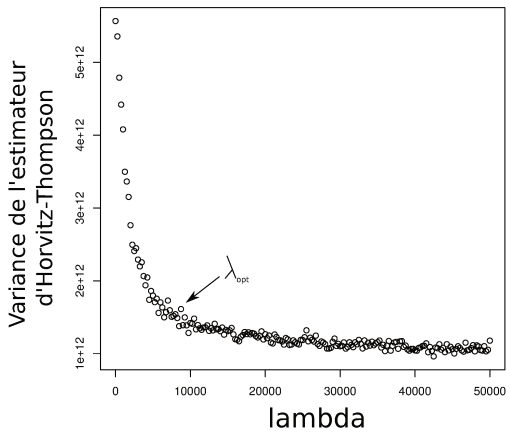


Fig. 1.1: Variance de l'estimateur d'Horvitz-Thompson en fonction du paramètre λ pour un jeu de données explicité au chapitre 2. Le choix de λ optimal (dans le coude de la courbe) est indiqué par un flèche.

On propose également une méthode de calcul pratique pour ce λ correspondant à l'allocation optimale, et on conjecture que le résultat est toujours valable pour d'autres allocations spécifiques.

Cette méthode a ensuite été mise en pratique avec succès au sein de l'Institut National de la Statistique et des Études Économiques (Insee) pour plusieurs enquêtes (notamment ECMOSS[1]).

[1] Enquête sur le Coût de la Main d'Oeuvre et la Structure des Salaires

Chapitre 3

Dans ce chapitre, on ne s'intéresse non plus au plan d'échantillonnage, mais au redressement de l'enquête, effectuée une fois que la collecte est terminée. Plus particulièrement, on utilise la méthode du calage sur marges ([DS92]). Les estimateurs par calage sont couramment utilisés en statistique officielle avec deux objectifs. Le premier objectif recherché est la cohérence entre les chiffres publiés issus de plusieurs sources différentes (bases administratives ou enquête par sondage). Le second, plus importante d'un point de vue statistique, consiste à améliorer la précision de l'estimation par rapport à l'estimation non redressée.

On dispose de variables auxiliaires dont on connaît les totaux sur la population, et l'on utilise cette information pour calculer des poids linéaires les "plus proches possibles" des poids de sondage et qui permettent de proposer des estimateurs exacts pour les variables dont les totaux sont connus. Plus formellement, on se donne une population $\mathcal{U}$ de taille N et un échantillon s de taille n, sélectionné par un plan de sondage p, de probabilités d'inclusions simples $(\pi_k)_{k\in\mathcal{U}}$. On note $d_k = 1/\pi_k$ les poids de sondages associés à l'estimateur Horvitz-Thompson. On dispose de J variables auxiliaires X_j, dont les totaux sur la population T_{X_j} sont connus par une source externe à l'enquête. Les poids calés w_k issus d'un calage sur ces J variables auxiliaires sont solutions du problème d'optimisation suivant :

$$\begin{cases} \min\limits_{w_k} \sum\limits_{k\in s} d_k G(\dfrac{w_k}{d_k}) \\ \text{s.c. } \sum_{k\in s} w_k \mathbf{x}_k = T_{\mathbf{X}} \end{cases} \tag{1.3}$$

où G est la fonction de (pseudo-)distance associée à la méthode de calage retenue (elle doit vérifier certaines conditions peu restrictives).

Le cadre d'intérêt est le même que celui du chapitre 2 : le plan de sondage ainsi que celui de redressement sont avant tout faits pour l'estimation d'une variable d'intérêt principale y, mais on doit garder à l'esprit que d'autres variables que celles qui ont été utilisées pour le redressement et le calcul des probabilités d'inclusion seront exploitées.

Dans la pratique du redressement des enquêtes, on souhaite donc souvent introduire des variables de calage nombreuses et variées. Malheureusement, le programme de calage, lorsqu'il est fortement contraint par de nombreuses équations à respecter, produit des repondérations fortes (à la hausse comme à la baisse). Cela rend les estimations finales peu robustes. Il est toutefois possible de contraindre l'étendue

des rapports de poids en utilisant la méthode logit ([DS92]). Cette méthode possède elle aussi des inconvénients.

Les fonctions de distance de ce type sont utiles dans des domaines variés, allant de la statistique (estimation par méthodes de quasi-vraisemblance empirique, traitées dans [Ber+14]) à l'optimisation (voir par exemple [Bor+03]).

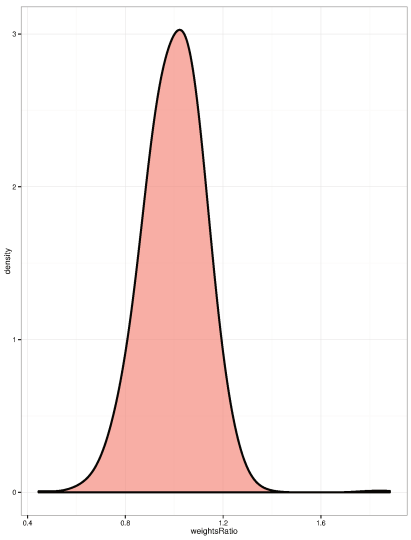
Premièrement, une solution n'existe pas nécessairement pour le choix de bornes effectué. Pour cela, on étudie le calage sur bornes minimales ([Gra14], [Van01]), qui permet de trouver les bornes minimisant l'étendue des facteurs de calage $g_k = w_k/d_k$ telles qu'une solution existe pour le problème de calage donné. Autrement dit, on veut déterminer les bornes L^* et U^* assurant une étendue minimale des **facteurs de calage** $g_k = w_k/d_k$. En notant $\widetilde{X}_s$ la matrice dont les coefficients (k, j) sont égaux à $(d_k \cdot X_{jk})$ pour $k = 1, \ldots, n$ et $j = 1, \ldots, J$, t le vecteur des totaux T_{X_j} et g le vecteur des facteurs de calage, les équations de calage s'écrivent sous la forme $\widetilde{X}_s' g = t$, et les bornes L^* et U^* sont alors solutions du programme de minimisation linéaire suivant :

$$\begin{cases} \min\limits_{g \in \mathbb{R}^n} \left(\max\limits_{k \in [[1,n]]} g_k - \min\limits_{k \in [[1,n]]} g_k \right) \\ \qquad \text{s. c. } \widetilde{X}_s' g = t \ ; \ g \geq 0 \end{cases} \tag{1.4}$$

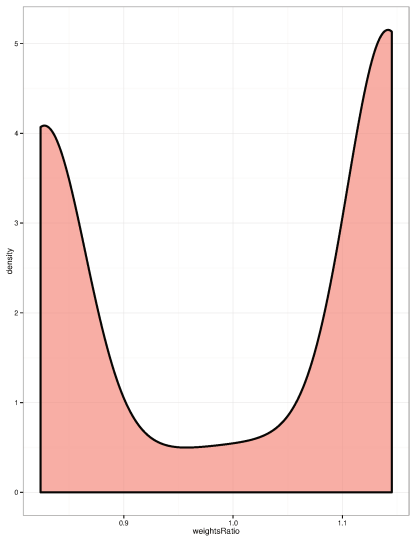
Deuxièmement, lorsque les bornes sont proches de l'étendue minimale, la distribution des poids de calage tend à montrer de fortes accumulations proches des bornes. La distribution présente alors une typique forme "en U", illustrée en figure 1.2.

On discute des raisons pour lesquelles on soupçonne que cette distribution est néfaste pour la qualité des estimations issues de ces poids calés, particulièrement lorsque l'exploitation est faite sur des domaines de diffusions. On effectue une étude par simulations, qui montre qu'on peut attendre une erreur quadratique d'estimation plus forte dès lors que la distribution présente de fortes accumulations.

On propose alors une heuristique qui consiste à trouver un compromis entre bornes à étendue réduite et limitation des accumulations (forme de distribution se rapprochant d'une gaussienne).



(a) Méthode linéaire - la distribution a une forme gaussienne



(b) Méthode sur bornes minimales - la distribution présente une caractéristique forme “en U”

Fig. 1.2: Distribution des facteurs de repondération pour un problème de calage avec deux choix de méthodes différents

Principales contributions

Dans le chapitre 3, on s’intéresse à un algorithme proposé par [Gra14], reposant sur le simplexe de [Dan+55] pour trouver des bornes de redressement les plus serrées autour de 1 lorsqu’une méthode bornée de calage est utilisée (logit par exemple). On montre que le programme (1.4) est équivalent au programme suivant :

$$\begin{cases} \min\limits_{g\in\mathbb{R}^n, a\in\mathbb{R}} \max\limits_{k\in[[1,n]]} |g_k - a| \\ \quad \text{s. c. } \widetilde{X}_s' g = t \ ; \ g \geq 0 \end{cases} \tag{1.5}$$

qui est plus efficace en mémoire.

On s’intéresse également aux propriétés statistiques des estimateurs par calage utilisant des bornes proches des bornes minimales, en regardant particulièrement le cas de l’estimation par domaine. Souvent, les distributions de facteurs de calage présentent une forme en U simiaire à celle de la figure 1.2 lorsque les bornes sont serrées. On montre par simulations sur données réelles de l’Insee qu’une telle forme de distribution avec beaucoup d’accumulations est associée à une erreur quadratique moyenne d’estimation plus forte qu’avec une fonction de distance aboutissant à

une distribution avec moins d'accumulations aux bornes (plus proche d'une forme gaussienne).

Chapitre 4

Dans ce chapitre, on présente le package R Icarus, qui implémente le calage sur marges et quelques unes de ses variantes, dans une interface proche de celle de la macro Calmar ([Sau03]). Icarus est disponible sur le répertoire CRAN[2], et permet donc une installation facile pour les utilisateurs.

Pour chaque calage lancé, Icarus donne dans la console des statistiques décrivant le redressement effectué : distribution des rapports de poids (ainsi qu'un graphique de la densité de cette distribution dans la fenêtre graphique), écart entre les estimtations initiales et les marges pour toutes les variables (et toutes les modalités lorsqu'il s'agit de variables quantitatives). La figure 1.3 illustre une fenêtre de travail avec Icarus sous le logiciel RStudio[3].

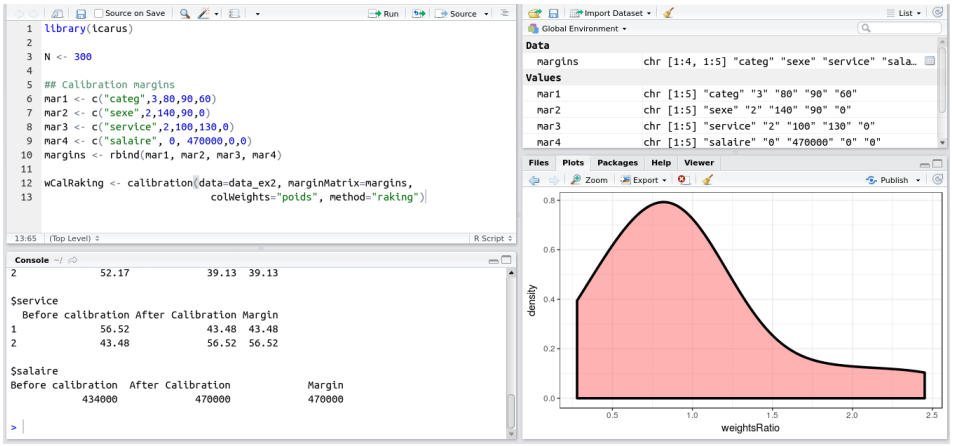


Fig. 1.3: Exemple de travail avec Icarus sous l'*IDE* Rstudio

Icarus implémente également certaines variantes du calage sur marges qui ne sont pas disponibles dans d'autres packages R en date de mai 2018 : le calage sur bornes minimales (étudié au chapitre 3), et aussi le calage pénalisé.

Le calage pénalisé consiste à relâcher la contrainte d'estimation exacte pour certaines marges de calage. Cela permet au programme de converger plus facilement, avec des structures de poids moins déformées. Les bonnes propriétés statistiques du calage

[2]The Comprehensive R Archive Network, `https://cran.r-project.org/`
[3]https://www.rstudio.com/

original (réduction de variance, biais asymptotiquement nul) sont conservées. On remplace le programme de calage classique (1.3) par le programme suivant :

$$\min_{w_k} \left[\lambda^* \sum_{k \in s} d_k G(\frac{w_k}{d_k}) + (1 - \lambda^*) \left[\sum_{j=1}^{J} c_j \left(\sum_{k \in s} w_k x_{jk} - T(X_j) \right)^2 \right] \right] \tag{1.6}$$

où c_j est le coût associé au calage non-exact sur le total $T(X_j)$ et λ^* est un réel compris entre 0 et 1, qui permet de contrôler l'étendue des facteurs de repondération.

Enfin, la construction de chaque nouvelle version de Icarus repose sur une intégration continue stricte. Des tests unitaires permettent de vérifier que chaque mise à jour n'entraîne pas de régression. Certains de ces tests mettent en jeu des résultats issus de calages effectués dans les mêmes conditions avec la macro Calmar, assurant que le passage en production des enquêtes officielles donne des résultats cohérents.

Principales contributions

Le package R Icarus, logiciel libre créé durant ce travail de thèse, implémente plusieurs fonctions utiles pour effectuer un redressement en production.

Icarus a pour but de donner une interface simple pour les statisticiens peu familiers avec les langages de programmation ou la théorie du redressement. Le calage sur marges est par exemple disponible dans les packages R [TM16] et [Lum+04], mais obligent l'utilisateur à créer lui-même la matrice utilisée dans le programme d'optimisation. Dans Icarus, les utilisateurs familiers avec l'interface de la macro Calmar ([Sau03]), très populaire au sein de la statistique publique, retrouveront une syntaxe proche. La syntaxe est également compatible avec l'opérateur "pipe" %>% ([BW14]), de plus en plus fréquemment utilisé et enseigné.

Par ailleurs, certaines de ses variantes qui n'étaient jusque là pas disponibles pour la production (calage sur bornes minimales, calage pénalisé), sont présentes dans le package. L'implémentation pour ces deux méthodes est efficace : le calage pénalisé utilise une méthode de Newton pour l'optimisation, et le calage sur bornes minimales utilise le simplexe sur des matrices sparse.

Dans tous les cas, des statistiques résumant le redressement sont affichées et disponibles à l'export (ce qui peut être utile pour écrire un rapport sur le redressement ou effectuer simulations par exemple)

Enfin, une intégration continue soignée assure la cohérence des résultats avec, entre autres, la macro SAS Calmar, et permet de minimiser le risque de bogue ou de régression.

Icarus a été utilisé en production à l'Insee (enquête Logement, enquête CVS [4]), et a également permis de mettre le calage pénalisé en production dans l'enquête ENRJ[5].

Part II - Asymptotics of survey sampling

Chapitre 5

Enfin, on présente une méthode originale pour obtenir un Théorème Central Limite pour les données observées par sondage utilisant le concept d'association négative ([BJ12]):

Définition. *Les variables aléatoires* $Z_1, \ldots, Z_n$ *sont dites négativement associées (NA) si pour toutes paires de sous-ensembles disjoints* A_1 *et* A_2 *de* $[\![1, N]\!]$ *on a :*

$$Cov\left(f((Z_i)_{i\in A_1}),\ g((Z_j)_{j\in A_2})\right) \leq 0, \tag{1.7}$$

pour toute fonctions croissantes mesurables à valeurs réelles $f : E^{\#A_1} \to \mathbb{R}$ *et* $g : E^{\#A_2} \to \mathbb{R}$.

On considère le cas de nombreux d'algorithmes de tirage et on montre qu'ils vérifient cette propriété. Utilisant ceci ainsi qu'un résultat général sur l'asymptotique des variables associées négativement, on donne des conditions sur les plans de sondage pour que le processus empirique d'Horvitz-Thompson indexé par des classes de fonctions converge en loi vers un processus gaussien. Cela étend des résultats donnés dans [Ber+16] à des plans de sondages plus généraux.

Principales contributions

Les contributions principales du chapitre 5 entourent l'utilisation de la propriété d'association négative pour obtenir un théorème central limite pour différents algorithmes d'échantillonnage.

Une contribution originale est la condition sur les probabilités d'inclusion simple et doubles pour obtenir la convergence. Dans leur utilisation en séries temporelles,

[4] Cadre de Vie et Sécurité
[5] Enquête Nationale sur les Ressources des Jeunes

il est courant d'exiger pour les observations une condition de stationnarité faible. On montre que cette condition est trop forte pour le cas du sondage, même dans le cas du sondage aléatoire simple. Les conditions que l'on donne reposent donc sur un théorème issu de [Pat+01] qui donne une condition plus faible pour obtenir un théorème central limite.

Ensuite, on expose un argument pour appliquer les résultats précédents aux sondages à plusieurs degrés (paragraphe 5.2.4). L'argument repose sur l'indépendance des tirages à chaque degré et dans chaque unité principale.

Théorème. *Si tous les plans d'échantillonnage utilisés pour tirer les unités à tous les degrés sont négativement associés (NA) et indépendants deux par deux, alors le plan d'échantillonnage à plusieurs degrés est également négativement associé (NA).*

Part III - Sampling big networks

Chapitre 6

Dans ce chapitre, on introduit les définitions des statistiques reliées aux graphes. On présente également une revue bibliographique d'études statistiques sur des populations de graphes.

Les méthodes par sondage ont ete utilisées pour de l'inférence sur des structures de graphes depuis des décennies ([Col58], [Goo61]). Parmi les grand succès de ces méthodes, de nombreuses études sur la consommation de drogues et la propagation de maladies contagieuses ([WB89], [Hec+99]), ou sur les structures de réseaux sociaux ([HJ01]).

Il faut toutefois remarquer que le cadre théorique de ces études est différent de celui qu'on peut adopter pour l'étude de populations de graphes dans les applications sur des données d'applications web ou de réseaux sociaux. En effet, dans les études par sondage menées pour les sciences sociales, l'observation des caractéristiques locales du réseaux (c'est-à-dire au voisinage de chaque nœud), n'est possible que partiellement.

Par exemple, la méthode du sondage déterminé selon les répondants ([Lav12]) consiste à sélectionner des individus selon un plan de sondage ne nécessitant pas d'information auxiliaire (sondage aléatoire simple, par exemple), et de demander à chaque répondant de cette première phase de sondage d'indiquer d'autres individus présentant la caractéristique d'intérêt de l'enquête au sein de son réseau de connaissances (des coupons sont utilisés comme incitatif). L'efficacité de la méthode vient du

fait que les meilleurs individus pour recruter des membres d'une population "cachée" (utilisateurs de drogues, par exemple) sont leurs propres pairs. Ainsi, l'observation des liens entre les individus du réseau (ou du sous-réseau d'individus portant le caractère d'intérêt) est déterminée par les indications des répondants eux-mêmes, et n'est donc que partielle, et potentiellement biaisée (observabilité partielle).

Dans le cadre de bases de données de réseaux informatisées, il est facile d'éviter cet écueil. En effet, pour chaque unité échantillonnée, il est possible de collecter l'ensemble de la topologie locale à moindre coût (observabilité totale). Ceci améliore beaucoup la qualité des estimations par rapport à la situation d'observabilité partielle. On montre que les deux concepts amènent à utiliser des estimateurs différents.

Enfin, on décrit brièvement quelques modèles probabilistes de graphes population qui seront utiles pour la suite. Le modèle de graphes aléatoires le plus étudié est celui d'Erdős-Rényi ([ER59]). Il ne permet malheureusement pas de répliquer les propriétés essentielles observées dans les réseaux sociaux (distributions leptokurtiques de degrés, densités élevées). On présente donc d'autres modèles aléatoires qui tentent de répliquer ces propriétés. En particulier, on retiendra le modèle Forest-Fire ([Les+05]) pour effectuer des simulations.

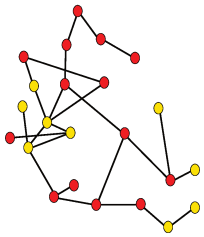
Principales contributions

Dans le chapitre 6, on introduit le concept d'observabilités partielle et totale du graphe lorsqu'il est parcouru pour être échantillonné. Cela permet de différencier les applications où on peut identifier tous les sommets voisins d'un sommet échantillonné (cadre des bases de données de graphes dans l'industrie) de celles où ce n'est pas possible (classique en statistique officielle et en sciences sociales).

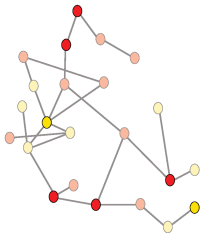
Chapitre 7

Dans ce chapitre, on définit plus précisément deux plans de sondages à extension pour les graphes : le sondage adaptatif et le sondage snowball (ou boule de neige, exemple en figure 1.4).

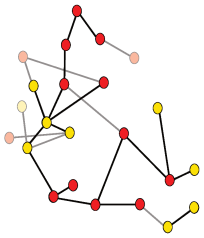
Ces deux plans de sondage pour les graphes sont dits "à extension" (ou "par chaînage") car ils consistent tous deux à tirer un échantillon de première phase (qui n'utilise pas d'information sur la structure locale du graphe), puis d'ajouter à l'échantillon des unités liées d'une certaine façon aux unités observées en première phase. Pour le sondage snowball, les unités ajoutées sont tous les nœuds qui sont liés par une arête aux unités de première phase. Pour le sondage adaptatif, on



(a) Exemple de graphe population



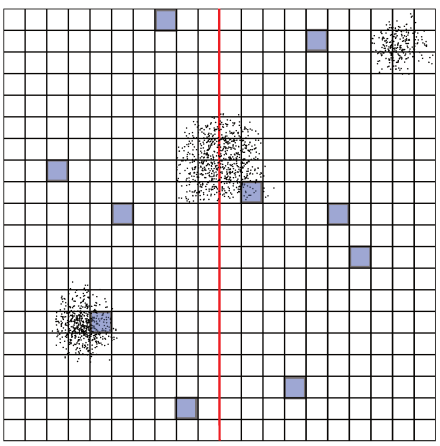
(b) Échantillon initial de noeuds



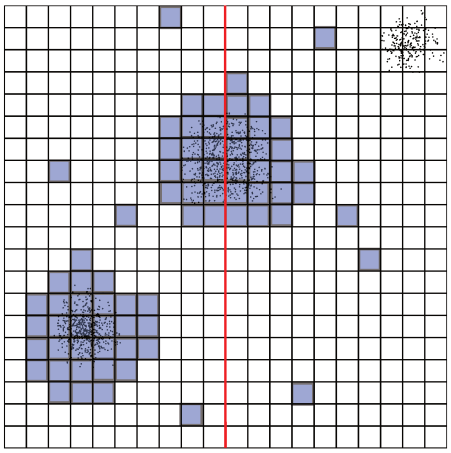
(c) Échantillon final, avec ajout de l'extension snowball

Fig. 1.4: Illustration de l'échantillonnage boule de neige (snowball sampling)

définit une sous-population d'intérêt, et on sélectionne l'intégralité des sous-réseaux auxquels les unités échantillonnée en première phase appartiennent. Le principe du sondage adaptatif est expliqué en figure 1.5.



(a) Première phase de sondage pour l'échantillonnage adaptatif



(b) Phase d'extension pour le sondage adaptatif

Fig. 1.5: Illustration d'un sondage adaptatif. La population est représentée sur un carré de 20×20 et les sous-réseaux sont marqués par les points noirs. Dans la première phase de sondage (a), 10 carreaux sont sélectionnés (marqués en bleu ciel). Dans la phase d'extension (b), on ajoute à l'échantillon la totalité des sous-réseaux qui peuvent être observés à partir de la première phase. Dans cette illustration, la population est également partitionnée en deux strates, marquées par la ligne verticale.

Enfin, on discute de l'intérêt pratique de ces deux méthodes, incluant l'efficacité algorithmique, en utilisant des données du réseau social Twitter. Dans cette application, on cherche à évaluer le nombre d'utilisateurs intéressés par la campagne de promotion du film Star Wars. On compare l'efficacité du sondage snowball et du sondage adaptatif. On s'aperçoit que la densité du réseau social conduit à sélectionner des tailles d'échantillon énormes dans le cas de l'adaptatif. On obtient finalement

des estimations très précises avec le sondage snowball, mais on constate qu'une précision similaire aurait pu être obtenue avec une stratification intelligente. Cette constatation constitue la motivation pour le chapitre 8 où l'on étudie la précision relative de l'échantillonnage snowball par rapport à des plans sans extension de taille équivalente.

Principales contributions

L'application du chapitre 7 montre qu'on peut obtenir des statistiques efficaces avec un échantillon de taille très limitée, en prenant avantage de l'observabilité complète. Enfin, on discute de la complexité en terme de calcul de ces méthodes. On montre que l'échantillonnage boule de neige est efficace algorithmiquement et statistiquement, et est préférable à l'échantillonnage adaptif.

Chapitre 8

Dans ce chapitre, on essaye d'évaluer l'efficacité pratique de l'échantillonnage snowball en fonction des variables d'intérêt.

On commence par montrer que l'estimation non pondérée, utilisée dans certains logiciels, présente des biais qui ne diminuent pas avec la taille de l'échantillon.

On cherche ensuite, via une analogie avec le sondage en grappes, un critère conciliant plan de sondage et topologie des graphes afin de prédire l'efficacité sous contrainte de taille du plan de sondage à extension. On montre que ces estimations ne permettent malheureusement pas de prédire l'efficacité de l'estimation snowball pour la plupart des variables d'intérêt.

Enfin, en utilisant des simulations sur des modèles de graphe et des graphes réels, on analyse l'efficacité du snowball en comparaison d'un sondage aléatoire simple de même taille (design effect). On montre qu'il est rare que la phase d'extension soit plus rentable en termes de précision qu'un échantillon à une seule phase de même taille (et lorsqu'elle l'est, il s'agit d'estimations peu précises dans l'absolu). Il reste que le sondage snowball permet d'obtenir de bonnes précisions pour des sous-populations d'intérêt, et qu'il est efficace algorithmiquement pour obtenir une taille d'échantillon.

Principales contributions

Dans le chapitre 8, on cherche à établir des lignes directrices pour l'élaboration de plan de sondages de graphes efficaces pour les praticiens. On part pour cela d'un premier résultat établi par [Fra77]. En notant i,j deux sommets du graphe population, B_i l'ensemble des sommets ayant une arête pointant vers le sommet i et $\pi(A) = 1 - \bar{\pi}(A)$ la probabilité d'inclusion dans l'échantillon de tous les sommets d'un ensemble A ($\pi_i = \pi(\{i\})$ est donc la probabilité d'inclusion dans l'échantillon du sommet i et $\pi_{ij} = \pi(\{i, j\})$ la probabilité d'inclusion double des sommets i et j) :

Théorème (Théorème 4, [Fra77])**.** *S'il existe une paire d'arêtes distinctes i et j telles que le graphe et le plan de sondage initial vérifient :*

$$B_i = \{i\} \subset B_j \text{ et}$$
$$\bar{\pi}(B_j)\pi_{ij} \neq \pi_{ij} - \pi_i\pi_j$$

alors ni l'estimation utilisant uniquement les données du plan initial, ni l'estimation utilisant les données provenant de l'échantillonnage snowball ne domine l'autre.

Cette condition suffisante d'indécidabilité montre qu'il est difficile d'évaluer l'efficacité de l'échantillonnage à extension dans le cas d'observabilité complète. Elle montre également que l'efficacité dépend de la topologie du graphe.

On montre ensuite qu'une analogie avec l'échantillonnage de grappes ne suffit en général pas à décider de l'efficacité de l'échantillonnage snowball, et ce pour les variables d'intérêt de graphe les plus populaires.

Enfin, en utilisant des simulations sur des modèles de graphe (Forest-Fire, Barabási-Albert) reproduisant des propriétés des graphes observés en pratiques, puis sur des exemples réels, on discute de l'efficacité en termes de précision de l'échantillonnage par extension. Intrinsèquement, l'échantillonnage snowball ne semble pas permettre un gain de précision pour la grande majorité des cas étudiés. Il reste toutefois une bonne option d'un point de vue algorithmique, ou pour obtenir des statistiques sous un sous-domaine.

Finalement, on suggère que le praticien qui souhaite utiliser l'échantillonnage par extension pour son analyse a tout intérêt à intégrer dans son processus de production une analyse de son propre graphe, qu'il conviendra de réajuster à intervalle réguliers.

Part I

Échantillonnage avec objectifs multiples

2 Optimisation d'une répartition mixte

Cet article, co-écrit avec Thomas Merly-Alpa, a été accepté pour publication dans la revue Survey Methodology

—

Cet article propose un critère de calcul du paramètre de compromis dans les répartitions dites "mixtes", c'est-à-dire qui mélangent deux répartitions classiques en théorie des sondages. Dans les enquêtes auprès des entreprises de l'Insee (*Institut National de la Statistique et des Études Économiques*), il est courant d'utiliser la moyenne arithmétique d'une répartition proportionnelle et d'une répartition de Neyman (correspondant à un paramètre de compromis de 0.5). Il est possible d'obtenir une valeur du paramètre de compromis aboutissant à de meilleures propriétés pour les estimateurs. Cette valeur appartient à une région qui est obtenue en résolvant un programme d'optimisation. Différentes méthodes de calcul de ce paramètre seront présentées. Une application sur des enquêtes auprès des entreprises est présentée, avec comparaison avec d'autres répartitions de compromis usuelles.

2.1 Introduction

Dans cet article, on se place dans un cadre qui réplique une partie des enquêtes réalisées en statistique officielle, notamment les enquêtes auprès des entreprises, dont le plan de sondage est le plus souvent un sondage aléatoire simple stratifié à un degré. Un plan de sondage est créé pour estimer le total $T(y)$ ou la moyenne $\bar{Y}$ d'une variable d'intérêt principale continue y, où y_k désigne la valeur de y pour le k-ième individu de la population. Les données d'enquêtes sont également utilisées pour estimer une collection d'autres variables, qui sont parfois décorrélées ou anti-corrélées de y.

Lorsqu'un plan stratifié est utilisé, le choix d'une répartition permet généralement de répondre à un but précis, selon la règle classique "un objectif, un échantillon".

Ainsi, dans le but d'estimer avec une précision maximale le total $T(y)$ de la variable d'intérêt, on peut utiliser la répartition de Neyman [Ney34]. Les répartitions spécifiques répondent à un besoin précis relatif à y. Dans le cas où les données d'enquêtes sont utilisées pour l'estimation de quantités issues d'autres variables, il est souhaitable que le plan de sondage utilisé ne détériore pas la qualité des estimateurs. Cochran [Coc63] et Chatterjee [Cha67] proposent par exemple une répartition spécifique pour une collection de variables d'intérêt. Cela ne règle toutefois pas le cas de variables qui ne peuvent être incluses dans la création du plan de sondage.

Or, si une variable est décorélée ou anti-corrélée aux variables utilisées pour le calcul d'une répartition spécifique, il est connu que la variance de l'estimation de son total peut être très forte (voir par exemple [Ard06]). L'utilisation d'une répartition proportionnelle, même lorsque l'on dispose d'information auxiliaire, peut ainsi être avantageuse. Elle permet d'être "agnostique" et d'éviter de construire un plan qui sera délétère pour l'estimation de certaines variables, de certains paramètres autres que des totaux ou des moyennes ou pour l'estimation sur des domaines spécifiques. On peut également se référer à Chiodini et al. [Chi+10b] et Chiodini et al. [Chi+10a] pour une discussion plus étendue sur l'intérêt de la répartition proportionnelle dans le compromis.

Nous nous intéressons à un certain type de répartitions pour des sondages stratifiés avec H strates (de tailles respectives N_h dont la somme vaut la taille de la population N) de taille fixe n. Le choix d'une répartition consiste à déterminer un vecteur $\mathbf{n} = (n_1, \ldots, n_H)$ vérifiant la contrainte $\sum_{h=1}^{H} n_h = n$. Nous étudions en particulier une répartition "mixte", qui consiste en un compromis entre la répartition proportionnelle et une autre répartition spécifique, répondant à un besoin précis sur une ou plusieurs variables d'intérêt d'une enquête:

$$\mathbf{n}_\alpha = \alpha \mathbf{n}_{prop} + (1 - \alpha)\mathbf{n}_{specifique} \tag{2.1}$$

où $\mathbf{n}_\alpha$, $\mathbf{n}_{prop}$ et $\mathbf{n}_{specifique}$ sont des vecteurs de taille H et $\alpha \in [0, 1]$. Cette répartition de compromis correspond à la méthode ROAUST ([Chi+17]) lorsque la répartition spécifique choisie est celle de Neyman [Ney34]. La répartition proportionnelle est définie par :

$$n_h = n\frac{N_h}{N},\ h = 1, \ldots, H$$

Le but de cet article est de proposer une méthode pour déterminer α. On cherche ainsi à calculer un paramètre satisfaisant un certain critère d'optimalité que nous détaillerons dans la section 2.3. Nous n'aborderons pas dans cet article la constitution des strates, sujet qui a été très largement exploré dans la littérature, par exemple dans Baillargeon et al. [Bai+07] and Dalenius and Hodges Jr [DHJ59]. D'autre part, nous ne cherchons pas ici à prendre en compte les phénomènes de non-réponse.

Remarquons que la répartition proportionnelle est celle qui minimise la dispersion des poids. Choisir une répartition proportionnelle revient ainsi à la logique plus générale de choisir un plan qui minimise la dispersion des poids de sondage. Ainsi, le plan de sondage de l'échantillon-maître de l'Insee a été conçu avec cet objectif ([CF09]), dans une logique fondée sur le plan de sondage. Dans une logique fondée sur le modèle, si l'on cherche à estimer des paramètres (coefficients de la droite de régression) et que le plan de sondage est non informatif, alors des poids de sondage constants minimisent la variance de l'estimation (voir [DD09] et [Sol+15]).

La répartition de compromis consiste ainsi à concilier deux objectifs opposés : créer un plan de sondage efficace pour une variable d'intérêt, tout en maintenant les poids les plus proches possibles afin de ne pas déteriorer l'estimation sur des variables très diverses. Dans la suite, nous formalisons le programme d'optimisation qui correspond à ces contraintes. Nous donnons un théorème qui définira le critère d'optimalité que l'on cherche à résoudre. Enfin, nous analysons les performances de la méthode de détermination de répartition que nous proposons en comparaison avec quelques autres méthodes existantes dans la littérature sur un cas pratique, celui d'une enquête auprès des entreprises réalisée par l'Insee (Institut National de la Statistique et des Études Économiques - France).

Plusieurs répartitions connues permettent déjà d'effectuer des compromis entre plusieurs objectifs. Une répartition fréquemment utilisée à l'Insee est une répartition de Neyman sous contraintes de précision locale, présentée dans Koubi and Mathern [KM09]. Une répartition mieux connue dans la littérature est la répartition puissance de Bankier [Ban88] qui permet de faire un compromis entre la répartition de Neyman et une répartition permettant un coefficient de variation uniforme de l'estimation du total d'une variable d'intérêt sur chacune des strates. Cette répartition s'écrit de la façon suivante :

$$n_h = n\frac{S_h(T_h(x))^q/\bar{Y}_h}{\sum_h S_h(T_h(x))^q/\bar{Y}_h}, \quad h = 1, \ldots, H$$

où q est un paramètre dans $[0\,;1]$, $T_h(x)$ est une mesure de la taille ou de l'importance de la strate h (par exemple, la taille de la strate ou l'importance économique de celle-ci), S_h^2 est la variance empirique de y dans la strate h et $\bar{Y}_h$ sa moyenne.

Dans l'expression de la répartition de Bankier, q est un paramètre qui, comme le paramètre α de la répartition que nous proposons, réalise l'arbitrage entre les deux objectifs contraires de la répartition : quand q est proche de 1, la répartition est très proche d'une répartition de Neyman, tandis que, quand q tend vers 0, la répartition s'approche d'une répartition garantissant des coefficients de variation égaux dans toutes les strates. Cependant, l'article de Bankier [Ban88] ne propose pas de méthode pour le choix de ce paramètre ; c'est une telle méthode que nous allons présenter dans cet article pour notre famille de répartitions mixtes.

Dans cet article, nous proposons de réaliser ce compromis en résolvant le programme suivant :

$$\min_{\mathbf{n}=(n_h)_{h\in[\![1,H]\!]}} \quad \sum_{h=1}^{H} n_h \left(\frac{N_h}{n_h} - \frac{N}{n}\right)^2 + \lambda \left\|\mathbf{n} - \mathbf{n}_{specifique}\right\|_p \tag{2.2}$$

avec $\lambda \in [0, +\infty[$, $p \geq 1$ et $\|.\|_p$ désignant la norme p d'un vecteur de taille H (dans cette équation le terme de droite représente donc une distance entre la répartition de compromis et la répartition spécifique choisie pour l'enquête). On remarque également que N/n est le poids moyen pour les unités échantillonnées. Comme dans le cadre d'un plan stratifié, le poids de sondage pour une unité de la strate h est N_h/n_h, le premier terme du programme d'optimisation correspond donc à l'écart quadratique moyen du vecteur de poids, autrement dit à la dispersion des poids. Ce programme correspond donc à un compromis entre les deux objectifs souhaités. On verra en partie 2.3 que l'intérêt de la méthode consiste en le choix d'une valeur de λ adaptée ; ce choix est déterminant pour trouver l'équilibre le plus pertinent entre les deux objectifs contraires que nous visons avec la répartition, i.e. l'optimalité pour certaines variables apportée par la répartition spécifique d'une part et l'équipondération d'autre part.

Le programme d'optimisation utilisé dans le cadre de ce papier est inspiré de celui utilisé dans l'algorithme CURIOS (*Curios Uses Representativity Indicators to Optimize Samples*, [MAR]) qui réalise un arbitrage afin de mettre en place une opération de priorisation de la collecte d'enquêtes en face-à-face, en déterminant une répartition de deuxième vague. Dans ce papier, nous ne considérons que le problème de détermination des répartitions *ex ante*, et n'utilisons donc pas l'algorithme dans le contexte de son introduction.

2.2 Programme d'optimisation

Le programme (2.2) est difficile à résoudre et à analyser. C'est pourquoi on va se contenter ici de chercher une solution sur un segment entre la répartition proportionnelle et une répartition spécifique donnée, la répartition de Neyman, qui est la plus fréquemment utilisée. Souvent, on considère que le choix d'un $\alpha = 1/2$ constitue un bon compromis. C'est ce qui est par exemple proposé dans Chiodini et al. [Chi+10b], ou dans certains plans d'enquêtes auprès des entreprises de l'Insee.

Cette méthode permet de combiner les bénéfices des deux méthodes à faible coût. Cependant, on peut s'interroger sur le choix arbitraire du facteur 1/2. Dans ce paragraphe, nous allons présenter une méthode basée sur un programme de minimisation faisant intervenir la dispersion des poids ainsi que la distance à la répartition de Neyman pour choisir un paramètre α tel que la répartition mixte « optimale » entre répartition proportionnelle et répartition de Neyman soit :

$$\mathbf{n}_\alpha^{opt} = \alpha \mathbf{n}_{prop} + (1-\alpha)\mathbf{n}_{Neyman} \tag{2.3}$$

Nous nous plaçons ici dans le cadre d'un tirage stratifié à H strates, en négligeant l'influence de la non-réponse. Celle-ci pourrait être intégrée via la prise en compte de taux de réponse anticipés ou d'une seconde phase poissonnienne, mais cela complique inutilement la forme des résultats obtenus. On s'intéresse ici à un jeu de répartitions $(\mathbf{n}_\alpha)$ qui parcourent un segment entre la répartition proportionnelle $(\mathbf{n}_{prop})$ et la répartition de Neyman $(\mathbf{n}_{Neyman})$, comme indiqué dans l'équation (2.3). On se limite donc à la réalisation du programme de minimisation suivant, forme simplifiée de celui de l'équation (2.2) :

$$\min_{\alpha\in[0,1]} \quad \sum_{h=1}^{H} n_{\alpha,h}\left(\frac{N_h}{n_{\alpha,h}} - \frac{N}{n}\right)^2 + \lambda\alpha \tag{2.4}$$

Le terme de droite correspond à la distance entre la répartition cherchée et la répartition de Neyman, à une constante près, intégrée dans le λ : ce résultat est démontré en Annexe 1.

Ce programme de minimisation dépend de la constante $\lambda \geq 0$ choisie. On remarque aisément que lorsque λ est suffisamment grand, le terme de distance devient prépondérant et on obtient $\alpha = 0$ et donc $\mathbf{n}_\alpha = \mathbf{n}_{Neyman}$. De même, quand

λ tend vers 0, le facteur représentant la dispersion des poids devient prépondérant et la répartition tend vers la répartition proportionnelle.

2.3 Choix du λ

Comme évoqué en partie 2.1, il faut choisir une valeur adaptée de λ, qui représente l'importance que l'on veut accorder à chacun des termes du compromis. On verra dans cette partie que le choix de cette valeur est crucial pour obtenir un bon paramètre de compromis. Pour cela, nous allons nous intéresser à la variance de l'estimateur de Horvitz-Thompson du total d'une variable d'intérêt de l'enquête obtenue avec une répartition donnée quand le plan de sondage appliqué dans chacune des strates est un sondage aléatoire simple.

En effet, l'idée est d'utiliser une propriété clef de la répartition de Neyman, qui est son caractère plat (voir par exemple [Ard06]). Cela signifie qu'au voisinage de la répartition, la variance de l'estimateur du total de la variable d'intérêt de l'enquête est proche de sa valeur minimale, ce qui est satisfaisant tant d'un point de vue théorique qu'empirique. La question consiste à bien définir ce voisinage : si nous arrivons à choisir une valeur λ permettant d'obtenir une répartition suffisamment proche de la répartition proportionnelle tout en appartenant à la zone plate autour de la répartition de Neyman, nous aurons ainsi réussi à obtenir des poids garantissant une estimation d'une précision proche de l'optimum pour la variable d'intérêt centrale de l'enquête avec une dispersion des poids minimale.

Le choix de λ et le choix de α sont en fait interchangeables. En effet, pour une valeur de λ fixée, on peut résoudre le programme d'optimisation de l'équation (2.4) et obtenir une valeur $\alpha(\lambda)$, et donc une répartition $\mathbf{n}_{\alpha(\lambda)}$. À l'inverse, choisir une valeur de α directement permet de favoriser un des deux aspects du programme d'optimisation (distance à la répartition de Neyman ou équipondération), de façon similaire au choix du λ. Le choix de conserver le paramètre λ permet de conserver un cadre d'application plus large.

Nous allons nous intéresser à la variance de l'estimateur obtenue lorsque λ varie. À partir de la répartition $\mathbf{n}_{\alpha(\lambda)}$, il est possible d'étudier la variance de l'estimateur d'Horvitz-Thompson du total d'une variable d'intérêt $\hat{T}_{HT}(y) = \sum_{i \in s} \frac{y_i}{\pi_i}$, avec π_i la probabilité d'inclusion dans l'échantillon de l'unité i (c'est-à-dire égale à n_h/N_h si h est la strate qui contient i). On montre alors qu'il existe au voisinage de l'optimum de précision (c'est à dire la répartition de Neyman, obtenue quand $\lambda \to +\infty$) une région "plate". Ainsi, choisir un λ sur cette région plate assure que la précision n'est

que peu déteriorée par rapport à l'optimum, tout en réduisant nettement la variance des poids de sondage. Mathématiquement, il s'agit de choisir comme λ le point de torsion de la courbe, dont l'existence est assurée par le théorème suivant :

Theorem 2. *Soit $V(\lambda)$ la fonction de variance de $\hat{T}_{HT}(y)$ pour la répartition obtenue pour le α solution du programme de minimisation de l'équation (2.4) pour un tel λ. Alors, il existe un segment $S \subset [0, +\infty[$ tel que :*

- *$\alpha(S) = [0, 1]$, où $\alpha(\lambda)$ associe à λ la solution du programme 2.4.*
- *$V(\lambda)$ est décroissante sur S.*
- *Sa dérivée seconde admet un maximum dans S qu'on appelle **point de torsion**.*

La démonstration de ce théorème est en Annexe 2.

On veut donc prendre λ au point de torsion de la courbe, qui est aussi un point d'inflexion de sa dérivée ; cela revient à situer le « coude » de la courbe, c'est à dire à être juste à la limite du plateau de variance dû à la proximité de la répartition de Neyman, lié au caractère plat de l'optimum. L'intuition qui justifie ce choix est que d'une part la variance de l'estimateur du total de la variable d'intérêt principale utilisée pour calculer la répartition de Neyman diminue quand λ augmente, car la répartition mixte s'approche alors de la répartition de Neyman, et que d'autre part, au delà d'un certain seuil, cette variance varie peu et est très proche de sa limite, la variance obtenue avec la répartition de Neyman. Ce seuil correspond, intuitivement, au moment où la variance cesse de diminuer sensiblement quand λ augmente. Ce point, dont le théorème permet de prouver l'existence, est ainsi bien identifié via l'analyse des variations des évolutions de la variance avec λ, i.e. en étudiant la dérivée seconde de la variance en tant que fonction de λ et le point où cette dérivée atteint un maximum, les dérivées de la variance étant négatives. Par ailleurs, le fait de se placer à la bordure du plateau permet de limiter au maximum la valeur de λ et donc la dispersion des poids. La Figure 2.1 illustre ce choix.

Dans les cas les plus simples, c'est à dire dès lors que toute l'information est disponible, et que le tirage s'effectue à un seul degré, sans prendre en compte d'autres paramètres, la solution la plus simple pour déterminer le λ consiste à étudier analytiquement la courbe de la Figure 2.1 en utilisant les formules classiques de calcul de variance de $\hat{T}_{HT}(y)$ dans un plan de sondage stratifié. Le point de torsion est obtenu en cherchant le maximum de la dérivée seconde de la courbe $V(\lambda)$. Cette dérivée est en général difficile à calculer analytiquement, mais il est tout

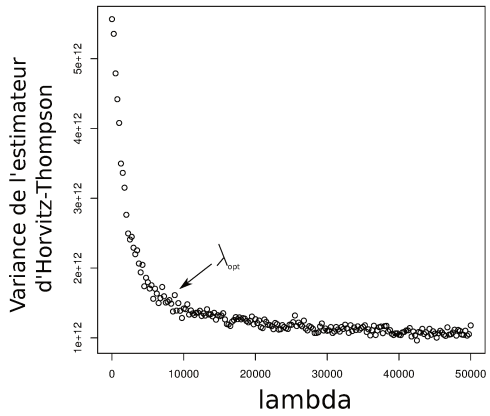


Fig. 2.1: Exemple d'un point de torsion de la fonction $V(\lambda)$ pour un plan de sondage explicité dans Merly-Alpa and Rebecq [MAR]

à fait possible de trouver un maximum numérique lorsque que l'on dispose d'une formule analytique (ou à défaut, une courbe suffisamment lisse) pour $V(\lambda)$.

Il n'est malheureusement pas toujours possible de calculer analytiquement la variance, par exemple quand d'autres contraintes (regroupement de strates, etc.) rentrent en jeu, ou si toute l'information n'est pas disponible au stade de l'échantillonnage. Dans ce cas, on remplace la courbe $V(\lambda)$ par une version estimée par méthode de Monte-Carlo :

1. On choisit λ dans $[0, 1]$.

2. On utilise les données disponibles pour simuler la variance de $\hat{T}_{HT}(y)$. Pour cela, on calcule la répartition issue de l'équation (2.4) pour λ et on effectue K tirages indépendants d'échantillons suivant ce plan de sondage. Pour chacun des $k = 1, \ldots, K$ tirages effectués, on calcule $\hat{T}^{(k)}_{HT}(y)$, l'estimateur d'Horvitz-Thompson de $T(y)$ obtenu avec les données de l'échantillon k. Ensuite, on calcule la quantité :

$$V_{MC}(\lambda) = \frac{1}{K-1} \sum_{k=1}^{K} \left(\hat{T}^{(k)}_{HT}(y) - \frac{1}{K} \sum_{j=1}^{K} \hat{T}^{(j)}_{HT}(y) \right)^2$$

   Cette quantité est un estimateur Monte-Carlo de $V(\lambda)$.
   Notons que ces simulations nécessitent une variable proxy de la variable d'intérêt disponible pour toutes les unités de la population. Dans le cas d'enquêtes entreprises, le chiffre d'affaire disponible dans les bases fiscales peut être un bonne variable de substitution pour le chiffre d'affaires réel.

3. On recomence pour d'autres valeurs de λ parcourant $[0, 1]$ avec un certain pas η. Les valeurs de η et de K doivent être choisies en prenant en compte le temps de calcul, qui peut être assez long selon la population d'origine, mais également afin de s'assurer que la variance due aux simulations n'est pas trop importante, ce qui invaliderait les résultats obtenus.

4. Une fois ces résultats obtenus pour différentes valeurs de λ, on trace la courbe de $V_{MC}(\lambda)$, que l'on espère suffisamment lisse. On peut alors afficher la courbe et placer visuellement le coude, ce qui permet de choisir la valeur de λ_{MC} finale. Une autre possibilité est de rechercher le maximum de la dérivée seconde de $V_{MC}(\lambda)$ à l'aide d'un algorithme d'optimisation suffisamment robuste au bruit. Par exemple, l'algorithme de Nelder and Mead [NM65] est implémenté dans la grande majorité des logiciels d'optimisation (en R ou en Python par exemple), et Rebecq and Merly-Alpa [RMA15] montrent qu'il donne des bons résultats pratiques pour ce type de problème.

Dans tous les cas, si la détermination du λ au coude est difficile, il convient de choisir une valeur qui assure que l'on se situe à droite du coude réel sur la courbe. Cette méthode conservative garantit que l'on se trouve sur la région plate de la courbe, et que la précision de l'estimateur de la variable d'intérêt n'est pas déteriorée.

2.4 Application pratique

On s'intéresse au tirage d'un échantillon de 1 000 entreprises de l'industrie selon différents plans de sondages stratifiés afin de connaître le chiffre d'affaires total du secteur. Le champ exact est défini comme suit :

- Entreprises actives situées en France.

- Entreprises dont l'effectif est compris entre 1 et 100.

- Entreprises dont le secteur d'activité, mesuré grâce au code d'activité principale exercée (APE) appartient à l'une des divisions de l'industrie dans la Nomenclature d'activité des communautées européennes (Nace, dont les divisions sont identiques aux 88 divisions de la Classification Internationale Type par Industrie de toutes les activités économiques appelée CITI ou ISIC en anglais), i.e. aux divisions de 10 à 33, hormis la 12 (Produits à base de tabac) et la 19 (Produits de la cokéfaction et du raffinage), qui ont une structure trop atypique pour notre étude.

La population initiale est de 102 172 entreprises. De manière générale, les entreprises ayant un fort effectif, par exemple plus de 100, sont souvent enquêtées exhaustivement. On se limite ici à la partie non exhaustive d'une enquête.

Cette population est stratifiée selon deux critères :

1. L'APE, au niveau division (deux premiers chiffres).

2. La tranche d'effectif, de la façon suivante : 1 à 9 salariés ; 10 à 19 salariés ; 20 à 49 salariés ; 50 salariés ou plus.

ce qui constitue 88 strates, qu'on notera par la suite (A, B) où A est le secteur d'activité et B l'effectif.

On calcule alors les répartitions proportionnelle et de Neyman relative à la dispersion du chiffre d'affaires dans chacune de ces strates, pour $n = 1\ 000$. Le tableau 2.1 résume les caractéristiques de ces deux répartitions, ainsi que les strates où la répartition est maximale, toutes deux dans la division 10 (Industries alimentaires).

Répartition	Min	Médiane	Max
Proportionnelle	1	3	278
Neyman	1	5	162

Strate	Rep. Proportionnelle	Rep. Neyman
(10, 1-9)	**278**	80
(10, 20-49)	18	**162**

Tab. 2.1: Distribution des tailles d'échantillon par strate pour les deux répartitions, et tailles d'échantillon pour les strates correspondant aux tailles d'échantillon maximales.

On souhaite choisir la répartition mixte optimale pour le problème présenté au paragraphe précédent. On choisit comme fonction de distance la distance euclidienne. L'équation 2.4 devient donc :

$$\min_{\alpha\in[0,1]} \quad \sum_{h=1}^{H} n_{\alpha,h}\left(\frac{N_h}{n_{\alpha,h}} - \frac{N}{n}\right)^2 + \lambda\sqrt{\sum_{h=1}^{H}(n_{\alpha,h} - n_{Neyman,h})^2} \tag{2.5}$$

Nous appliquons ensuite la méthode suivante pour calculer la répartition optimale :

- Calculer pour différentes valeurs de λ la valeur de α solution du programme de minimisation de l'équation (2.5).

- Pour chaque α, calculer la répartition correspondante.

- Pour chacune des répartitions, calculer analytiquement la variance de l'estimateur d'Horvitz-Thompson du total du chiffre d'affaires. Cela est possible car on dispose du chiffre d'affaires des entreprises dans le répertoire qui sert de base de sondage.

On obtient finalement la courbe représentée en Figure 2.2. On remarque que sa forme correspond globalement à ce qui était attendu en appliquant le Théorème 2. On détermine visuellement le point de torsion, qui semble situé vers $1 \cdot 10^7$. On pose donc $\lambda_{coude} = 1 \cdot 10^7$, qui se situe légèrement à droite du coude, sur la partie plate de la courbe $V(\lambda)$.

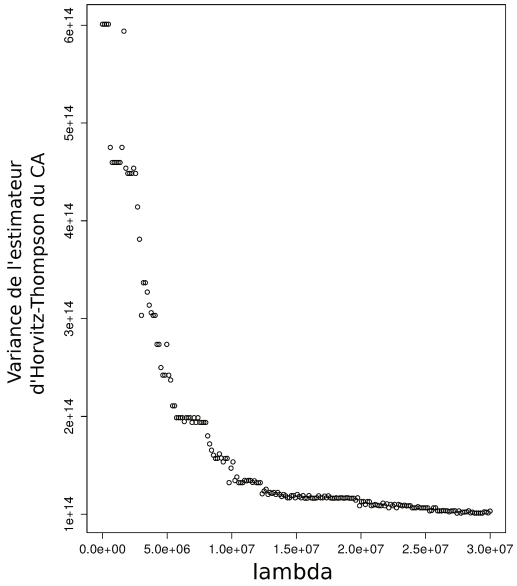


Fig. 2.2: Variance de l'estimateur d'Horvitz-Thompson du total du chiffre d'affaire dans le cadre d'un compromis avec la répartition de Neyman.

On peut alors utiliser la valeur de λ_{coude} pour déterminer α_{coude}, à l'aide du programme d'optimisation de l'équation (2.5). Nous obtenons ici $\alpha_{coude} = 0.644$. Cette valeur de α obtenue peut être interprétée directement. Elle est assez proche de 0.5, ce qui montre que la répartition finale est également assez proche de la répartition qu'on appelle classiquement mixte, mais elle est supérieure à 0.5, ce qui montre que l'optimum du programme se rapproche sensiblement de la répartition proportionnelle. La répartition obtenue est décrite dans le tableau 2.2, et comparée

à la répartition mixte usuelle utilisant la moyenne arithmétique entre les deux répartitions initiales.

Répartition	Min	Médiane	Max
Proportionnelle	1	3	278
Coude	1	4	208
Mixte	1	4	179
Neyman	1	3	162

Répartition	Proportionnelle	Coude	Mixte	Neyman
α	*1*	*0.644*	*0.5*	*0*
Strate (10, 1-9)	278	208	179	80
Strate (10, 20-49)	18	69	90	162

Tab. 2.2: Distribution des tailles d'échantillon par strate pour la répartition obtenue, ainsi que pour les deux strates correspondant aux tailles d'échantillon maximales pour la répartition de Neyman et la répartition proportionnelle.

En termes de tailles d'échantillons dans les strates pour les diverses répartitions, on peut constater que l'on obtient un maximum pour la même strate que la répartition proportionnelle (10,1-9), mais avec une distribution moins étendue. D'autre part, la strate (10,20-49) qui a l'effectif le plus important dans la répartition de Neyman, voit effectivement sa taille augmenter par rapport à la répartition proportionnelle, mais reste toutefois bien inférieure à la répartition de Neyman. On voit bien l'apparition d'un compromis entre les répartitions, comme dans le cas de la répartition mixte usuelle.

Il reste cependant à s'intéresser aux deux critères qui motivent cette analyse, c'est à dire d'une part l'écart-type de l'estimateur d'Horvitz-Thompson du total du chiffre d'affaires (en milliards d'euros), et d'autre part la dispersion des poids et son influence sur la précision des estimateurs liés à d'autres concepts : pour l'évaluer, nous introduisons une variable z non corrélée au chiffre d'affaires. Nous choisissons ici la variable z liée à l'implantation géographique de l'entreprise définie de la façon suivante :

$$z_i = \begin{cases} 1 & \text{si l'entreprise } i \text{ est située en Ile-de-France} \\ 0 & \text{sinon.} \end{cases}$$

Nous allons comparer sur ces trois critères notre méthode avec les répartitions initiales (proportionnelle, Neyman), mais également avec la répartition mixte classique (avec un facteur 0.5), avec des répartitions puissance de Bankier [Ban88] pour différentes valeurs de q (où $T_h(x)$ est pris égal à la somme du chiffre d'affaires dans la strate h) et avec la répartition de Neyman sous contraintes de précision

locale de Koubi and Mathern [KM09]. Les résultats obtenus sont exposés dans le tableau 2.3. Dans ce tableau, $\hat{T}_{HT}(CA)$ désigne l'estimateur d'Horvitz-Thompson du chiffre d'affaires, et $\hat{T}_{HT}(z)$ l'estimateur d'Horvitz-Thompson de la variable z.

Répartition	Paramètre	Écart-type de $\hat{T}_{HT}(CA)$	Dispersion des poids	Écart-type de $\hat{T}_{HT}(z)$
Proportionnelle	$\alpha = 1$	24.7	47	10.7
Coude	*0.644*	12.5	1929	11.6
Mixte	*0.5*	11.4	3473	12.3
Neyman	*0*	9.8	18585	17.9
Bankier	$q = 0.25$	13.1	36250	22.2
	0.5	11.2	25922	19.7
	0.75	10.1	20187	18.2
Koubi-Mathern	·	12	35680	22.7

Tab. 2.3: Dispersion des poids et variance des estimateurs du chiffre d'affaires et de z pour plusieurs répartitions.

On remarque ici que la répartition obtenue à l'aide de λ_{coude} a une précision pour l'estimation du chiffre d'affaires total assez proche de la répartition de Neyman, alors que la répartition proportionnelle entraîne un écart-type de l'estimateur de Horvitz-Thompson du total de chiffre d'affaires bien plus grand. Or, cette légère perte de précision est très largement contrebalancée par le gain en dispersion des poids par rapport à la répartition de Neyman et par un gain important en termes de précision sur le total de la variable géographique z. Notons que la dispersion des poids n'est pas nulle dans le cadre de la répartition proportionnelle à cause des arrondis. Lorsque l'on compare la répartition obtenue à la stratégie "mixte" utilisant le facteur $\alpha = 1/2$, on remarque que la perte d'un facteur 1.1 en précision du total de chiffre d'affaires est compensée par le gain d'un facteur 1.8 en dispersion des poids et de 1.1 sur la précision du nombre total d'entreprises situées en Ile-de-France. La répartition finale satisfait bien à nos contraintes, et répond à notre demande : avoir une bonne précision et une faible dispersion des poids.

La comparaison avec les méthodes de la littérature permet d'illustrer l'apport du compromis sur la dispersion des poids. Pour les répartitions puissance, on constate qu'en choisissant des valeurs de q élevées, correspondant à des répartitions proches de celles de Neyman, on obtient une précision meilleure pour l'estimation du chiffre d'affaires total que pour notre répartition. On constate que pour la totalité des répartitions de Bankier ainsi que pour la répartition de Neyman sous contraintes, la dispersion des poids obtenue est supérieure à celle de la répartition de Neyman, et donc largement supérieure à celle de notre répartition. De façon symétrique, et comme attendu, toutes ces répartitions conduisent à dégrader la précision de l'estimation du total de la variable z.

L'objectif de ces méthodes concurrentes étant d'obtenir une meilleure précision locale, nous allons nous intéresser à plusieurs sous-domaines de notre champ (nomenclature A17 de l'économie française) :

- Domaine C1 : Fabrication de denrées alimentaires, de boissons ;
- Domaine C3 : Fabrication d'équipements électriques, électroniques, informatiques ; fabrication de machines ;
- Domaine C4 : Fabrication de matériels de transport ;
- Domaine C5 : Fabrication d'autres produits industriels.

Nous comparons alors la précision de l'estimateur du total de chiffre d'affaires pour chacun de ces secteurs. Les résultats sont compilés dans le tableau 2.4.

Répartition	Paramètre	C1	C3	C4	C5
Proportionnelle	$\alpha = 1$	0.29	0.30	0.46	0.16
Coude	*0.644*	0.16	0.20	0.35	0.07
Mixte	*0.5*	0.15	0.18	0.30	0.07
Neyman	0	0.12	0.15	0.25	0.06
Bankier	$q = 0.25$	0.21	0.13	0.18	0.07
	0.5	0.17	0.13	0.19	0.06
	0.75	0.14	0.14	0.22	0.06
Koubi-Mathern	·	0.11	0.11	0.11	0.09

Tab. 2.4: Précisions locales de l'estimateur du total de chiffre d'affaires pour plusieurs répartitions.

On constate ici que la répartition que nous proposons donne des résultats légèrement moins bons que la répartition mixte classique sur la précision locale de l'estimateur du total de chiffre d'affaires. Elle est en revanche nettement meilleure que la répartition proportionnelle, et de façon moins marquée, moins efficace que la répartition de Neyman. Notre méthode de choix de α est ainsi un compromis efficace pour réduire la dispersion des poids sans trop impacter la précision globale et locale des estimateurs.

En revanche, et comme cela était attendu, les répartitions ayant pour objectif de compromis de maximiser ou d'uniformiser la précision locale sont meilleures que la répartition proposée sur la majorité des secteurs d'activité. Ainsi, choisir entre le compromis que nous proposons et celui proposé par Bankier [Ban88] revient à choisir entre une meilleure précision pour des variables non corrélées à la variable d'intérêt y (via la dispersion des poids), comme la variable z définie ici, pour notre famille de répartitions mixtes, ou choisir une meilleure précision locale pour uniquement cette

variable y dans le cas de la répartition puissance. L'avantage de notre méthode est cependant de pouvoir proposer une valeur du paramètre de compromis α optimal sur un certain critère, ce que ne fait pas la méthode de Bankier avec le paramètre q.

2.5 Conclusion

Nous avons étudié pour les plans stratifiés une répartition de compromis située sur un segment entre la répartition proportionnelle et la répartition de Neyman. Un théorème garantit l'existence d'une région plate au voisinage de l'optimum et d'un point particulier qui donne un paramètre de compromis optimal selon un certain critère. On effectue des simulations dans le cadre d'une enquête auprès des entreprises de l'industrie qui montrent comment le calcul peut être effectué en pratique, et que le choix usuel d'un paramètre de 1/2 n'est pas toujours le plus efficace. Une comparaison avec d'autres répartitions de compromis comme la répartition classique de Bankier montre que notre objectif de dispersion des poids permet une meilleure équipondération aux dépens d'une moins bonne précision pour la variable d'intérêt sur des sous-domaines du champ. Elle illustre cependant la variabilité des résultats obtenus pour les répartitions de compromis selon la valeur du paramètre utilisé ; notre méthode de détermination du paramètre α permet ainsi de pallier ce problème souvent rencontré dans l'étude de ces familles de répartitions.

Il est possible de remplacer dans le compromis la répartition de Neyman par d'autres répartitions spécifiques *ad hoc*. On postule que la méthode reste applicable pour obtenir les mêmes propriétés désirables. Différentes applications de ces travaux ont été réalisées à l'Insee avec d'autres répartitions spécifiques. Dans le cas de l'enquête annuelle sur le coût de la main d'oeuvre et la structure des salaires (ECMOSS), la répartition spécifique utilisée pour le tirage des entreprises interrogées s'inscrit dans un plan de sondage à deux degrés où, dans chaque établissement échantillonné au premier degré, un échantillon de salariés est tiré. La répartition utilisée au premier degré est alors optimisée pour obtenir la variance d'estimation la plus faible sur l'estimation du total du salaire net sur l'échantillon final de salariés compte-tenu de la dispersion des salaires dans chaque établissement. La répartition intègre également des contraintes de précision sur certains domaines de diffusion. On obtient toutefois des courbes de la forme désirée et la répartition de compromis a pu être mise en œuvre.

Annexe 1 : Terme de distance dans l'équation (2.4)

Le choix de la distance (c'est-à-dire d'une valeur pour p) dans le second terme du programme d'optimisation (2.2) n'est pas crucial dans le contexte proposé, car nous allons pouvoir réécrire le second terme de la façon suivante, où C_p est une constante strictement positive ne dépendant que du choix de p :

$$\|\mathbf{n}_\alpha - \mathbf{n}_{Neyman}\|_p = \alpha C_p \tag{2.6}$$

Démontrons ce résultat. Par définition (2.3), on a dans chaque strate h :

$$n_{\alpha,h} = \alpha n_{prop,h} + (1-\alpha) n_{Neyman,h}$$

et donc :

$$n_{\alpha,h} - n_{Neyman,h} = \alpha(n_{prop,h} - n_{Neyman,h})$$

On a donc pour tout choix de p :

$$\begin{aligned}
\|\mathbf{n}_\alpha - \mathbf{n}_{Neyman}\|_p &= \left(\sum_{h=1}^{H} |n_{\alpha,h} - n_{Neyman,h}|^p\right)^{\frac{1}{p}} \\
&= \left(\sum_{h=1}^{H} \alpha^p |(n_{prop,h} - n_{Neyman,h})|^p\right)^{\frac{1}{p}} \\
&= \alpha \left(\sum_{h=1}^{H} |n_{prop,h} - n_{Neyman,h}|^p\right)^{\frac{1}{p}} \\
&= \alpha C_p
\end{aligned}$$

Par la suite, on intègrera C_p, constante strictement positive, dans le λ.

Annexe 2 : Démonstration du Théorème 2

Pour un $\lambda \geq 0$, la fonction de minimisation du programme (2.4) s'écrit :

$$\begin{aligned} f(\alpha) &= \sum_{h=1}^{H} n_{\alpha,h} \left(\frac{N_h}{n_{\alpha,h}} - \frac{N}{n} \right)^2 + \lambda\alpha \\ &= \sum_{h=1}^{H} \left(\frac{N_h^2}{n_{\alpha,h}} - 2\frac{N}{n}N_h + \frac{N^2}{n^2}n_{\alpha,h} \right) + \lambda\alpha \\ &= \sum_{h=1}^{H} \frac{N_h^2}{n_{\alpha,h}} - 2\frac{N^2}{n} + \frac{N^2}{n} + \lambda\alpha \\ &= \sum_{h=1}^{H} \frac{N_h^2}{\alpha\frac{nN_h}{N} + (1-\alpha)n_{Neyman,h}} - \frac{N^2}{n} + \lambda\alpha \\ &= \sum_{h=1}^{H} \frac{N_h}{\alpha\frac{n}{N} + (1-\alpha)\frac{n_{Neyman,h}}{N_h}} - \frac{N^2}{n} + \lambda\alpha \end{aligned}$$

On pose maintenant pour tout $h \leq H$:

$$\beta_h = \frac{n}{N} - \frac{n_{Neyman,h}}{N_h}$$

Les β_h représentent pour chaque strate l'écart entre le taux de sondage uniforme et celui de Neyman. Quand $\beta_h < 0$, cela signifie que la répartition de Neyman est plus grande que celle proportionnelle : la variable d'intérêt est plus dispersée dans cette strate. Dérivons maintenant f :

$$f'(\alpha) = \sum_{h=1}^{H} \frac{-N_h\beta_h}{\left(\alpha\beta_h + \frac{n_{Neyman,h}}{N_h}\right)^2} + \lambda \tag{2.7}$$

On déduit de l'équation (2.7) que la dérivée s'annule lorsque :

$$\lambda = \sum_{h=1}^{H} \frac{N_h\beta_h}{\left(\alpha\beta_h + \frac{n_{Neyman,h}}{N_h}\right)^2} =: g(\alpha)$$

Or la fonction g_h définie de la façon suivante :

$$g_h \ : \ \alpha \to \frac{N_h \beta_h}{\left(\alpha\beta_h + \frac{n_{Neyman,h}}{N_h}\right)^2}$$

est décroissante. En effet :

- Si β_h est négatif, le dénominateur décroit quand α augmente. Dans ce cas, son inverse croît avec α. Or, on multiplie par β_h pour obtenir g_h, ce qui implique que g_h est décroissante.

- Si β_h est positif, le dénominateur croît lorsque α augmente. En inversant puis multipliant par β_h, on trouve bien que g_h est décroissante.

Donc, si $\lambda \in [g(1), g(0)]$, on sait qu'il existe un α_0 qui annule la dérivée. Comme f' évolue de façon opposée à g, f' est croissante et donc α_0 est bien le minimum de f sur $[0, 1]$.

D'autre part, comme $g(\alpha_0) = \lambda$ par définition, la décroissance de g implique que lorsque que λ augmente dans $[g(1), g(0)]$, alors α_0 diminue. On utilise alors le lemme suivant, admis car relatif à une propriété classique de la répartition de Neyman :

Lemme 1. *La fonction qui à α associe la variance de l'estimateur de Horvitz-Thompson de la variable d'intérêt X pour la répartition $n_{\alpha,h}$ est croissante.*

On en déduit que $V(\lambda)$ est décroissante sur S. Enfin, par continuité, V'' admet un maximum sur S.

3

Calage sur bornes minimales et choix des bornes de calage

> *Ce travail a été présenté au 9eme colloque francophone sondages (Gatineau, Canada, octobre 2016) sous le titre "Calage sur bornes minimales et choix des bornes de calage, Emmanuel Gros et Antoine Rebecq")*

Le calage sur marges est largement utilisé en sondages afin de réduire la variance des estimations. Afin de se prémunir contre le risque d'augmentation du biais et de la variance pour certains sous-domaines de taille réduite, les méthodes « bornées » logit ou linéaire tronquée sont en général privilégiées, car elles permettent de limiter l'ampleur des ajustements de poids induits par le calage. On peut imaginer chercher les bornes conduisant à une étendue minimale des facteurs de calage pour ces méthodes bornées : on parle de « calage sur bornes minimales ». Dans ce papier, on propose un algorithme linéaire de résolution du programme de calage sur bornes minimales de taille $\mathcal{O}(n^2)$.

On effectue ensuite une étude par simulations afin de quantifier l'impact du choix des bornes de calage sur la qualité des estimateurs calés. On compare en particulier deux stratégies de choix des bornes de calage :

- le calage sur bornes minimales, qui limite au maximum l'étendue des facteurs de calage mais conduit en général à de fortes accumulations aux bornes ;
- le calage sur des bornes plus lâches, permettant d'assurer un contrôle suffisant des déformations maximales de poids induites par le calage tout en évitant les phénomènes d'accumulation aux bornes.

Il s'avère que le calage sur bornes minimales conduit à des estimateurs calés moins efficaces, et ce d'autant plus que le phénomène d'accumulation de rapports de poids aux bornes est prononcé.

Introduction

Le calage sur marges est l'une des méthodes de redressement les plus populaires et les plus utilisées en statistique d'enquête. Il consiste à remplacer les pondérations initiales des unités de l'échantillon par des pondérations calées aussi proches que possible des pondérations initiales au sens d'une certaine distance, et permettant d'assurer la cohérence entre les estimations issues de l'enquête et les totaux connus sur la population entière pour un certain nombre de variables auxiliaires. Le calage sur marges conduit en outre à un estimateur plus efficace que l'estimateur non calé pour les variables d'intérêt bien corrélées aux variables auxiliaires.

Le choix de la fonction de distance à utiliser constitue une question classique que se pose tout utilisateur de la procédure de calage sur marges. Dans leur article fondateur [DS92], Deville et Särndal soulignent que, toutes les fonctions de distance conduisant à des estimateurs calés asymptotiquement équivalents à l'estimateur par la régression, le choix de la fonction de distance n'a au final qu'un impact très marginal sur les propriétés statistiques de l'estimateur calé pour des estimations portant sur des domaines de taille moyenne ou grande. Ils notent cependant que les distances classiques du khi-deux et de l'entropie relative – correspondant aux méthodes linéaire et raking ratio – présentent un certain nombre d'inconvénients :

- elles peuvent conduire à des poids calés négatifs (pour la méthode linéaire) ou positifs mais inférieurs à 1 (pour les méthodes linéaire et exponentielles), potentiellement problématiques lors de la diffusion du fichier de données individuelles contenant les poids calés ;

- elles peuvent également induire des déformations de poids très importantes à la hausse et donc engendrer l'apparition de poids extrêmement élevés. Or de tels poids sont susceptibles de créer des unités influentes et donc de conduire à des estimations peu robustes sur certaines sous-populations de taille réduite.

C'est afin de pallier ces problèmes que [DS92] introduisent deux fonctions de distance – correspondant aux méthodes logit et linéaire tronquée – conduisant à des fonctions de calage bornées, permettant de limiter l'étendue des corrections de poids induites par le calage. Depuis, un consensus s'est établi quant au choix de la méthode de calage, qui conduit en général à privilégier l'utilisation d'une méthode bornée – logit ou linéaire tronquée, sans préférence pour l'une ou l'autre de ces méthodes – par rapport aux méthodes non bornées – linéaire ou raking ratio.

Si un consensus s'est fait jour quant au choix de la méthode de calage, la question des bornes de calage L et U à retenir lorsque l'on procède à un calage via une méthode bornée n'a en revanche été que très peu abordée à notre connaissance dans la littérature, et ne semble en tout cas pas tranchée :

- certains praticiens (cf. [Gra14] par exemple) conseillent de limiter au maximum l'étendue des corrections de poids, afin de se prémunir autant que faire se peut contre de potentiels problèmes lors d'estimations portant sur des domaines de taille réduite ;

- pour d'autres (cf. [RV01] par exemple) en revanche, les bornes L et U doivent certes être choisies de manière à éviter les déformations excessives de poids induites par le calage mais sans pour autant être trop « strictes » [1]. En effet, le choix de bornes L et U très « serrées » autour de 1 conduit en général à de fortes accumulations de rapports de poids à ces bornes, qui nous éloignent de la philosophie initiale du calage sur marges consistant à modifier « le moins possible » les poids initiaux tout en respectant les équations de calage. En outre, dans une telle configuration, on peut imaginer que des estimations portant sur certains domaines particuliers – pour lesquels tous les poids initiaux auraient été ajustés à la hausse d'un facteur U par le calage par exemple – s'avèrent problématiques.

Il semble impossible de trancher entre ces deux approches d'un point de vue théorique, puisqu'on ne dispose que de propriétés asymptotiques pour les estimateurs calés et qu'on s'intéresse justement ici à des estimations portant sur des domaines de taille réduite.

Dans cet article, nous nous intéressons dans un premier temps à la première approche de « calage sur bornes minimales ». Cette approche, déjà explorée par le passé (cf. [Van01] et [Gra14] par exemple), consiste à chercher, via une procédure d'optimisation numérique, les bornes L^* et U^* limitant au maximum l'étendue des facteurs de calage, *i.e.* telle que $U^* - L^*$ soit minimale. On présente ici ce programme ainsi qu'un algorithme optimisé de résolution, implémenté dans le package R Icarus ([Reb16]).

Puis, dans un second temps, nous procédons à une étude par simulations afin d'essayer de quantifier l'impact du choix des bornes de calage sur la qualité des estimateurs calés, pour des estimations portant sur des domaines de taille moyenne ou réduite.

[1] Il s'agit de l'approche retenue à l'Insee.

Méthode	Forme fonction de distance
Linéaire	$G(r) = \frac{1}{2}(r-1)^2$
Raking ratio	$G(r) = r\log(r) - r + 1$
Linéaire tronquée	$G(r) = \frac{1}{2}(r-1)^2$ si $r \in]L;U[$ ou $+\infty$ sinon où $L < 1 < U$
Logit	$G(r) = \frac{1}{A}\left[(r-L)\log\frac{r-L}{1-L} + (U-R)\log\frac{U-r}{U-1}\right]$ si $r \in]L;U[$ ou $+\infty$ sinon où $L < 1 < U$ et $A = \frac{U-L}{(1-L)(U-1)}$

Tab. 3.1: Fonctions de distance associées aux méthodes de calage classiques

3.1 Le calage sur bornes minimales

3.1.1 Notations et programme

On se donne une population $\mathcal{U}$ de taille N et un échantillon s de taille n, sélectionné par un plan de sondage p, de probabilités d'inclusions simples $(\pi_k)_{k\in\mathcal{U}}$. On note $d_k = 1/\pi_k$ les poids de sondages associés à l'estimateur Horvitz-Thompson. On dispose de J variables auxiliaires $X_j = (x_{jk})$, dont les totaux sur la population T_{X_j} sont connus par une source externe à l'enquête. On note $T_{\mathbf{X}} = (T_{X_1} \dots T_{X_J})'$ le vecteur des totaux de variables auxiliaires et $\mathbf{x}_k = (x_{1k} \dots x_{Jk})$ le vecteur ligne assemblant les observations des variables auxiliaires pour l'individu k. Les poids calés w_k issus d'un calage sur ces J variables auxiliaires sont solutions du problème d'optimisation suivant :

$$\begin{cases} \min\limits_{w_k} \sum\limits_{k\in s} d_k G(\frac{w_k}{d_k}) \\ \text{s.c. } \sum_{k\in s} w_k \mathbf{x}_k = T_{\mathbf{X}} \end{cases} \tag{3.1}$$

où G est la fonction de distance associée à la méthode de calage – linéaire, raking ratio, logit ou linéaire tronquée – retenue. On rappelle en table 3.1 les formes des fonctions de distances correspondant à ces méthodes de calage.

L'objectif est ici, dans le cadre d'une méthode de calage bornée, de déterminer les bornes L^* et U^* assurant une étendue minimale des **facteurs de calage** $g_k = w_k/d_k$. En notant $\widetilde{X}_s$ la matrice dont les coefficients (k,j) sont égaux à $(d_k \cdot X_{jk})_{1\leq k\leq n, 1\leq j\leq J}$,

t le vecteur des totaux T_{X_j} et g le vecteur des facteurs de calage, les équations de calage de (3.1) s'écrivent sous la forme :

$$\widetilde{X}_s' g = t$$

et les bornes L^* et U^* sont alors définies respectivement comme $\min_{k\in[[1,n]]} g_k^*$ et $\max_{k\in[[1,n]]} g_k^*$, où g^* est solution du programme de minimisation suivant :

$$\begin{cases} \min\limits_{g\in\mathbb{R}^n} \left(\max\limits_{k\in[[1,n]]} g_k - \min\limits_{k\in[[1,n]]} g_k \right) \\ \qquad \text{s. c. } \widetilde{X}_s' g = t \ ; \ g \geq 0 \end{cases} \tag{3.2}$$

La contrainte $g \geq 0$ est imposée par la méthode du simplexe utilisée pour la résolution, mais elle n'est pas gênante en pratique. En effet, la méthode linéaire est peu mise en œuvre en production car elle peut justement conduire à des poids calés négatifs. On peut donc supposer sans perte d'efficacité que $L^* > 0$.

Notons que la distance de calage n'apparaît pas dans ce programme d'optimisation. Il s'agit dans un premier temps de trouver L^* et U^* telles que $U^* - L^*$ soit minimale tout en respectant les contraintes de calage. Une fois trouvée une telle solution g_{min}, on pose $L^* = \min g$ et $U^* = \max g$ et on effectue le calage avec une méthode bornée prenant ces bornes en paramètres. L'introduction de la distance à cette étape permettra de discriminer entre plusieurs éventuelles solutions du programme de calage sur bornes minimales. Si la solution au programme est unique, g_{min} sera la solution choisie par le calage sur marges.

On peut montrer (via une écriture matricielle non détaillée ici) que le programme (3.2) peut s'écrire sous la forme d'un programme d'optimisation linéaire de taille $n \times (n-1) \times (n+1)$. On considère également deux autres programmes :

$$\begin{cases} \min\limits_{g\in\mathbb{R}^n, a\in\mathbb{R}} \max\limits_{k\in[[1,n]]} |g_k - a| \\ \qquad \text{s. c. } \widetilde{X}_s' g = t \ ; \ g \geq 0 \end{cases} \tag{3.3}$$

$$\begin{cases} \min\limits_{g\in\mathbb{R}^n} \max\limits_{k\in[[1,n]]} |g_k - 1| \\ \text{s. c. } \widetilde{X}_s' g = t \ ; \ g \geq 0 \end{cases} \tag{3.4}$$

Là encore, ces deux programmes peuvent s'écrire sous la forme de programmes d'optimisation linéaires, de tailles respectives $2n \times (n+2)$ et $2n \times (n+1)$.

3.1.2 Résolution du programme

On note respectivement S_2, S_3 et S_4 les ensembles solution des programmes 3.2, 3.3 et 3.4. Le théorème 3 caractérise les solutions de ces programmes:

Theorem 3. *On suppose que la contrainte de calage $\widetilde{X}_s' g = t$ possède au moins une solution. Alors on a:*

- $S_2 = S_3$
- $S_4 \subset S_2$

Tous ces programmes linéaires sont solubles par la méthode du simplexe de Dantzig ([Dan+55]). Résoudre le programme sur bornes minimales implique donc de résoudre par cette méthode le programme 3.3, de dimension inférieure au programme 3.2 (utilisation de la mémoire en $\mathcal{O}(n^2)$ contre $\mathcal{O}(n^3)$).

Le package R Icarus, développé dans l'optique de proposer des fonctions de calage adaptées au cadre de la production statistique (en reprenant largement l'interface de la macro SAS Calmar, [Sau93]), contient une implémentation du calage sur bornes minimales. Ce package s'appuie sur l'implémentation du simplexe proposée par le solveur linéaire *RGlpk* ([TH14]). Elle possède l'avantage de se reposer sur la gestion efficace des matrices creuses du package *slam* ([Hor+15]). En pratique en statistique publique, il arrive fréquemment que les variables de calage utilisées soient catégorielles : la matrice $\widetilde{X}_s$ possède donc généralement de très nombreux coefficients nuls.

Le programme 3.4 ne donne qu'une solution sous-optimale au problème, mais contrairement aux programmes précédents, il est soluble par dichotomie. Ainsi, pour des problèmes de taille très importante, l'algorithme du simplexe en $\mathcal{O}(n^2)$ peut imposer une charge mémoire très importante et un temps d'exécution relativement long. Bien que la convergence de la méthode par dichotomie soit théoriquement lente, elle est potentiellement plus rapide que le simplexe lorsqu'il s'agit de problèmes de grande taille. Afin de ne pas générer de surcharge mémoire sur des machines qui sont potentiellement des ordinateurs personnels, la résolution par dichotomie du programme 3.4 est utilisée quand un calage sur bornes serrées est lancé dans Icarus avec une matrice de calage contenant plus de $5 \cdot 10^8$ enregistrements.

3.2 Impact du choix des bornes de calage sur la qualité des estimateurs calés

3.2.1 Données et protocole utilisés pour l'étude par simulations

Pour réaliser notre étude par simulations, nous nous sommes appuyés sur les données du dispositif Esane [2] portant sur l'année 2013, en nous restreignant au champ du commerce (section G de la NAF rév. 2) hors exhaustif [3], soit une population $\mathcal{U}$ composée de 580 180 entreprises.

En effet, le dispositif Esane nous permet de disposer, outre de la base de sondage de l'enquête ESA, d'un grand nombre de variables fiscales disponibles pour l'ensemble des unités du champ. Ces variables fiscales vont donc d'une part pouvoir être utilisées comme variables auxiliaires dans un calage sur marges, et d'autre part permettre l'évaluation de la qualité des différents estimateurs calés via le le calcul d'erreurs quadratiques moyennes.

Afin de quantifier l'impact du choix des bornes de calage sur la qualité des estimateurs calés, nous avons procédé comme suit :

- nous avons sélectionné K=40 000 échantillons de taille 1 000 au sein de $\mathcal{U}$ par sondage aléatoire simple stratifié avec allocation proportionnelle ;

- chaque échantillon a ensuite été calé selon trois scénarios de calage – raking ratio, méthode logit avec bornes L=0,5 et U=2 (qui permet un contrôle correct des déformations maximales de poids tout en induisant *a priori* peu d'accumulation de rapports de poids aux bornes) et méthodes logit avec bornes minimales (qui devrait *a priori* conduire a de fortes accumulations de rapports de poids aux bornes) sur les marges suivantes :

[2] Le dispositif Esane (pour Élaboration des Statistiques ANnuelles d'Entreprises) combine l'enquête statistique ESA (pour Enquête Sectorielle Annuelle) et l'exploitation des déclarations fiscales des entreprises. Pour plus de détails sur le sujet, voir [Bri11].

[3] L'exhaustif de l'ESA est composé des plus grandes entreprises de chaque secteur, qui sont incluses d'office dans l'échantillon. Les grandes entreprises de chaque secteur sont définies comme celles dont l'effectif salarié, le chiffre d'affaire ou le total de bilan dépassent certains seuils, variables en fonction des secteurs. Dans les secteurs du commerce, l'exhaustif est composé en première approximation des entreprises de plus de 20 salariés ou présentant un chiffre d'affaires supérieur à 38M euros ou un total de bilan supérieur à 75M euros.

– structures par secteur d'activité (au niveau 3 caractères de la NAF rév.2), tranches d'effectif (0 salarié / 1 salarié / 2 à 5 salariés / 6 à 10 salariés / plus de 10 salariés) et ZEAT (zone d'études et d'aménagement du territoire) ;

– totaux de chiffre d'affaires, de valeur ajoutée, d'actif total et de passif total.

- Pour une variable d'intérêt Y donnée, on calcule ensuite les estimateurs calés pour chaque échantillon et chaque scénario de calage :

$$\hat{T}^k_{Y,w^{RR}} = \sum_{i \in s_k} w_i^{RR} y_i, k = 1, ..., K$$
$$\hat{T}^k_{Y,w^{logit\ [0,5-2]}} = \sum_{i \in s_k} w_i^{logit\ [0,5-2]} y_i, k = 1, ..., K$$
$$\hat{T}^k_{Y,w^{logit\ min}} = \sum_{i \in s_k} w_i^{logit\ min} y_i, k = 1, ..., K$$

- On s'intéresse plus particulièrement aux estimateurs par division, c'est-à-dire pour les variables d'intérêt Y_d définis par $y_{kd} = y_k \mathbb{1}\{k \in \mathcal{U}_d\}, k = 1, \ldots, K$ où $\mathcal{U}_d$ indique la division d

- enfin, on évalue la qualité des 3 estimateurs calés $\hat{T}_{Y_d,w^{exp}}$, $\hat{T}_{Y_d,w^{logit\ [0,5-2]}}$ et $\hat{T}_{Y_d,w^{logit\ min}}$ à l'aide de la racine carrée de leurs erreurs quadratiques moyennes relatives Monte-Carlo :

$$RRMSE(\hat{T}_{Y_d,w}) = 100 \frac{\sqrt{K^{-1} \times \sum_{k=1}^{K} (\hat{T}^k_{Y_d,w} - T_{Yd})^2}}{T_{Yd}}$$

3.2.2 Résultats

Notons tout d'abord qu'en termes d'impact du calage sur les pondérations, les résultats observés sur les simulations pour chacun des trois scénarios envisagés sont conformes aux résultats attendus :

– le calage par raking ratio conduit, quel que soit l'échantillon considéré, à une distribution des rapports de poids présentant une allure log-normale centrée aux alentours de 1, sans point d'accumulation mais avec des déformations maximales potentiellement extrêmes, certains facteurs de calage pouvant aller à la baisse jusqu'à $2,42 \cdot 10^{-16}$ et à la hausse jusqu'à 80 ;

– le calage par méthode logit avec des bornes L=0,5 et U=2 conduit, quel que soit l'échantillon considéré, à une distribution des rapports de poids présentant une allure relativement log-normale centrée aux alentours de 1 avec peu d'accumulations aux bornes (moins de 10 % des rapports de poids collés à l'une ou l'autre borne) ;

– enfin, le calage par méthode logit sur bornes minimales conduit généralement à de fortes accumulations de rapports de poids aux bornes : plus de 50 % des échantillons présentent au moins 50 % des rapports de poids collés à l'une ou l'autre des deux bornes.

Les graphiques en figures 3.1 et 3.2 permettent de comparer les RRMSE des trois estimateurs calés, pour des estimations portant sur 16 variables d'intérêt [4] ventilées selon les trois grands secteurs d'activité du commerce commerce automobile (117 unités parmi les 1 000 unités de chaque échantillon), commerce de gros (331 unités parmi les 1 000 unités de chaque échantillon) et commerce de détail (552 unités parmi les 1 000 unités de chaque échantillon). On obtient ainsi $16 \cdot 3 = 48$ estimations dont les RRMSE varient entre 0 % et 90%. Le graphique en figure 3.2 est un « zoom » du graphique du haut sur la portion [0% - 30%] des RRMSE (axe des abscisses).

On constate que le calage sur bornes minimales – induisant de fortes accumulations de rapports de poids aux bornes – conduit à un estimateur moins efficace que celui obtenu avec un calage avec des bornes L=0,5 et U=2 beaucoup moins serrées – permettant un contrôle correct des déformations maximales de poids tout en générant peu d'accumulation de rapports de poids aux bornes – lui même légèrement moins efficace que l'estimateur du raking ratio. Ce dernier présente toutefois l'inconvénient de générer des poids potentiellement extrêmes, susceptibles de créer des unités influentes pour certaines variables, raison pour laquelle on lui préfère en général une méthode bornée.

On observe des résultats similaires (présentés dans [Gro16]) pour les estimations ventilées par tranches d'effectif (les tailles de ces domaines parmi les 1 000 unités de chaque échantillon variant de 119 à 415), ainsi que pour les estimations portant sur l'ensemble du champ.

[4] Nombre d'entreprises, chiffre d'affaires, ventes de marchandises, ventes de bien, ventes de services, total des achats, valeur ajoutée, effectifs salariés, masse salariale, excédent brut d'exploitation, résultat comptable, total des amortissements, total des provisions, actif total, passif total et investissement corporel brut.

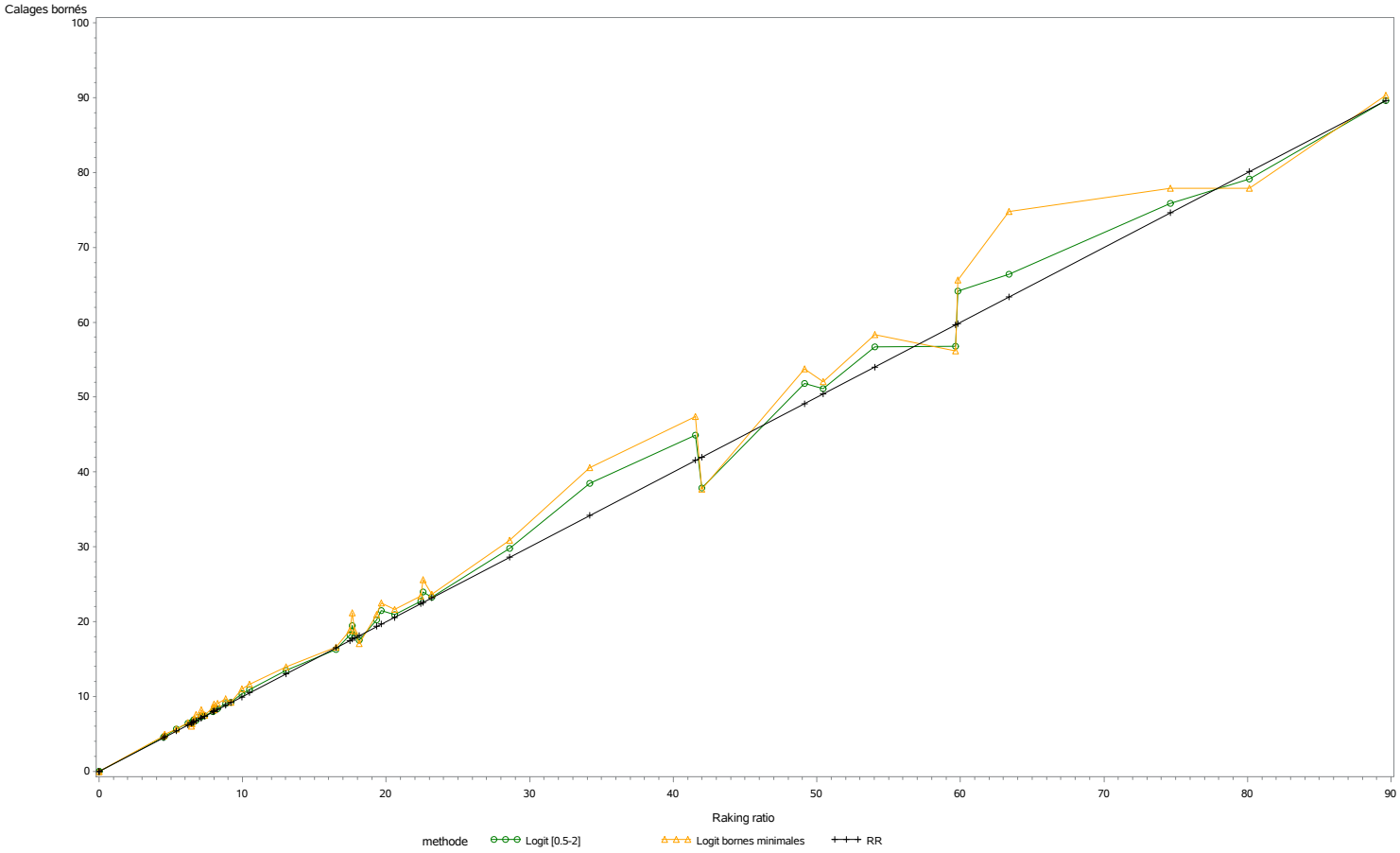


Fig. 3.1: RRMSE (en %) des estimateurs par division. *Note de lecture : chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.*

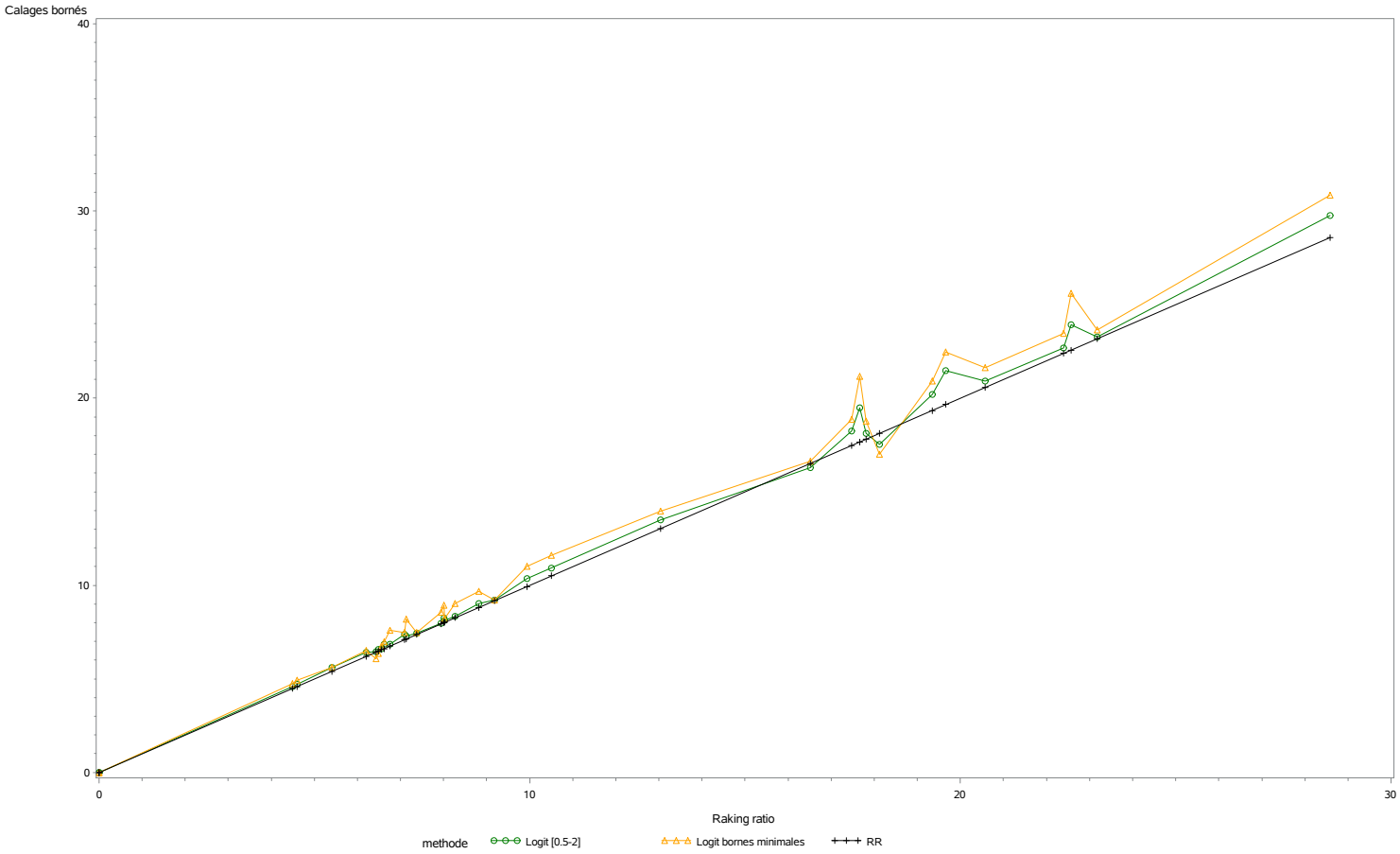


Fig. 3.2: RRMSE (en %) des estimateurs par division (zoom du graphique en figure 3.1). *Note de lecture : chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.*

Par ailleurs, cette inefficacité de l'estimateur calé sur bornes minimales par rapport à un estimateur calé sur des bornes plus lâches augmente avec l'intensité de la « saturation » aux bornes de calage : plus l'accumulation de rapport de poids aux bornes est importante, moins l'estimateur calé est efficace.

C'est ce que l'on observe lorsque l'on calcule les RRMSE de l'estimateur calé sur bornes minimales sur des sous-ensembles des 40 000 échantillons définis en fonction de l'intensité de la saturation aux bornes de calage : échantillons pour lesquels moins de 50 % des rapports de poids sont collés à l'une ou l'autre des bornes ($\sim$50 % des échantillons), échantillons pour lesquels plus de 50 % des rapports de poids sont collés à l'une ou l'autre des bornes ($\sim$50 % des échantillons), échantillons pour lesquels plus de 70 % des rapports de poids sont collés à l'une ou l'autre des bornes (sous-ensemble du sous-ensemble précédent, environ 10 % des échantillons). Ces RRMSE sont représentés sur les graphiques en figures 3.3 et 3.4, en sus des RRMSE des estimateurs déjà présentés en figures 3.1 et 3.2.

Là encore, on observe des résultats similaires (présentés dans [Gro16]) pour les estimations ventilées par tranches d'effectifs ainsi que pour les estimations portant sur l'ensemble du champ.

Il semble donc que le fait de choisir des bornes de calage trop serrées nuise à la qualité de l'estimateur calé, en particulier lorsque ces bornes de calage serrées conduisent à de fortes accumulations de rapports de poids à ces bornes.

Conclusion

Après avoir présenté la problématique du calage sur bornes minimales, nous avons proposé un algorithme linéaire de résolution de ce programme de taille $\mathcal{O}(n^2)$, implémenté dans le package R Icarus.

Ce programme de calage sur bornes minimales nous a permis de réaliser une étude par simulations, visant à quantifier l'impact du choix des bornes de calage sur la qualité des estimateurs calés. L'objectif était en particulier de trancher entre deux approches fréquemment utilisées en pratique pour choisir les bornes de calage :

- procéder à un calage sur bornes minimales, quitte à observer de fortes accumulations de rapports de poids aux bornes ;

- retenir des bornes L et U permettant d'assurer un contrôle suffisant des déformations maximales de poids induites par le calage tout en évitant les phénomènes d'accumulation aux bornes.

Les résultats de ces simulations tendent à invalider la première approche au bénéfice de la seconde, qui semble donc constituer une bonne heuristique permettant d'assurer un choix judicieux des bornes de calage.

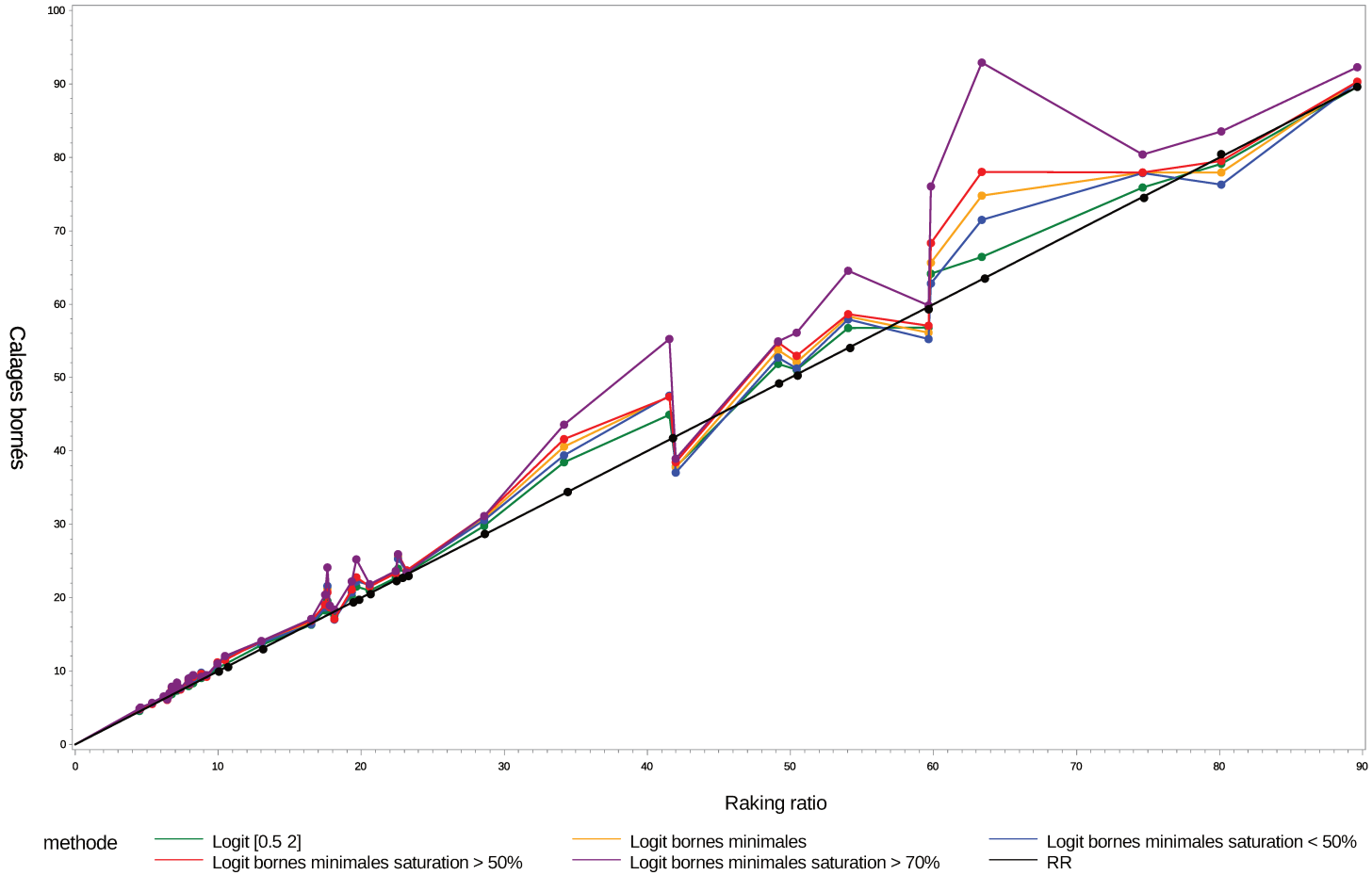


Fig. 3.3: RRMSE (en %) des estimateurs par division. *Note de lecture : chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.*

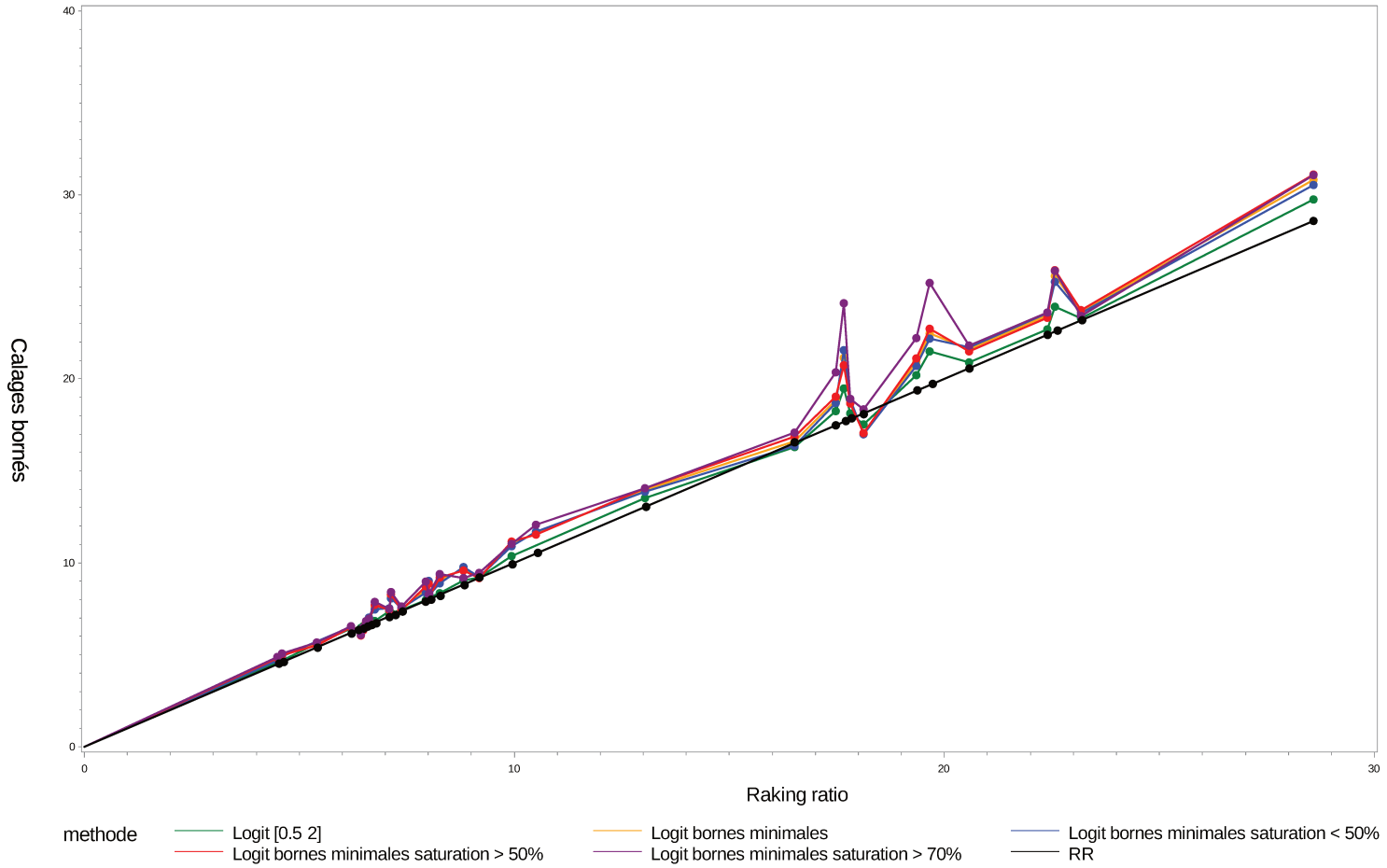


Fig. 3.4: RRMSE (en %) des estimateurs par division (zoom du graphique en figure 3.3). *Note de lecture : chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.*

Annexe: preuve du théorème 3

Soit $g \in S_1$. On pose : $a = \dfrac{\max g_k + \min g_k}{2}$, ce qui assure que $g \in S_2$. On a donc : $S_1 \subset S_2$.

On note $S_{1=}$ (resp. $S_{1\geq}$) l'ensemble des solutions de la contrainte d'égalité (resp. d'inégalité) dans le programme 3.2. On utilise les même notations pour les solutions des contraintes du programme 3.3. On a donc $S_1 = \min(S_{1=} \cap S_{1\geq})$ et $S_2 = \min(S_{2=} \cap S_{2\geq})$.

On a également $rg(A_1) = n$ et $rg(A_2) = n + 1$.

Rang de A_2 : On écrit $A_2(n)$ la matrice A_2 lorsque l'échantillon est de taille n. On commence par remarque que $rg(A_2(2)) = 3$. On traite ensuite la matrice $A_2(n)$ ($n \geq 2$) par la méthode du pivot de Gauss :

$$\begin{pmatrix} 1 & 1 & -1 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & -1 & \dots & 0 \\ \vdots & \vdots & 0 & 0 & \ddots & 0 \\ 1 & \vdots & 0 & \dots & \dots & -1 \\ -1 & \vdots & 1 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & 1 & \dots & 0 \\ \vdots & \vdots & 0 & 0 & \ddots & 0 \\ -1 & 1 & 0 & \dots & \dots & 1 \end{pmatrix}$$

On effectue, pour $i \in [[2, n]]$, la transformation $L_i \leftarrow L_i - L_1$ et pour $i \in [[n + 1, 2n]]$ la transformation $L_i \leftarrow L_i + L_1$, ce qui donne :

$$\begin{pmatrix} 1 & 1 & -1 & 0 & \dots & 0 \\ 0 & 0 & 1 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ \vdots & 0 & 1 & \dots & \dots & -1 \\ \vdots & 2 & 0 & \dots & \dots & 0 \\ \vdots & \vdots & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 2 & -1 & 0 & \dots & 1 \end{pmatrix}$$

On intervertit les lignes $n+1$ à $2n$ et 2 à n, puis pour les lignes 3 à n on effectue la transformation $L_i \leftarrow L_i - L_2$, e qui donne :

$$\begin{pmatrix} 1 & 1 & -1 & 0 & \dots & 0 \\ 0 & 2 & 0 & \dots & \dots & 0 \\ \vdots & 0 & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ \vdots & \vdots & -1 & 0 & \dots & 1 \\ \vdots & \vdots & 1 & -1 & \dots & 0 \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 1 & 0 & \dots & -1 \end{pmatrix}$$

On reconnaît dans le bloc encadré en bas à droite $-A_2(n-1)$ privée de la colonne e_{2n-2}, qui n'est pas une combinaison linéaire des autres colonnes de $-A_2(n-1)$. On conclut par récurrence que le noyau de $A_2(n)$ est de taille 1, c'est-à-dire $rg(A_2) = n+1$.

Rang de A_1 : On utilise les mêmes notations que pour le rang de A_2 et on procède également par récurrence et par pivot de Gauss. On a:

$$A_1 = \begin{pmatrix} & A_{11} \\ e_{n(n-1)} & \\ & -A_{11} \end{pmatrix}$$

La colonne $e_{n(n-1)}$ est linéairement indépendante de toutes les autres. Il suffit donc de déterminer le rang de A_{11} pour obtenir le rang de A_1. On a pour commencer : $rg(A_{11}(2)) = rg\left[\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}\right] = 1$. On remarque que A_1 peut s'écrire comme l'empilement de n à 1 des matrices $K(n) = (e_n, -I_n)$:

$$A_1 = \begin{pmatrix} & & K(n) \\ 0 & & K(n-1) \\ 0 & 0 & K(n-2) \\ \vdots & & \end{pmatrix}$$

On effectue pour les lignes 2 à n la transformation $L_i \leftarrow L_i - L_1$. On obtient :

$$\begin{pmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & & & K(n-1) & \\ 0 & & & K(n-1) & \\ 0 & 0 & & & K(n-2) \\ & & & \vdots & \end{pmatrix}$$

qui est de même rang que :

$$\begin{pmatrix} 1 & -1 & 0 & \dots & \dots & 0 \\ 0 & & & K(n-1) & & \\ 0 & 0 & & & K(n-2) & \\ & & & \vdots & & \end{pmatrix}$$

dans laquelle on reconnaît en bas à droite $A_{11}(n-1)$. On conclut par récurrence que le noyau de $A_{11}(n)$ est de dimension 1, puis que : $rg(A_1) = n$.

Les matrices A_1 et A_2 augmentées respectivement des colonnes 0_{n+1} et 0_{n+2} ont donc même rang que A_1 et A_2. On a donc par théorème de Rouché-Fontené (ou Rouché-Capelli) : $dim(S_{1=}) = 1$ et $dim(S_{2=}) = 1$. Or : $0 \in S_{1=}$ et $0 \in S_{2=}$. Donc $S_{1=}$ et $S_{2=}$ sont deux sous-espaces linéaires de même dimension. On a finalement $S_{1=} = S_{2=}$. De plus, $S_{1\geq} = S_{2\geq}$ car dans les deux cas, il s'agit simplement d'assurer les équations de calage. On en déduit finalement : $S_1 = S_2$

Soit $g \in S_3$. Alors on a immédiatement $g \in S_2$ (choix de $a = 1$). Or d'après l'item précédent, $S_1 = S_2$. On a donc $S_3 \subset S_1$.

4

Icarus : un package R pour le calage sur marges et ses variantes

Ce travail a été présenté au 9eme colloque francophone sondages (Gatineau, Canada, octobre 2016) sous le titre "Icarus : un package R pour le calage sur marges et ses variantes, Antoine Rebecq")

4.1 Introduction

Le calage sur marges, introduit par Deville et Särndal en 1992 ([DS92]) est depuis largement utilisé en production en statistique publique et par les instituts de sondage. L'utilisation du calage en statistique publique permet de présenter des estimations cohérentes entre différentes sources, mais aussi d'augmenter la précision des estimations si le calage est effectué sur des variables bien liées aux variables d'enquête. Les macros Calmar[1] ([Sau93]) et Calmar2 ([Sau03]) implémentent le calage sur marges en SAS. Ces macros ont été créées avec la vocation d'être largement utilisées dans le système statistique public.

Le langage de programmation R possède beaucoup d'avantages et est de plus en plus utilisé en statistique publique. R est un logiciel libre et open source qui comporte un langage de programmation orienté vers la manipulation de données vectorielles. Contrairement à d'autres langages utilisés fréquemment en statistique, R permet la programmation fonctionnelle (et également orientée objet), ce qui facilite le développement de méthodes réutilisables et modulaires. Ces propriétés sont largement mises à profit par la communauté statistique. Ainsi, de nombreux packages implémentent les méthodes des diverses branches de la statistique, parfois

[1] Disponible sur le site de l'INSEE : `http://www.insee.fr/fr/methodes/default.asp?page=outils/calmar/accueil_calmar.htm`

à la pointe de la recherche. Par ailleurs, en production comme pour l'utilisateur moins avancé, R peut également être utilisé sous forme de scripts.

Concernant le calage sur marges, les packages R *sampling* ([TM16]) et *survey* ([Lum+04]) en proposent une implémentation. Toutefois, ces fonctions ne présentent pas d'interface spécifiquement pensée pour l'utilisateur non expert. Ainsi, il faut transformer et réorganiser les variables (particulièrement les variables catégorielles), contrairement à l'esprit dans lequel avait été pensé la macro Calmar. Le package Icarus[2] a été développé avec la même ambition que la macro SAS. Le mécanisme d'entrée des marges est ainsi très similaire entre Icarus et Calmar : il s'agit d'indiquer dans une matrice les totaux (ou les pourcentages) associés aux modalités de chaque variable (voir partie 4.2.2). Dans le cas où l'utilisateur entrerait plusieurs contraintes de marge colinéaires (redondantes), le package Icarus comme la Macro Calmar2 traitent le problème de façon transparente en utilisant l'inverse généralisée dans le calcul de la solution.

Le package Icarus est disponible sur le CRAN, en version 0.2.1 au 29 août 2016[3]. Il peut être donc installé en utilisant la commande :

```
R > install.packages("icarus")
```

4.2 Calage sur marges

4.2.1 Notations

Considérons une population $\mathcal{U}$ de taille N pour laquelle on essaie de mesurer des caractères $Y \in \mathbb{R}^N$. Pour cela on tire un échantillon s de taille n avec un plan p de probabilités d'inclusion simples π_k. On se donne J variables auxiliaires (X_j) (définies sur toute la population, donc en particulier sur l'échantillon), desquelles on connaît les totaux $T(X_j)$. Le vecteur $T_X \in \mathbb{R}^J$ regroupe ces totaux (ou marges de calage). G désigne une pseudo-distance (ou "méthode de calage", comme définie dans [DS92]). On note enfin X_s la matrice de dimensions $n \times J$ des valeurs prises

[2] Icarus Calibrates and Reweights Units in Samples
[3] `https://cran.r-project.org/web/packages/icarus/index.html`

par les X_j sur s et $d_k = \frac{1}{\pi_k}$ le poids de sondage de l'unité $k \in s$. Le calage consiste à trouver les poids $w = (w_k)_{k \in s}$ vérifiant :

$$\begin{cases} \arg\min_{w_k} \sum_{k \in s} d_k G(\frac{w_k}{d_k}) \\ \text{s.c. } X_s' w = T_X \end{cases} \tag{4.1}$$

4.2.2 Exemple

Pour cet exemple, on considère une population de taille 300 et un échantillon par sondage aléatoire simple stratifié, où la variable d'intérêt est la fréquentation du cinéma. La population est décrite plus précisément dans le wiki du package [4]. Elle est également utilisée dans certains tests unitaires (voir partie 4.5).

Lors de l'installation de Icarus, les populations de test sont également chargées. Il est donc possible de visualiser l'échantillon de taille 15 après chargement du package:

```
R > library(icarus)
R > data_ex2
```

	id	service	categ	sexe	salaire	cinema	poids
1	a01	1	1	1	1000	1	10
2	a02	1	2	2	1100	2	10
3	a03	2	2	2	1500	4	10
4	a04	2	3	1	2300	15	10
5	a05	2	1	1	1000	2	10
6	a06	1	1	2	500	3	10
7	a07	2	2	2	1000	1	10
8	b01	1	3	2	2000	0	20
9	b02	1	1	1	2100	0	20
10	b03	2	2	1	2000	3	20
11	b04	2	1	2	3200	6	20
12	b05	1	1	2	1800	0	20
13	b06	1	2	1	2800	0	20
14	b07	1	3	1	1100	1	20
15	b08	2	1	2	2500	1	20

[4] https://github.com/haroine/icarus/wiki/Calibration

On connaît par ailleurs sur la population les nombres d'individus porteurs de chaque modalité pour les variables catégorielles "categ", "sexe" et "service" ainsi que le total de la variable quantitative "salaire". On peut créer la matrice des marges de calage ainsi :

```
## Calibration margins
mar1 <- c("categ",3,80,90,60)
mar2 <- c("sexe",2,140,90,0)
mar3 <- c("service",2,100,130,0)
mar4 <- c("salaire", 0, 470000,0,0)
marges <- rbind(mar1, mar2, mar3, mar4)
```

Comme dans Calmar2, il s'agit d'indiquer le nom et le type des variables de la table de données (en l'occurence "data_ex2") sur lesquelles le calage doit s'effectuer (première colonne de la matrice "marges"). La deuxième colonne renseigne le nombre de modalités de la variable catégorielle (on indique "0" lorsqu'il s'agit d'une variable quantitative). Les colonnes suivantes contiennent les totaux pour chaque modalité, classée par ordre alphanumérique. La matrice devant être carrée, chaque ligne de taille inférieure à 5 (la ligne correspondant à la variable "categ" contient le nom de la variable, le nombre de modalités (3), puis les 3 marges associées) est complétée par des zéros.

Il reste à effectuer le calage, par exemple avec la méthode logit et des bornes 0.4 et 2.1 :

```
R> wCalesLogit <- calibration(data=data_ex2, marginMatrix=marges,
                              colWeights="poids", method="logit",
                              bounds=c(0.4,2.1), description=TRUE)

################### Summary of before/after weight ratios ###################
Calibration method : logit
 L bound : 0.4
 U bound : 2.1
    0%     1%    10%    25%    50%    75%    90%    99%   100%
0.4010 0.4018 0.4101 0.4909 0.8523 1.2445 1.7829 2.0770 2.0903
```

Le paramètre "description = TRUE" indique à Icarus qu'on souhaite afficher dans la sortie des statistiques résumant la procédure de calage. La première partie de ces statistiques (affichée plus haut pour notre exemple) concerne la distribution

des “facteurs de calage” $g_k = \frac{w_k}{d_k}$. L'étude de la distribution des facteurs de calage permet d'apprécier l'étendue de la repondération qui est effectuée par le calage sur marges. La forme de cette distribution est parfois cruciale pour limiter le risque d'erreur d'estimation dans le cas d'enquêtes produisant des statistiques multiples (voir par exemple [GR16]). Icarus présente dans la sortie texte quelques quantiles de la distribution, et dans la sortie graphique on affiche le graphe de la densité estimée par noyau.

La seconde partie de la sortie (non retranscrite ici) indique l'écart entre l'estimation initiale et les marges de calage pour chaque variable du problème. Ceci permet de faciliter le déboguage des programmes de redressement, mais aussi parfois d'interpréter plus finement la repondération qui est effectuée.

4.3 Calage pénalisé

4.3.1 Principe

Le calage pénalisé consiste à relâcher la contrainte de marges et à l'intégrer dans le programme d'optimisation. Cette méthode a été étudiée entre autres dans [BB08; GT10]. Elle permet de faciliter la convergence de la procédure et ainsi d'augmenter le nombre de variables pour lesquelles la valeur de l'estimation redressée est contrôlée tout en préservant une distribution des facteurs de calage peu étendue. On conserve les notations de la partie 4.2. On note également $\widehat{T_{Xw}}$ le vecteur des estimateurs utilisant les poids w pour les variables auxiliaires X_j (vecteur des $\sum_{k\in s} w_k x_{jk}$). On se donne C est un vecteur de coût de taille le nombre de marges dans le programme de calage et $diag(C)$ désigne la matrice diagonale de dimensions $J \times J$ où les coefficients diagonaux sont les valeurs du vecteur C. Le programme de calage pénalisé s'écrit :

$$\min_{w_k} \sum_{k\in s} d_k G(\frac{w_k}{d_k}) + \lambda(\widehat{T_{Xw}} - T_X)diag(C)(\widehat{T_{Xw}} - T_X) \tag{4.2}$$

Il est à noter que l'on peut requérir un calage exact pour certaines marges en fixant un coût infini. Le paramètre λ est compris entre 0 et $+\infty$ et représente l'importance relative dans le programme de la distance aux poids initiaux et de l'écart aux marges des estimations redressées.

Tout comme pour le calage sur marges avec contraintes, on peut utiliser plusieurs distances G. On s'aperçoit néanmoins qu'introduire des distances bornées n'est pas

utile pour limiter l'étendue des facteurs de calage (de repondération). En effet, si $\lambda \to +\infty$, alors le terme de coût est prépondérant : les contraintes de marges sont satisfaites en priorité, ce qui éloigne les facteurs de calage de 1. Si $\lambda \to 0$, alors le terme de distance est prépondérant, ce qui tend à rapprocher les facteurs de calage de 1.

Ainsi, Icarus choisit la plus grande valeur de λ telle que l'étendue de la distribution des facteurs de calage est inférieure à une valeur choisie par l'utilisateur (paramètre *gap*). Les distances bornées ne sont donc plus nécessaires. Ainsi, les deux distances disponibles pour le calage pénalisé sont la distance du khi-deux (" method='linear' ") et l'entropie relative (" method='raking' "). Dans le cas de la méthode linéaire, il existe une solution analytique. Pour la méthode du raking ratio, Icarus calcule la solution à l'aide de l'algorithme ICRS décrit par Bocci et Beaumont ([BB08]).

Le calage pénalisé a été récemment mis en œuvre en production à l'INSEE[5] en utilisant Icarus ([RFM16; Gro15]). Il convient de noter qu'il n'existe pas à ce jour de moyen d'imposer une erreur relative d'estimation *a priori* pour les variables de cadrage. L'obtention d'une solution satisfaisante pour le statisticien se fait de manière empirique en jouant à la fois sur les paramètres de coût et de gap.

4.3.2 Exemple

On reprend l'exemple développé en partie 4.2.2. Cette fois ci, on souhaite resserrer les bornes du calage de manière à obtenir une étendue de distribution inférieure ou égale à 1.4. On utilise donc la méthode logit avec des bornes 0.6 et 2.0 :

```
R > calibration(data=data_ex2, marginMatrix=marges,
            colWeights="poids", method="logit",
            bounds=c(0.6,2.0), popTotal = 230)
```

Malheureusement, ce programme de calage ne possède pas de solution, ce qu'indique Icarus :

```
Warning messages:
In calibAlgorithm(Xs, d, total, q, inverseDistance, updateParameters,  :
  No convergence
```

[5] Institut National de la Statistique et des Études Économiques, institut de statistique officielle en France

On choisit alors d'utiliser le calage pénalisé : on relâche la contrainte pour certaines marges afin d'obtenir une convergence avec l'étendue de distribution souhaitée. On décide que le calage doit être exact pour la variable "salaire", mais que les autres marges ont la même importance relative :

```
R > costs <- c(1,1,1,Inf)
R > calibration(data=data_ex2, marginMatrix=margins, colWeights="poids"
                , costs=costs, gap=1.4, description=TRUE, popTotal=230)
```

Après quelques itérations, Icarus affiche dans la log la sortie suivante :

```
Test with lambda = 0.00674285392857323
[1] 0.1949513
[1] 1.595005
Found lambda = 0.00674285392857323 ; count = 30

################### Summary of before/after weight ratios ###################
Calibration method : linear
Mean : 0.9661
    0%     1%    10%    25%    50%    75%    90%    99%   100%
0.1950 0.2434 0.5774 0.7712 0.9579 1.1268 1.4663 1.5819 1.5950

#################### Comparison Margins Before/After calibration ####################
Careful, calibration may not be exact
$Total
Before calibration  After Calibration             Margin
               230                230                230

$categ
  Before calibration After Calibration Margin
1                110             92.77     80
2                 70             84.08     90
3                 50             53.15     60

$sexe
  Before calibration After Calibration Margin
1                110             128.9    140
2                120             101.1     90

$service
```

```
  Before calibration After Calibration Margin
1                130            110.14    100
2                100            119.86    130

$salaire
Before calibration  After Calibration             Margin
            434000             470000             470000
```

Ainsi, le calage est bien effectué exactement pour la variable “salaire”, et il subsiste une erreur des estimations pour les autres variables “de cadrage” (pour lesquelles le calage n’est pas effectué de façon exacte). On peut aussi remarquer que la distribution finale n’est pas centrée sur 1, ce qui n’est pas surprenant étant donné que l’échantillon est de taille très faible.

4.4 Calage sur bornes serrées

Les distances permettant d’inclure des bornes sont utiles lors d’un calage sur marges, car cela permet d’éviter de prendre le risque de repondérer trop fortement (notamment à la hausse) des unités influentes. De même, cela limite les risques d’explosion de la variance pour les estimations sur des domaines qui concentreraient un bon nombre de facteurs de repondération extrêmes ([RV01]).

Pour prolonger cette logique, on pourrait chercher les bornes L et U les plus rapprochées telles que l’équation de calage (4.1) possède toujours une solution. C’est le principe du calage sur bornes serrées. Le programme à résoudre est le suivant :

$$\begin{cases} \min\limits_{g\in\mathbb{R}^n} \left(\max\limits_{k\in[\![1,n]\!]} g_k - \min\limits_{k\in[\![1,n]\!]} g_k \right) \\ \text{s. c. } \widetilde{X}_s' g = T_X \ ; \ g \geq 0 \end{cases} \tag{4.3}$$

avec:

$$\widetilde{X}_s = diag(d) X_s$$
$$\forall k \in [[1,n]], g_k = \frac{w_k}{d_k} \text{ facteurs de calage}$$

Ce programme est linéaire et peut-être résolu par la méthode du simplexe (voir par exemple [Van01] ou [GR16]). Le meilleur algorithme à notre disposition (et implémenté dans Icarus) nécessite toutefois un nombre $\mathcal{O}(n^2)$ d'enregistrements en mémoire vive, ce qui peut être prohibitif pour des grandes tailles d'échantillon. Une méthode moins intensive en mémoire consiste à chercher ces bornes par dichotomie (voir [GR16]). Dans ce cas, la solution obtenue est sous-optimale, mais les résultats obtenus sont proches pour la plupart des échantillons.

Cette méthode par dichotomie est sélectionnée par défaut dans Icarus dès lors que $n \geq 10000$, afin de ne pas excéder la capacité en mémoire vive d'un PC de bureau typique en 2016. Il est toutefois possible (par exemple si l'on dispose d'un serveur de calcul) de forcer l'utilisation de l'algorithme du simplexe pour résoudre le programme (et donc aboutir à une solution optimale) en utilisant le paramètre "forceSimplex = TRUE".

Si les bornes serrées optimales ont été obtenues par la méthode du simplexe, un calage logit est testé avec les bornes obtenues. Il est toutefois rare que le calage converge en un temps acceptable avec ces bornes optimales. Pour cette raison, les bornes optimales sont tronquées avant que le calage soit testé. Si la convergence n'est toujours pas obtenue, une solution convergente est cherchée par dichotomie entre les bornes obtenues par méthode linéaire et les bornes obtenues par la méthode du simplexe. La précision avec laquelle la solution finale est demandée peut être réglée avec le paramètre *precisionBounds*.

Le calage sur bornes serrées se distingue par la forme de la distribution des facteurs de calage, qui présente fréquemment une forme "en U". Une étude statistique de l'estimation par calage sur bornes serrées (en particulier quand la distribution des facteurs de calage présente cette forme typique) est effectuée dans [GR16].

4.5 Tests unitaires et développements futurs

En développement logiciel un test unitaire est une procédure permettant de vérifier qu'une fonction (ou partie) d'un programme fonctionne comme attendu. Cela permet d'assurer que les spécifications des fonctions écrites sont respectées, et également que l'ajout futur de fonctionnalités n'entraînera pas de régression (défaut introduit dans un logiciel à l'occasion d'un changement - par exemple correction d'un bug). Dans la logique de l'intégration continue, les tests doivent être lancés et validés pour chaque modification en production.

Icarus ayant vocation a être largement utilisé en production dans les instituts de statistique publique, il convient de limiter au maximum le risque d'une erreur provenant de l'implémentation. Ainsi, de nombreux tests unitaires ont été intégrés au processus de compilation de Icarus, couvrant une majorité des paramètres utilisables de la fonction "calibration". Ces ajouts ont été facilités par l'excellent package test_that de Hadley Wickham ([Wic11]).

L'ambition pour le futur d'Icarus est de maintenir le développement actif, en étant particulièrement attentifs aux remontées d'utilisateurs qui pourraient être faites[6]. Il est également prévu d'ajouter des fonctions pour le calage simultané ([CVS12]) et le calage sur variables non linéaires ([Les09]), ainsi que des fonctions facilitant la correction de la non-réponse par repondération (notamment par GRH[7]).

[6] La page privilégiée pour rapporter un bug ou une demande de fonctionnalité est la page "issues" de GitHub : `https://github.com/haroine/icarus/issues`

[7] Groupes de Réponse Homogène

Part II

Asymptotics of survey sampling

5 Asymptotics of survey sampling

This work was done in collaboration with Patrice Bertail

Introduction

In many statistical applications, the data is collected through some survey sampling designs or subject to some known biased sampling scheme. In that case, statisticians have at their disposal some (random or fixed) weights corresponding either to true probabilities of inclusion, as is often the case for institutional data, or to some post-stratification weights. Ignoring the method used to form the database can often result in significant biases (a problem which becomes increasingly problematic for big data), thereby completely ruining the estimation and prediction problems (see [Cha+11], [Wan12]). To avoid such drawbacks, Horvitz-Thompson type of estimators can incorporate the underlying sampling design weights to obtain unbiased estimators. From another point of view, when the available data is so voluminous that a single computer cannot treat the entire dataset, survey sampling appears as a natural remedy, see [Car+13b],[Car+13a] or [Ber+16]. As opposed to simple sub-sampling, sampling designs with unequal probabilities offer a control over the efficiency of estimators. If in addition their implementation is not too expensive, they can clearly benefit the statistician in the big data era.

Many asymptotic results for different sampling methods have been obtained in the late seventies but are only available for very specific survey sampling plans. The first important central limit theorem for Horvitz-Thompson estimators was obtained by [Háj64] for the rejective algorithm. [Ber98] proved asymptotic normality for some sampling designs (including Rao-Sampford, successive sampling) close in term of L1 metric or entropy metric to the rejective sampling plan. Since very recently, very few functional results were available for empirical processes.

Obtaining asymptotic results in sampling is difficult (and somehow unnatural) essentially for two reasons. First, the inclusion probabilities are different for each unit of the population (and chosen to optimize the precision of the estimator), so that the observations are clearly not identically marginally distributed. Simple conditions on particular sampling designs, ensuring that a Lindeberg-Feller type conditions hold, are examined in [Ber+16]. Second, observations coming from sampling designs are generally not independent nor exchangeable. The purpose of this paper is to further study Horvitz-Thompson empirical process indexed by some regular classes of functions or functional Central Limit Theorems (CLT) for a large class of survey sampling designs, including multi-stage sampling. This problem has been addressed by [BW07] and [Gil+88] in the particular case of a stratified sampling plan, where the individuals are selected at random (without replacement) in each stratum, by means of bootstrap limit results. These papers clearly show that the literature on bootstrap and resampling plans may also be useful in the framework of survey sampling, the notion of resampling plan being simply

replaced by the survey sampling plans (taking only values in {0,1}). However the main difference is that resampling plans in the bootstrap literature possess some exchangeable properties which is not the case for general survey sampling plans. Recently [Ber+16], [Boi+17] have obtained some functional CLT paving the way for studying Hadamard and Fréchet differentiable functionals and more generally nonparametric procedures in the framework of survey sampling. [Boi+17] focus at the empirical cdf for quite general plan for which one has conditions on the fourth order inclusion probabilities (conditions which may be difficult in practice). On the other hand, [Ber+16] obtain the behavior of empirical processes indexed by general classes of functions but only for conditional Poisson or rejective survey sampling designs.

In this paper, we focus more precisely on the general study of the Horvitz-Thompson empirical process indexed by classes of functions, as considered in [Ber+16], for a much more general class of survey sampling possessing the negative association property. This covers a very large class of survey sampling plans including Poisson sampling, Rejective sampling ([Háj64]), pivotal sampling ([DT98]), determinantal sampling ([LM18]).

Indeed a very useful framework to study dependent variables in survey sampling is the theory of negative dependence (see [JDP83]), which has gained considerable attention in the last years. A particularly important breakthrough was made by [BB09] and applied to sampling in [BJ12]. However, this paper does not propose an adequate criterion or a tailor-made CLT to obtain asymptotic normality of survey sampling designs even in the case of rejective sampling.

In this article, we first review various concepts and definitions related to negative dependence and show/recall that many of the sampling algorithms that are used in practice satisfy these properties. This also includes some resampling designs used in the weighted bootstrap literature (see [BB95], [BC94]). Then, we establish conditions on the sampling designs so that the estimates are asymptotically normal, using a variation of the CLT for stationary negatively associated r.v.'s (notice that, in general, stationarity can not be assumed for survey sampling designs). These conditions only depend on the first and second-order inclusion probabilities. Next, we prove a Donsker type theorem for Horvitz-Thompson type empirical processes just like [Ber+16] for general survey sampling designs having the negative association properties. The result is obtained for general class of functions under a standard L2 condition on the envelop of the class and a classical Dudley's condition on the uniform entropy of the class. We show that these hypotheses are satisfied by a large class of survey sampling designs and classes of functions. As a by product we show

how we can obtain a FCLT's for two stages sampling design satisfying the negative association properties at each stage. From these results we deduce the asymptotic distribution for plug-in estimators of Hadamard differentiable functionals.

5.1 Negative association of survey sampling and resampling designs

[BB09] and [BJ12] have recently emphasized the importance of the concept of negative association in survey sampling. Negative and positive associations are investigated in the master paper of [JDP83]. Negative and positive association are often used when studying stationary time-series, see for instance the monograph of [Oli12]. However, from the work of [JDP83], it can be seen that many resampling designs appearing in resampling procedures (including multinomial, Dirichlet, exchangeable weights summing to 1 with a log-concave distribution) are actually negatively associated and as a consequence any positive linear combination of the preceding also satisfy this property. The same kind of properties also holds for many "non-stationary" survey sampling designs as shown in [BJ12]. This is actually also a key tool in proving the tightness of the empirical process in the paper by [Ber+16] and it is also successfully used to obtain straightforward Bernstein or Bennett inequalities in [BC10]. We thus recall a few important facts on negative association and show it automatically leads to a central limit theorem for many sampling designs.

5.1.1 Notations and definitions

In the following we denote by $[\![1, N]\!]$ the set of all integers from 1 to N. Let $\mathbf{Z}_N = (Z_i)_{i\in[\![1,N]\!]}$ be a sequence of (possibly dependent) random variables defined on the same probability space, valued in a measurable space $(E, \mathcal{E})$.

Definition 1. *Pairwise negative correlation (PNC).* $\mathbf{Z}_N$ *is said to be pairwise negatively correlated if for* $i, j \in [\![1, N]\!]$*:*

$$EZ_iZ_j \leq (EZ_i)(EZ_j).$$

Example 4. *Any set of exchangeable random variables (such that the distribution of* $(W_{1,N}, ..., W_{N,N})$ *is invariant by any permutation of the indexes) and such that*

$\sum_{i=1}^{N} W_{i,N} = n$ *(for some fixed* n*) is negatively correlated. Indeed, by exchangeability one has that* $EW_{i,N} = n/N$ *so that one gets that*

$$E\left(W_{1,N}\sum_{i=1}^{N}(W_{i,N} - \frac{n}{N})\right) = 0,$$

yielding, by exchangeability, that

$$(N-1)Cov(W_{i,N}, W_{j,N}) = -V(W_{1,N}),$$

implying in turn the pairwise negative correlation property. Such weights appear naturally in many resampling procedures (see [BB95]) as well as in sampling with equal probabilities. Examples of exchangeable survey sampling designs are Poisson sampling with equal probabilities, rejective sampling with equal inclusion probabilities or subsampling.

The bootstrap resampling design corresponds to the multinomial distribution $Mult(N, (1/N, ..., 1/N)$*, the mooN (m out of* N*) bootstrap corresponds to* $Mult(m, (1/N, ..., 1/N),\ m < N$ *(here* $n = m$*). The bayesian bootstrap corresponds to the Dirichlet distribution* $D(1/N, 1/N, ..., 1/N)$*, etc. Actually these weights enjoy much more stronger properties: negative cylinder and negative association properties as can be seen from [JDP83].*

Definition 2. *Negative cylinder property (NCP). Define* $A = A_1 \cup A_2$ *where* A_1 *and* A_2 *are disjoint subsets of* $[\![1, N]\!]$*.* $\mathbf{Z}_N$ *is said to satisfy the negative cylinder property if:*

$$E\prod_{i\in A_1} Z_i \prod_{j\in A_2} Z_j \leq \prod_{k\in A} EZ_k.$$

PNC and NCP follow from the following general definition of negative association (see [JDP83]).

Definition 3. *The r.v's* $Z_1,\ \ldots,\ Z_N$ *are said to be negatively associated iff for any pair of disjoint subsets* A_1 *and* A_2 *of the index set* $\{1,\ \ldots,\ N\}$

$$Cov\left(f((Z_i)_{i\in A_1}),\ g((Z_j)_{j\in A_2})\right) \leq 0, \tag{5.1}$$

for any real-valued measurable functions $f : E^{\#A_1} \to \mathbb{R}$ *and* $g : E^{\#A_2} \to \mathbb{R}$ *that are both increasing in each variable.*

Example 5. *Another interesting example of negative association in the resampling literature is the case of bootstrap Bayesian Clones (see [Lo91; Lo93]) also studied in [BB95]. Indeed if we choose* $W_{i,N} = N\gamma_i / \sum_{i=1}^{N} \gamma_i$ *where* γ_i *are i.i.d strictly positive random variables with a log-concave density, then from [JDP83] it is immediate to see that the vector* $(W_{1,N}, ..., W_{N,N})$ *is exchangeable (summing to* $n = N$*) negatively*

correlated but also negatively associated. The particular case where the $\gamma_{i's}$ are i.i.d exponentially distributed with parameter 1, $\gamma(1)$, corresponds to the Bayesian bootstrap. As recalled in example 1, the multinomial and Dirichlet weights corresponding to respectively to naive bootstrap and bayesian bootstrap are also negatively associated. The (survey) sampling design corresponding to subsampling (without replacement) frequently used in the resampling literature (see [Pol+99]) is also negatively associated. As we will see after most of the usual survey sampling plan are negatively associated.

For this we recall a stronger property called the Strong Rayleigh property. The study of this property is fairly recent and has proven to be very useful (see [BJ12]) in checking negative association for several sampling designs.

Definition 4. *Strong Rayleigh property (SR). Assume now that E is a discrete (countable) set. Denote F the probability-generating function of the discrete measure μ taking it value on the set E^N, defined as:*

$$F(z) = \sum_{x=(x_1,..,x_N)\in E^N} \mu(x) z^x,$$

where $z = (z_1, \ldots, z_N) \in \mathbb{C}^N$ and

$$z^x = \prod_{i=1}^{N} z_i^{x_i} .$$

*Then μ is said to be **strongly Rayleigh** (SR) if for all $x \in \mathbb{R}^N$ and $1 \leq i < j \leq N$:*

$$F(x)\frac{\partial^2 F}{\partial x_i x_j}(x) \leq \frac{\partial F}{\partial x_i}(x)\frac{\partial F}{\partial x_j}(x).$$

Theorem 6 (Pemantle,2000)**.** *Strong Rayleigh property implies Negative Association.*

Proof. see Item (i) of theorem 6.9 of [Pem11]. □

5.1.2 Negative association of survey sampling designs

In the rest of this article, we will mainly focus on the special case of survey sampling (without replacement).

We consider a finite population of $N \geq 1$ distinct units, $\mathcal{U}_N = [\![1, N]\!]$. We denote $\mathcal{P}(\mathcal{U}_N)$ the power set of $\mathcal{U}_N$. A survey sample of (possibly random) size $n \leq N$ is any subset $s = \{i_1, \ldots, i_{n(s)}\} \in \mathcal{P}(\mathcal{U}_N)$ of size $n(s) = n$. A sampling design without replacement is defined as a probability distribution R_N on the set of all

possible samples $s \in \mathcal{P}(\mathcal{U}_N)$. Under this framework, we are interested in some positive characteristic Y of the population. Y is called the variable of interest, and can be either random not. We assume positivity for the sake of simplicity, however most of the results obtained here are also valid for general variables by decomposing the variable into its positive part $Y^+ = \max(Y, 0)$ and it negative one $Y^- = \max(-Y, 0)$ with $Y = Y^+ - Y^-$.

We shall assume first the following superpopulation model: all N random vectors Y_i, $i \in \mathcal{U}_N$ have equal marginal distribution P_Y. For sake of simplicity, when there is no ambiguity, we will denote $\mathcal{U}_N$ by $\mathcal{U}$.

The sample data is used to construct a statistical estimate of the total

$$T(Y) = \sum_{k \in \mathcal{U}} Y_k.$$

We denote for $k \in \mathcal{U}$: $\eta_k = \mathbb{1}\{k \in s\}$ the indicator that unit k is included in the sample, $\pi_k = P(k \in s) = E\eta_k$ the *first-order inclusion probabilities* and $\pi_{kl} = P(\{k, l\} \in s) = E(\eta_k \eta_l)$, $(k, l) \in \mathcal{U}^2$ the *second-order inclusion probabilities*.

The Horvitz-Thompson estimator of the total of the variable Y is given by

$$\widehat{T} = \sum_{k \in \mathcal{U}} \frac{Y_k}{\pi_k} \eta_k.$$

The variables η_k are neither identically distributed (because of the unequal inclusion probabilities) nor independent. For instance, a very common feature in sampling is to require a sample of fixed size $\sum_{k \in \mathcal{U}} \eta_k = n$. In that case the inclusion indicators are clearly dependent.

Definition 5. *Sen-Yates-Grundy condition*

The Sen-Yates-Grundy condition (which is very often discussed in survey sampling literature but also in practice) writes, for all $k, l \in \mathcal{U}$:

$$\Delta_{kl} = P(\{k, l\} \in s) - P(k \in s)P(l \in s) = \pi_{k,l} - \pi_k \pi_l \leq 0.$$

This condition ensures that the Horvitz-Thompson estimators of $V(\widehat{T})$ (derived first in [HT52], and which are those that are almost always used in practice) are positive. We see that this condition is exactly PNC (see definition 1) for the η_k's. As negative

association implies pairwise negative correlation, in the case of single stage sampling, NA sampling designs thus yield positive variance estimators.

Now we note that in the case of survey sampling sampling the $\eta_i's$ are 0-1 valued. Thus their generating function F is multi-affine. We denote by $\mathbb{C}[z_1, \ldots, z_N]$ the set of polynomial of degree N on $\mathbb{C}$. To characterize negative association, the following additional definition and results are needed:

Definition 6. *Stable polynomial. A polynomial $f \in \mathbb{C}[z_1, \ldots, z_N]$ is called stable if $f(z_1, \ldots, z_N) \neq 0$ whenever $\mathrm{Im}(z_j) > 0$ for $1 \leq j \leq N$. A stable polynomial with all real coefficients is called real stable.*

We can then present a very useful theorem due to [Bor+09]: see also Proposition 2 of [BJ12].

Theorem 7 (Borcea, 2009)**.** *Stability for multi-affine generating functions: Let $\eta_1, \ldots, \eta_N$ be 0-1 random variables and μ the corresponding probability measure. Then μ is strongly Rayleigh if and only if the generating polynomial F is stable in the sense of definition 6.*

This very powerful theorem makes it relatively easy to check if some collection of variables (sampling designs as for section 5.2) possesses the Strong Rayleigh property, and consequently are NA.

5.2 Some negatively associated sampling algorithms

The value of $\widehat{T}$ and $V(\widehat{T})$ only depend on the first and second-order inclusion probabilities. Thus, implementing a sampling design means finding an algorithm able to select units from a population, while respecting the value of π_k and π_{kl}. Often, a few other conditions on the sample will be required (fixed sample size, balancing constraints, etc.). Sometimes we can even just focus on the first-order inclusion probabilities if we do not need a closed-form formula for the variance of the estimates (see for example [DT04] and [Cha11]). When the π_k are constructed to be proportional to an auxiliary variable X called the size), we will refer to these sampling plans as *pps sampling* ("proportional-to-size sampling").

Fortunately a lot of algorithms implementing equal or more generally unequal probability sampling satisfy the strong Rayleigh property and are thus negatively

associated. In practice, we will always have at our disposal one negatively associated algorithm to implement pps sampling with or without extra conditioning on the sample. A counter-example is successive sampling (see [Ale89]). However, the successive sampling algorithm is neither efficient nor has good statistical properties, and to the knowledge of the authors, it is hardly ever used in practice.

5.2.1 Rejective sampling

Hajek's rejective algorithm

Rejective sampling was first introduced by Hajek ([Háj59; Háj64]). Given N parameters $p_1, \ldots, p_N$ such that $\forall i \in [\![1, N]\!],\ 0 < p_i < 1$, a sampling design will be called *rejective* of size n if the probability of selecting s among the 2^N elements of the power set of $\mathcal{U}$ writes:

$$\begin{aligned} R(s) &= C \prod_{k \in s} p_k \prod_{k \in \mathcal{U}-s} (1 - p_k) \text{ if } \#s = n \\ &= 0 \qquad\qquad \text{otherwise,} \end{aligned}$$

where $C > 0$ is a normalization constant.

This means that samples of n units are selected until we find one with n distinct units. The expected value of number of attempts needed until such a sample is found is of the order of $\sqrt{n}$. The definition of the rejective algorithm can also be broadened to add more conditions than just the sample size. In this case, the number of attempts needed to find an adequate sample can become huge, depending on the number of constraints we add.

In addition of not being efficient in terms of speed and memory, rejective sampling has the major drawback that the parameters p_k are not equal to the first-order inclusion probabilities π_k, though a recursive algorithm allows to find p_k corresponding to the desired π_k. This algorithm consists in solving a linear system of equations or by using recursive algorithms (see [Che+94]). Rejective sampling, however, is very important in sampling theory, and many asymptotic results depend on the influential paper [Háj64] in which were derived some important properties of the rejective algorithm. For example, [Ber98] proves asymptotic normality for any sampling under a criterion of distance to the rejective design. Generalization of the CLT to

functional CLT and Donsker type theorem have been obtained for rejective sampling in [Ber+16].

Conditional Poisson Sampling

Rejective sampling may also be interpreted as a Conditional Poisson Sampling (CPS) (see [Háj64]). Suppose we want to implement a pps sampling design which yields samples of fixed size n. A simple way to do so would be to do Poisson experiments for each unit k (with parameter π_k) to decide whether or no the unit is included in the sample. If the final sample does not contain exactly n units, we reject the sample and carry on with the procedure until a sample of size n is obtained. CPS has also drawn a lot of attention in sampling theory: [Che+94] proved that CPS is in fact the maximum entropy sampling design conditional to the size of the sample. An equivalent way to implement CPS is to pick samples of n units independently according to $\frac{cp_i}{1-p_i}$ until a sample with n distinct units is obtained.

Rao-Sampford sampling

Another closely related sampling design is the Rao-Sampford algorithm. The distribution of the design is given, for all $s \in \mathcal{P}(\mathcal{U})$, by

$$Q(s) = C \left(\sum_{k \notin s} \pi_k(Q) \right) \left(\prod_{k \in s} \frac{\pi_k(Q)}{1 - \pi_k(Q)} \right),$$

where $C > 0$ is a normalization constant. The procedure starts by picking one unit according to probability $\frac{\pi_k}{n}$ (that is according to a multinomial scheme $Mult(1, (\frac{\pi_k}{n})_{k \llbracket 1,N \rrbracket})$). The remaining $n-1$ units are selected with a CPS of parameters $\frac{c\pi_k}{1-\pi_k}$ (c being a normalization constant) with the additional condition that the first unit selected is not included in the sample of size $n-1$. The main advantage of Rao-Sampford's method is that the parameters used in the algorithm are directly the first-order inclusion probabilities of the units ([Háj81]). In addition, its entropy is very close to theoretical achievable maximum.

Theorem 8. *Rejective, Conditional Poisson and Rao-Sampford samplings are Negatively associated*

The proof is carried out in [BJ12], using the so-called theorem on conditional-balls-and-bins (CBAB) sampling. Donsker type results for Horvitz-Thompson empirical process have been proved in [Ber+16] (for general class of functions) and are also

covered by the results of [Boi+17] for the cdf. In the following we give additional plans for which such type of results does not exist to our knowledge.

5.2.2 Determinantal sampling

Definition 7. *A sampling design is said to be determinantal if there exists K an hermitian matrix of dimension N^2 (called the kernel) such that for all $s' \subset \mathcal{P}(\mathcal{U})$:*

$$P(s' \subset s) = \det\Big(K_{|s'}\Big),$$

where $K_{|s'}$ is the restriction of K with the rows and columns corresponding to units of s'

A good review on determinantal sampling designs with a lot of interesting results can be found in [LM18]. Determinantal algorithms can be implemented using the algorithm described in [Hou+06]. Their main advantage is that the first and second-order inclusion probabilities can be computed in closed form. Consequently, the mean squared error of the Horvitz-Thompson estimator also has a closed form expression, which makes it easy to create samples respecting some additional criteria on the mean squared error (for example balancing constraints).

Theorem 9. *Determinantal sampling is Strongly Raleigh and thus NA.*

The proof of this theorem can be found in [Lyo03].

5.2.3 Pivotal (or Srinivasan) sampling

Pivotal sampling was introduced by [DT98]. In computer science literature, this algorithm is known as Srinivasan sampling. This algorithm has drawn of lot of attention because of its simplicity. The entropy of this sampling design is much lower than the entropy of the rejective methods. One way of increasing the entropy of the pivotal algorithm is to re-order randomly the units before the draw is made.

Theorem 10. *Pivotal (Srinivasan) sampling is NA.*

Proof. The proof can be found in theorem 3 of [BJ12] □

5.2.4 Cluster and multistage sampling designs

The previous results apply in the case of single stage sampling, where units of the population $\mathcal{U}$ are directly drawn. In many real-life situations, indirect sampling designs such as cluster or multistage sampling are often carried out, mostly because of operational constraints. In this section we show that the a NA property can still be obtained for the Horvitz-Thompson estimator using data from a multistage sampling design with some additional hypotheses on the designs used at each stage (even if the full design is not NA).

Consider a population $\mathcal{U}_N = \bigcup_{j=1}^{M} \mathcal{U}_j$ where the $\mathcal{U}_j,\ j = 1, \ldots, M$ are disjoint (and are called primary units). The size of each $\mathcal{U}_j$ is N_j. The two stage sampling design $p_{N,2d}$ consists in first selecting m units among the M elements of $\mathcal{U}$ using a design $p_{M,I}$, which we require to be NA. As seen in part 5.1.1, this is not a very restrictive hypothesis since many sampling designs used in practice are NA. The obtained sample is denoted s^I. Then, among each one of the $j \in [\![1, m]\!]$ primary units in s^I independent sampling algorithms p_{M,II_j} are run. The samples selected at the second stage are denoted s_j^{II} and we also write: $s = \bigcup_{j=1}^{m} s_j^{II}$. Finally we write $\forall j \in [\![1, M]\!],\ \eta_j^I = \mathbb{1}\{j \in s^I\},\ \forall j, \forall k \in [\![1, N_j]\!],\ \eta_{jk}^{II} = \mathbb{1}\{k \in s_j^{II}\}$ and $\forall i \in \mathcal{U},\ \eta_i = \mathbb{1}\{i \in s\}$. All these definitions can be easily generalized to d-degree sampling for any $d > 2$.

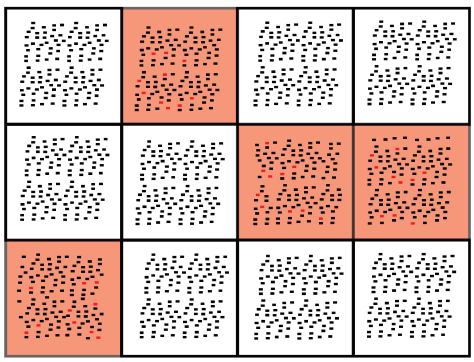


Fig. 5.1: Selection of a sample of size $n = 40$ in a population of units (black dots) divided in $M = 12$ primary units of equal size $N_j = 64,\ j = 1, ..., M$ ($N = 768$) with a two-stage design. First, a sample of $m = 4$ first-degree units are selected between the $M = 12$ with simple random sampling (in pink). Then in each of theses 4 selected primary units, 10 units of the population are selected (red dots), again via simple random sampling.

Now, consider in each strata, $\hat{Y}_{HT,j}$ the Horvitz-Thompson estimator of $T(Y)_j = \sum_{k\in\mathcal{U}_j} y_k$ (using data from s_j^{II}). then $\widehat{T}$, the Horvitz-Thompson estimator of $T(Y)$ using data from the whole sample is given by

$$\widehat{T} = \sum_{j=1}^{M} \eta_j^I \frac{1}{\pi_j^I} \hat{Y}_{HT,j}$$

with

$$\hat{Y}_{HT,j} = \sum_{k=1}^{N_j} \frac{\eta_k^{II,j}}{\pi_k^{II,j}} Y_k$$

The following results is inspired from the one developed in [Mir+15] for obtaining Edgeworth expansions for multi-stage samplings. It can easily be carried out for d-degree sampling ($d > 2$).

Theorem 11. *If all the sampling designs $p_{M,I}$ and p_{M,II_j} for all $j = 1, \ldots, J$ are NA and two-by-two independent, then the two-stage design $p_{N,2d}$ may not be NA. However, the r.v. $\eta_j^I \hat{Y}_{HT,j} \quad j = 1, ..., M$ satisfy the NA properties and the Horvitz Thompson estimator can be seen as a sum M NA random variables.*

Proof: We have $\forall i \in \mathcal{U}, \eta_i = \eta_{j(i)}^I \eta_{j(i),i}^{II}$ where $j(i)$ is the index of the primary unit containing unit i. Note that the variable η_i is not necessarily NA (see the example below), even if all designs involved $R_{M,I}$ and R_{N,II_j} are NA. Now consider the rv's $X_j = \eta_j^I \hat{Y}_{HT,j}, \ j = 1, ..., M$. Write A and B two disjoint subsets of $\mathcal{U}$ and $X_A = (X_j)_{j\in A} \underset{not.}{=} \eta_A^I \hat{Y}_{HT,A}$ and $X_B = (X_j)_{j\in B} \underset{not.}{=} \eta_B^I \hat{Y}_{HT,B}$ the vectors valued on respectively A and B. Let f: $\mathbb{R}^A \to \mathbb{R}$ and $g : \mathbb{R}^B \to \mathbb{R}$ be two increasing functions. Then we have:

$$\begin{aligned}
&E\left[f(X_A)g(X_B)\right] \\
&= E\left[f(\eta_A^I \hat{Y}_{HT,A}) g(\eta_B^I \hat{Y}_{HT,B})\right] \\
&= E\left[E\left(f(\eta_A^I \hat{Y}_{HT,A}) g(\eta_B^I \hat{Y}_{HT,B}) | \hat{Y}_{HT,A}, \hat{Y}_{HT,B}\right)\right] \\
&\leq E\left[E\left(f(\eta_A^I \hat{Y}_{HT,A}) | \hat{Y}_{HT,A}, \hat{Y}_{HT,B}\right) E\left(g(\eta_B^I \hat{Y}_{HT,B}) | \hat{Y}_{HT,A}, \hat{Y}_{HT,B}\right)\right] \\
&\text{(because } \eta^I \text{ is NA and } Y \text{ is positive)} \\
&= E\left[E\left(f(\eta_A^I \hat{Y}_{HT,A}) | \hat{Y}_{HT,A}\right) E\left(g(\eta_B^I \hat{Y}_{HT,B}) | \hat{Y}_{HT,B}\right)\right] \\
&\text{(by independence of the sampling plans in each strata)} \\
&= E\left[E\left(f(\eta_A^I \hat{Y}_{HT,A}) | \hat{Y}_{HT,A}\right)\right] \ E\left[E\left(g(\eta_B^I \hat{Y}_{HT,B}) | \hat{Y}_{HT,B}\right)\right] \\
&\text{(by independence)} \\
&= Ef(X_A) Eg(X_B).
\end{aligned}$$

It follows by that $\widehat{T} = \sum_{j=1}^{M} \frac{X_j}{\pi_j^I}$ is a sum of negatively associated random variables.

Example (two-stage designs are not NA): Consider a two stage sampling design where the sampling design at the first stage p^I and all the other sampling designs within the strata p_j^{II} are simple random sampling with respective sampling sizes m (among M) and n_j (among N_j). A particular case of two stage sampling is cluster sampling when all the units from the strata are selected at the second stage ($n_j = N_j$). For $k \in \mathcal{U}$ we denote (k) the index of the first degree unit (in $\mathcal{U}^I$) containing unit k. Then the first and second order inclusion probabilities are given by

$$\pi_k = \frac{m}{M}\frac{n_{(k)}}{N_{(k)}}$$

and

$$\Delta_{kl} = \pi_{kl} - \pi_k\pi_l$$
$$= \begin{cases} \frac{m}{M}\frac{n_{(k)}}{N_{(k)}}\left(\frac{n_{(k)}-1}{N_{(k)}-1} - \frac{m}{M}\frac{n_{(k)}}{N_{(k)}}\right) > 0 \text{ if k and l are in the same first degree unit} \\ \frac{m}{M}\frac{n_{(k)}}{N_{(k)}}\frac{n_{(l)}}{N_{(l)}}\left(\frac{m-M}{M(M-1)}\right) < 0 \text{ if k and l are not in the same first degree unit} \end{cases}.$$

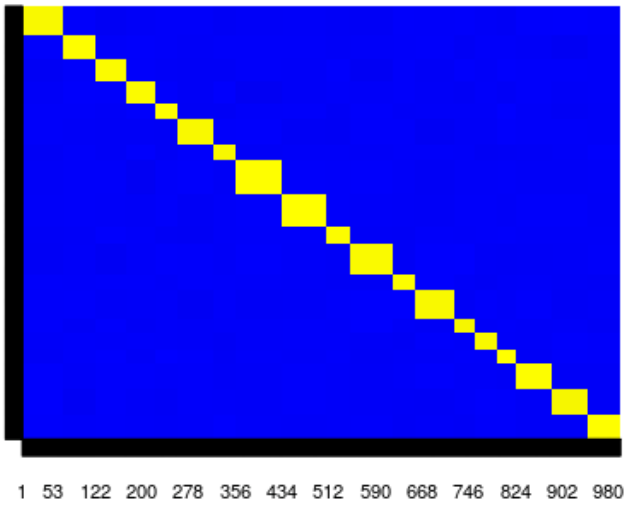


Fig. 5.2: Matrix of the $\Delta_{ij}, i, j \in [\![1, 1000]\!]$ with M = 20, m = 5, $n_0 = 10$. The positive coefficients are shown in yellow, the negative ones in blue.

Figure 5.2 illustrates that the two-stage design itself is not NA but using 11 it will still possible to get CLT (at least when the number of strata $M \to \infty$). This will the subject of further work.

5.3 A tailor-made CLT for negatively associated sampling plans

For sake of completeness, we recall the following theorem from [Pat+01] (see their Theorem 2.4 with an elegant three lines proof). This result is also used in [BJ12] to advocate that the CLT may be obtained for many sampling plans with the help of negative correlation properties.

Theorem 12. *Consider a triangular array $(Z_{i,N})_{1\leq i\leq N}$ of centered negatively associated random variables then, under the conditions*

(i) $S_N^2 = Var(\sum_{i=1}^N Z_{i,N}) \to \infty$ *as* $N \to \infty$,
(ii) $\frac{1}{S_N^2}\sum_{i=1}^N \sum_{i<j} cov(Z_{i,N}, Z_{j,N}) \to 0$ *as* $N \to \infty$,
(iii) for any $\varepsilon > 0$, $\frac{1}{S_N^2}\sum_{i=1}^N E(Z_{i,N}^2 \mathbb{I}_{\{|Z_{i,N}|>\varepsilon S_N}) \to 0$ *as* $N \to \infty$,
we have

$$\frac{1}{S_N}\sum_{i=1}^N Z_{i,N} \underset{N\to\infty}{\overset{d}{\to}} N(0,1).$$

Remark 1: This theorem yields immediately a CLT for most exchangeable resampling plans. Consider for instance the mooN bootstrap (sampling with replacement in term of survey sampling).Then $(W_{i,N})_{1\leq i\leq N}$ follows the multinomial distribution $Mult(m, 1/N, ..., 1/N))$ and we have

$$\begin{aligned} E(W_{i,N}) &= \frac{m}{N} \\ V(W_{i,N}) &= \frac{m}{N}(1-1/N) \\ cov(W_{i,N}, W_{j,N}) &= -V(W_{i,N})/(N-1) = -m/N^2 \end{aligned}$$

It follows that if we consider the weighted bootstrap sums with $Z_{i,N} = (W_{i,N} - \frac{m}{N})Y_i$ where the Y_i $'s$ are positive i.i.d. r.v.'s with $E(Y_i^{2+\eta}) < \infty$ for some $\eta > 0$, then the $Z_{i,N}$ $'s$ are negatively associated and we get

$$\begin{aligned} \sum_{i=1}^N V(Z_{i,N}) &= \sum_{i=1}^N (Y_i - \overline{Y}_N)^2 V(W_{i,N}) \\ &= m\sigma^2(1-1/N)(1+o(1)) \to \infty \text{ , a.s. as soon as } n \to \infty, \text{ and} \\ \sum_{i=1}^N \sum_{i<j} cov(Z_{i,N}, Z_{j,N}) &= -m\frac{1}{N^2}\sum_{i=1}^N \sum_{j=1,\ i\neq j}^N (Y_i - \overline{Y}_N)(Y_j - \overline{Y}_N) \\ &= -m\left(\left(\frac{1}{N}\sum_{i=1}^N (Y_i - \overline{Y}_N)\right)^2 + \frac{1}{N^2}\sum_{i=1}^N (Y_i - \overline{Y}_N)^2\right) = o(m) \text{ a.s.} \end{aligned}$$

Then conditions (i), (ii) are satisfied. More other (iii) is implied by the easier condition $E(Y_i^{2+\eta}) < \infty$. The validity of the mooN bootstrap follows immediately for any size m. We have

$$\sqrt{m}\frac{\frac{1}{m}\left(\sum_{i=1}^N W_{i,N}Y_i\right) - \overline{Y}_N)}{\sigma_N} =$$
$$\frac{1}{\sqrt{m}}\frac{\sum_{i=1}^N (W_{i,N} - \frac{m}{N})(Y_i - \overline{Y}_N)}{\sigma_N} \to N(0,1) \ \ a.s. \text{ as } n \to \infty,$$

where

$$\sigma_N^2 = \frac{1}{N}\sum_{i=1}^N (Y_i - \overline{Y}_N)^2.$$

The same proof may be used for the Bayesian bootstrap and bootstrap Bayesian Clones (based on i.i.d rv's with log-concave density). This is a simple way to recover well known results in a straightforward way (see [BB95]).

Remark 2: However this theorem is not suitable for proving a central limit for most interesting survey sampling plans with unequal probabilities. Indeed even in the case of sampling without replacement or rejective sampling, condition (ii) does not hold. Notice that we have

$$Var(\sum_{i=1}^N Z_{i,N}) = \sum_{i=1}^N Var(Z_{i,N}) + \frac{2}{S_N^2}\sum_{i=1}^N \sum_{i<j} cov(Z_{i,N}, Z_{j,N})$$

It can be seen immediately that the condition (ii) implies that

$$\sum_{i=1}^N Var(Z_{i,N})/S_N^2 = 1 + o(1)$$

so that the main results of the theorem would imply that

$$\frac{1}{(\sum_{i=1}^N Var(Z_{i,N}))^{1/2}}\sum_{i=1}^N Z_{i,N} \underset{N\to\infty}{\overset{L}{\to}} N(0,1).$$

In the particular case of a survey sampling, if we put $Z_{i,N} = (\frac{\eta_i}{\pi_i} - 1)Y_i$ we get $\sum_{i=1}^N Var(Z_{i,N}) = \sum_{i=1}^N \frac{Y_i^2}{\pi_i}(1-\pi_i)$ which is essentially the variance of a Poisson sampling plan. But this variance is incorrect for rejective sampling with unequal (or even equal) probabilities, see [Háj64]. A straightforward calculus shows actually that $\frac{2}{S_N^2}\sum_{i=1}^N \sum_{i<j}^N cov(Z_{i,N}, Z_{j,N})$ does not converge to 0 in this case.This problem appears for most sampling plans (except the Poisson case).

Many CLT's have been obtained for negatively associated r.v.'s however, most of them assume stationarity which is not the case in survey sampling framework. For this

reason, we give the following theorem tailored to survey sampling based on some previous arguments by [PU97]. It shows that for negatively associated triangular arrays, that the condition (ii) may be totally removed under some simple additional assumptions. This is due to the fact that negative association (in opposition to positive association) allows to control directly the sums of the covariances which are automatically convergent as shown by the following lemma.

Lemma 13. *Consider a triangular array $(Z_{i,N})_{1\leq i\leq N}$ of centered negatively associated random variables. Assume that*

$$(iv)\ 0 < \lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} Var(Z_{i,N}) < \infty,$$

then, for any $i \in \mathbb{N}$,as $N \to \infty$,we have

$$\lim_{N\to\infty} \left| \sum_{j=i}^{N} cov(Z_{i,N}, Z_{j,N}) \right| < \infty,$$

which in turn implies that, for any $l_N \to \infty$,

$$\sup_{i\in\mathbb{N}} \left| \sum_{|j-i|>l_N} cov(Z_{i,N}, Z_{j,N}) \right| \to 0.$$

Proof: We have

$$0 \leq \frac{S_N^2}{N} = \frac{1}{N}\sum_{i=1}^{N} Var(Z_{i,N}) + \frac{2}{N}\sum_{i=1}^{N}\sum_{i<j\leq N} cov(Z_{i,N}, Z_{j,N}).$$

since by negative association $\sum\limits_{i=1}^{N}\sum_{i<j} cov(Z_{i,N}, Z_{j,N}) \leq 0$. It follows that we have the convergence of the following sums with

$$\left| \frac{1}{N}\sum_{i=1}^{N}\sum_{i<j\leq N} cov(Z_{i,N}, Z_{j,N}) \right| = \frac{1}{N}\sum_{i=1}^{N}\sum_{i<j\leq N} |cov(Z_{i,N}, Z_{j,N})| \leq \frac{1}{N}\sum_{i=1}^{N} Var(Z_{i,N}) < \infty.$$

By the reciprocal of the generalized Cesaro theorem, since $U_{i,N} = \sum_{j:\ i<j}^{N} |cov(Z_{i,N}, Z_{j,N})|$ is decreasing in i, $U_{i,N}$ converges to the same limit, so that the tail behavior of this sum is also controlled uniformly in i.

■

We are now ready to state a CLT for (non stationary) negatively associated arrays.

Theorem 14. *Consider a triangular array $(Z_{i,N})_{1\leq i\leq N}$ of centered negatively associated random variables such that*
(i) the Lindeberg-Feller conditions (iii) of Theorem 12 holds.
(ii) S_N^2 correctly standardized converges at least in probability to a limit such that

$$0 < \lim_{N\to\infty} \frac{S_N^2}{N} < \infty.$$

Then we have

$$\frac{1}{S_N}\sum_{i=1}^{N} Z_{i,N} \underset{N\to\infty}{\overset{L}{\to}} N(0,1).$$

Proof: The proof combined the blocking arguments of [PU97] and some arguments from the CLT proved in [Oli12]. The slight difference is the following: instead of considering an infinite sequence of r.v.'s as in [PU97], we are dealing with sequences of arrays of random variables. Moreover since our variables are negatively associated, we do not need their condition 2.4, which is automatically satisfied in our case by lemma 13. For sake of completeness, we give a short proof of the result. As noticed by [Oli12] (see his Theorem 4.8 p. 108) under the Lindeberg-Feller condition (iii) of Theorem 12, there exists a sequence $l_N \to \infty$ with $\frac{l_N^2}{N} \to 0$ such that for any $\varepsilon > 0$,

$$\frac{l_N^2}{S_N^2}\sum_{i=1}^{N}\int_{|Z_{i,N}|>\varepsilon S_N/l_N} z^2 P(dz) \to 0 \text{ as } N\to\infty.$$

Notice also that similarly to the proof of [PU97] p. 452 (take $a_{Ni} = 1\sqrt{N}$ and $u = l_N$) and using lemma 13, we have for this l_N,as $N \to \infty$,

$$\left|\frac{1}{N}\sum_{i=1}^{N-l_N}\sum_{i+l_N<j\leq N} cov(Z_{i,N}, Z_{j,N})\right| \to 0. \tag{5.2}$$

Now define the sums of non-overlapping blocks of size l_N given by

$$Y_{j,l_N} = \sum_{i=(j-1)l_N+1}^{jl_N} Z_{i,N}, \quad j = 1, ..., k_N,$$

with

$$k_N = [\frac{n}{l_N}].$$

Then consider the following slight modification of the construction in [PU97]. From (5.2), for any $\varepsilon > 0$ there exists N such

$$\left| \frac{1}{N} \sum_{i=1}^{N-l_N} \sum_{i+l_N<j\leq N} cov(Z_{i,N}, Z_{j,N}) \right| \leq \varepsilon.$$

Then put $K = [1/\varepsilon]$ and define the non empty sets

$$A_j = \{i,\ 2Kj \leq i \leq 2Kj + K, |cov(Y_{i,N}, Y_{i+1,l_N})| \leq 2/K \sum_{i=2Kj}^{2Kj+K} var(Y_{i,N}).$$

Define recursively the sequence of integers, $m_{j,},\ j = 1, ...N$ as follows:

$$\begin{aligned} m_0 &= 0 \\ m_{j+1} &= \min\{m, m > m_j, m \in A_j\} \end{aligned}$$

and the corresponding sets

$$\Delta_j = \{u(m_j + 1), ..., u(m_{j+1})\}.$$

Now define the sums of block of blocks (the construction ensures that we capture the covariances)

$$B_{j,l_N} = \sum_{i=m_j+1}^{m_{j+1}} Y_{i,l_N} = \sum_{i\in\Delta_j} Z_{i,N}.$$

By the permanence property of association over positive weighted sums, these r.v.'s are still negatively associated. Moreover, as noticed in [PU97], the sets have at most $3Kl_N$ elements. As a consequence the blocks B_{j,l_N} still satisfy the Lindeberg-Feller assumption (i) because of our choice of l_N.

This construction is precisely built to ensure that we have in probability,

$$\frac{1}{N} \sum_{j=1}^{k_N} Var(B_{j,l_N}) = \frac{1}{k_N} \sum_{j=1}^{k_N} Var(B_{j,l_N})/l_N \to \lim_{N\to\infty} \frac{S_N^2}{N} < \infty. \tag{5.3}$$

Now recall that by Theorem 1 of [New80], we have the inequality

$$\left| E \exp\left(\frac{i \sum_{j=1}^{N} Z_{j,N}}{S_N} \right) - \prod_{j=1}^{m_N} E \exp\left(it \frac{B_{j,l_N}}{S_N} \right) \right| \leq \frac{2t^2}{S_N^2} \sum_{j=1}^{m_N-1} \sum_{k=j+1}^{m_N} |cov(B_{j,l_N}, B_{k,l_N})| \to 0 \tag{5.4}$$

because we have by (5.3)

$$\frac{1}{S_N^2}\sum_{j=1}^{m_N} Var(B_{j,l_N}) - 1 = \frac{1}{S_N^2}\sum_{j=1}^{m_N-1}\sum_{k=j+1}^{m_N} |cov(B_{j,l_N}, B_{k,l_N})| \to 0$$

Now use exactly the same arguments as [Oli12, p. 109] to prove that the Z_{j,l_N} also satisfy the Lindeberg condition and apply theorem 4.4 of [Oli12, p. 105]. Indeed (5.4) essentially means that the sum $\frac{1}{S_N}\sum_{j=1}^{m_N} Z_{j,N}$ behaves like the sum of independent copies of the blocks $B_{j,N}$ say $B^*_{j,N}$ standardized by their true variance:

$$\frac{\sum_{j=1}^{m_N} B^*_{j,l_N}}{S_N} = \left(\frac{1}{S_N^2}\sum_{j=1}^{m_N} Var(B_{j,l_N})\right)^{1/2} \frac{\sum_{j=1}^{m_N} B^*_{j,l_N}}{\left(\sum_{j=1}^{m_N} Var(B^*_{j,l_N})\right)^{1/2}}.$$

Since these blocks satisfy the Lindeberg-Feller condition, the right hand side of this expression is asymptotically normal.■

We are now ready to establish a CLT adapted to survey sampling plan. When there is no ambiguity we omit the indexes N and n in all the inclusion probabilities and all the auxiliary variables, that is, we write X_i instead of $X_{i,N,n}$ and similarly π_i for $\pi_{i,N,n}$. One should recall that, when we are considering asymptotic limit

in survey sampling, we are dealing with triangular arrays of variables/inclusion probabilities.

Assumption 1. *The first-order inclusion probabilities $\pi = (\pi_1, \ldots, \pi_N)$ depend on some auxiliary variables X through the link function $p : \mathcal{X} \to [p_\star, 1]$, $p_\star > 0$. For this link function, we will thus write*

$$\pi_i = p(X_i)$$

with $\sum_{i=1}^{N} p(X_i) = n.$

Because the variance of the Horvitz-Thompson estimator also depends on the second order inclusion probability, we also assume the following

Assumption 2. *There exists some symmetric bounded positive function $0 < h(.,.) \leq K$, such that $\pi_i\pi_j - \pi_{i,j} = \frac{h(X_i,X_j)}{N}$, $i \neq j$.*

As a consequence, we have

$$|\pi_{i,j} - \pi_i\pi_j| \leq \frac{K}{N}.$$

We also extend the function h at pairs by putting

$$h(X_i, X_i)/N = \pi_i\pi_i - \pi_{ii} = -p(X_i)(1 - p(X_i)).$$

Then recall that, for a fixed survey sampling size n, we have the following properties

$$\sum_{j=1}^{N} \pi_{i,j} = n\pi_i$$

implying that

$$\sum_{i=1}^{N} \frac{h(X_i, X_j)}{N} = \sum_{j=1}^{N} \frac{h(X_i, X_j)}{N} = 0 \text{ for any } i.$$

Moreover we have

$$\sum_{i=1}^{N}\sum_{j\neq 1}^{N} \pi_{i,j} = n(n-1) \text{ and } \sum_{i=1}^{N}\sum_{j=1}^{N} \pi_{i,j} = n^2,$$

yielding

$$\sum_{i=1}^{N}\sum_{j=1}^{N} (\pi_i\pi_j - \pi_{i,j}) = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{h(X_i, X_j)}{N} = 0.$$

With these assumptions the Horvitz-Thompson estimator becomes

$$\widehat{T} = \sum_{k\in\mathcal{U}} \frac{Y_k}{p(X_k)}\eta_k$$

Notice that for a survey sampling with fixed size we have the Sen-Yates-Grundy form of the variance given by

$$\begin{aligned} V(\widehat{T}) &= \frac{1}{2N}\sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N} (\pi_i\pi_j - \pi_{i,j})\left(\frac{Y_i}{\pi_i} - \frac{Y_j}{\pi_j}\right)^2 \\ &= \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N} h(X_i, X_j)\left(\frac{Y_i}{p(X_i)} - \frac{Y_j}{p(X_j)}\right)^2. \end{aligned}$$

Under the preceding assumptions we have

$$V(\widehat{T}) \leq \frac{K}{2}\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N}\left(\frac{Y_i}{p(X_i)} - \frac{Y_j}{p(X_j)}\right)^2.$$

The right-hand side of this inequality is convergent at least in probability provided that $\frac{1}{N}\sum_{i=1}^{N}\left(\frac{Y_i}{p(X_i)}\right)^2$ is convergent. This is the case for instance if the $(Y_i, X_i), i = 1, ...N$ form a set of exchangeable variables and $E\left(\frac{Y_i^2}{p(X_i)^2}\right) < \infty$ but this is also true for many ergodic generating processes. To avoid complicated assumption, we essentially assume

Assumption 3. *The random vectors* $(Y_1, X_1), \ldots, (Y_N, X_N)$ *are exchangeable random vectors (distributionally invariant by permutation) with common marginal distribution* $P_{Y,X}$*, such that* $E_{P_{Y,X}} \frac{{Y_i}^2}{p(X_i)^2} < \infty$.

Notice that assumption 3 is satisfied if $E_{P_Y} {Y_i}^2 < \infty$, since by assumption 1, the function $p()$ is lower bounded. Actually if $p(X)$ is strongly correlated with Y (the strongest form being full correlation $Y = \alpha p(X),\ \alpha \neq 0$) then the moment condition in assumption 3 is weaker.

Theorem 15. *Let* $(\eta_1, ..., \eta_N)$ *be a array of negatively associated r.v.'s, defining a survey sampling plans, satisfying assumptions 1, 2, 3. Assume in addition that the following Lindeberg condition holds:*

(i) for any $\varepsilon > 0$, $\frac{1}{S_N^2} \sum_{i=1}^{N} E_\eta \left(\left| \frac{\eta_k}{\pi_k} - 1 \right| |Y_k| \mathbb{I}_{\left\{ \left| \frac{\eta_k}{\pi_k} - 1 \right| |Y_k| > \varepsilon S_N \right\}} \right) \xrightarrow{pr.} 0$*, as* $N \to \infty$.

(ii) Assume that $V(\widehat{T})\ /N$ *converges at least in probability to a limit such that*

$$0 < \lim_{N \to \infty} \frac{V(\widehat{T})}{N} < \infty$$

then in probability (along the sequence Y), we have

$$\frac{\widehat{T} - T(Y)}{V(\widehat{T})^{1/2}} \underset{n,N \to \infty}{\xrightarrow{d}} N(0,1).$$

Proof: Put $Z_{i,N} = (\frac{\eta_i}{p(X_i)} - 1) Y_i$ then by permanence of the negative association property, the $Z_{i,N},\ i = 1, ..., N$ form an array of negatively associated r.v's. Thus to prove the CLT, we essentially have to check the hypotheses of theorem 14 . Since the Lindeberg condition and the condition on the variance are directly satisfied under the assumptions of the theorem, apply directly 14 to get the result.

5.4 Functional central limit theorems for negatively associated survey sampling

For many applications in survey sampling it is important to obtain Donsker-type central limit theorems over some specific classes of functions. For instance, this allows to obtain asymptotic distribution for the empirical cdf, which may be used to get uniform confidence band for the cdf, but also for many functionals having smooth properties (Fréchet or Hadamard differentiable in some adequate space): see [Boi+17]. The most general framework is classically the one used in [VW96].

Several partial results have been recently obtained in the framework of survey sampling either by considering particular cases of empirical processes or particular type of surveys ([Ber+16], [Boi+17]). In the following paragraph, we show how it is possible to generalize the results of [Ber+16] proved in the framework of Poisson sampling or conditional Poisson sampling to much more general sampling plans. For this we recall some of their notations before stating the main results.

5.4.1 Framework for empirical processes in survey sampling (single stage)

We essentially use the same notations as in [Ber+16]. Let $(Y_1, ..., Y_N)$ be the vector of characteristics on the whole population taking their values on a measurable space $(\mathcal{Y}, \mathcal{A})$. We denote by P_N the *empirical measure*

$$P_N = \frac{1}{N} \sum_{i=1}^{N} \delta_{Y_i}$$

where

$$\delta_{Y_i}(A) = \begin{cases} 1 \text{ if } Y_i \in A \\ 0 \qquad \text{else} \end{cases}, \ A \in \mathcal{A}.$$

is the Dirac measure. For any signed measure $\mathbb{Q}$ on a measurable space $(\mathcal{Y}, \mathcal{A})$ and any $\mathbb{Q}$-integrable function $f : \mathcal{Y} \to \mathbb{R}$, we use the operator notation to design the mean of the function f under $\mathbb{Q}$:

$$\mathbb{Q}f := \int_{\mathcal{Y}} f(y)\, \mathbb{Q}(dy).$$

Then we have by definition

$$P_N f = \frac{1}{N} \sum_{i=1}^{N} f(Y_i)$$

Then consider a class $\mathcal{F}$ of P-integrable real-valued functions f .It is assumed as usual that the class $\mathcal{F}$ admits a square integrable envelop, that is, there exists a measurable function $H : \mathcal{Y} \to \mathbb{R}$ such that $|f(y)| \leq H(y)$ for all $y \in \mathcal{Y}$ and any $f \in \mathcal{F}$ with

$$\int_{\mathcal{Y}} H^2(y)\, P(dy) < \infty.$$

Then the class $\mathcal{F}$ may be seen as a subset of the space

$$L_2(P) := \left\{ f : \mathcal{Y} \to \mathbb{R}, \quad f \text{ measurable with } \|f\|_{2,P}^2 := E_P\left(f^2(Y)\right) < +\infty \right\}.$$

The usual empirical process indexed by the class of functions $\mathcal{F}$ is then defined by the sequence

$$\mathbb{G}_N f := \sqrt{N}\,(P_N - P)\,f, \text{ over } f \in \mathcal{F}. \tag{5.5}$$

In the i.i.d. case, assuming that the $(Y_1, ..., Y_N)$ are i.i.d. r.v's with common distribution P, the behaviour of $\{\mathbb{G}_N f\}_{f\in\mathcal{F}}$ and conditions on the class of functions ensuring that the process converges to a tight limit are now well understood (see [VW96])

Recall that the map $G_N : f \in \mathcal{F} \mapsto \mathbb{G}_N f \in \mathbb{R}$ may be seen as an element of $\ell^\infty(\mathcal{F})$, the space of maps $G : \mathcal{F} \to \mathbb{R}$ with bounded norm, when equipped with the uniform convergence norm defined by

$$||G||_{\mathcal{F}} = \sup_{f\in\mathcal{F}} |Gf|.$$

The convergence of the empirical process (5.5) essentially depends on the size of the class of functions measured by some entropy measure. For this, define the ρ_P-semimetric under P as, for all $(f, g) \in \mathcal{F}^2$

$$\rho_P(f, g) := E_P\left((f(Y) - g(Y))^2\right) =: \|f - g\|^2_{2,P}.$$

We denote by $\mathcal{N}(\varepsilon, \mathcal{F}, ||.||^2_{2,P})$ the minimal number of balls of radius $\varepsilon > 0$ for a given semimetric $||.||^2_{2,P}$ needed to cover $\mathcal{F}$, referred to as the *covering number*.

Define $BL_1(\ell^\infty(\mathcal{F}))$ the set of 1-Lipschitz functions on $\ell^\infty(\mathcal{F})$ bounded by 1. Using Hoffman-Jørgensen weak convergence ([HJ91]) (in outer expectation), weak convergence is metrized by the bounded Lipschitz metric on the space $\ell^\infty(\mathcal{F})$, defined for all random functions G_1 and G_2 valued in $\ell^\infty(\mathcal{F})$ as

$$d_{BL}(G_1, G_2) = \sup_{b\in BL_1(\ell^\infty(\mathcal{F}))} \left|E(b(G_1)) - E(b(G_2))\right|.$$

In empirical process theory, $\mathcal{F}$ is said to be Donsker iff the process $G_N := \{\mathbb{G}_N f,\ f \in \mathcal{F}\}$ converges to a tight Gaussian process $\mathbb{G}$, with covariance operator $(cov(f, g), f \in \mathcal{F}, g \in \mathcal{F})$ and continuous path in $\ell^\infty(\mathcal{F})$ iff

$$d_{BL}(G_N, \mathbb{G}) \to 0 \text{ as } N \to \infty.$$

5.4.2 Convergence of the HT-empirical process

Define now the *Horvitz-Thompson (HT) empirical measure* by

$$P^{\boldsymbol{\pi}}_{R_N} := \frac{1}{N}\sum_{i=1}^{N}\frac{\eta_i}{\pi_i}\,\delta_{Y_i}.$$

and the corresponding *HT empirical process* indexed by $\mathcal{F}$,

$$\mathbb{G}^{\boldsymbol{\pi}}_{R_N} f := \sqrt{N}\,\left(P^{\boldsymbol{\pi}}_{R_N} - P_N\right) f = \frac{1}{\sqrt{N}}\sum_{i=1}^{N}\left(\frac{\eta_i}{\pi_i} - 1\right) f(Y_i),\ f \in \mathcal{F}. \tag{5.6}$$

Under the envelop conditions, $\mathbb{G}^{\boldsymbol{\pi}}_{R_N}$ may also be considered as an element of $\ell^\infty(\mathcal{F})$. In [Ber+16], the authors gave some conditions for the *HT empirical process* indexed by $\mathcal{F}$ to be Donsker, when the survey sampling plan $(\eta_1, \eta_2, ..., \eta_N)$ has a distribution R_N either of Poisson type or Rejective type. They extend their results to survey sampling plans close to rejective sampling. Here we will generalize their results to negatively associated survey sampling plans satisfying the assumptions (1), (2) and (3) of the preceding part. For this, we will assume that the functions $f \in \mathcal{F}$ are positive functions. This is by no means restrictive since we may always decompose the process into two parts (which are orthogonal)

$$\mathbb{G}^{\boldsymbol{\pi}}_{R_N} f = \mathbb{G}^{\boldsymbol{\pi}}_{R_N} f^+ - \mathbb{G}^{\boldsymbol{\pi}}_{R_N} f^-,$$

with $f^+ = \max(f, 0)$ and $f^- = \max(-f, 0)$.

Under the assumption (1), (2) and (3), for a survey sampling with fixed size we have the Sen-Yates-Grundy form of the variance given by

$$\begin{aligned} S^2_N(f) = V(\mathbb{G}^{\boldsymbol{\pi}}_{R_N} f) &= \frac{1}{2N}\sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N}(\pi_i\pi_j - \pi_{i,j})\left(\frac{f(Y_i)}{\pi_i} - \frac{f(Y_j)}{\pi_j}\right)^2. \\ &= \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N} h(X_i, X_j)\left(\frac{f(Y_i)}{p(X_i)} - \frac{f(Y_j)}{p(X_j)}\right)^2, \end{aligned}$$

yielding

$$S^2_N(f) \leq \frac{K}{2}\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1, i\neq j}^{N}\left(\frac{f(Y_i)}{p(X_i)} - \frac{f(Y_j)}{p(X_j)}\right)^2.$$

The right hand side of this equation is convergent and thus bounded a.s. provided that $E\frac{f(Y_i)^2}{p(X_i)^2} < \infty$. In [Ber+16] it is assumed that the $(Y_i, X_i), i = 1, ..., N$ are i.i.d. However as noticed in their paper, exchangeability ensuring that some law of large numbers for some sums appearing in covariance operator holds, is sufficient. Actually one only needs the ergodicity of the sequence (Y_i, X_i) , $i = 1, ..., N$. This surprising result is essentially due to the fact that survey sampling artificially destroys the

dependence between the individuals. As a consequence, survey sampling is very robust to the data generation mechanism. This is one of the main difference with resampling procedures : we do not try to reproduce the distribution of the empirical process on the original data (a task which may be complicated because of the dependence structure) but only try to get a limit conditionally to the data (in probability or a.s. along the sequence of the data).

Thus we assume either

Assumption 4. *The random vectors* $(Y_1, X_1), \ldots, (Y_N, X_N)$ *are exchangeable random vectors with common marginal distribution* $P_{Y,X}$*, with* $E_{P_{Y,X}} \frac{H(Y_i)^2}{p(X_i)^2} < \infty$.

or the higher level assumption

Assumption 5. *The data generating mechanism is such that*

$$\frac{1}{N^2} \sum_i \sum_{j \neq i} \frac{h(X_i, X_j)}{p(X_i)p(X_j)} f(Y_i)g(Y_j)$$

has a limit (in probability) given by

$$E_{P_{Y,X}^{\otimes 2}} \left(\frac{h(X_i, X_j)}{p(X_i)p(X_j)} f(Y_i)g(Y_j) \right), \ i \neq j.$$

And $\frac{1}{N} \sum_i \frac{(1-p(X_i))}{p(X_i)} f(Y_i)g(Y_i)$ *converges in probability to* $E_{P_{Y,W}} \left(\frac{(1-p(X_i))}{p(X_i)} f(Y_i)g(Y_i) \right)$.

Under this condition, define the covariance operator, for $(f, g) \in \mathcal{F}^2$

$$\Sigma^p(f, g) := E_{P_{Y,X}} \left(\frac{(1 - p(X_i))}{p(X_i)} f(Y_i)g(Y_i) \right) - E_{P_{Y,X}^{\otimes 2}} \left(\frac{h(X_i, X_j)}{p(X_i)p(X_j)} f(Y_i)g(Y_j) \right).$$

Using the preceeding identities on h, we also have the Sen-Yates-Grundy form

$$\Sigma^p(f, g) = \frac{1}{2} E_{P_{Y,X}^{\otimes 2}} \left(h(X_i, X_j) \left(\frac{f(Y_i)}{p(X_i)} - \frac{g(Y_j)}{p(X_j)} \right)^2 \right).$$

Indeed under assumptions (1) and (2), $\Sigma^p(f,g)$ is simply the limit as $N \to \infty$ of the conditional (to the $(Y_i, X_i)'s$) covariance of the empirical process

$$cov_{R_N}(\mathbb{G}^{\boldsymbol{p}}_{R_N} f, \mathbb{G}^{\boldsymbol{p}}_{R_N} g) = \frac{1}{N}\sum_i \sum_j cov\left(\left(\frac{\eta_i}{p(X_i)} - 1\right)\left(\frac{\eta_j}{p(X_j)} - 1\right)\right) f(Y_i)g(Y_j) \tag{5.7}$$

$$= -\frac{1}{N^2}\sum_i \sum_j \frac{h(X_i, X_j)}{p(X_i)p(X_j)} f(Y_i)g(Y_j)$$

$$= \frac{1}{2N^2}\sum_{i=1}^N \sum_{j=1}^N h(X_i, X_j)\left(\frac{f(Y_i)}{p(X_i)} - \frac{g(Y_j)}{p(X_j)}\right)^2.$$

Notice that, in practice, the weights may be chosen proportional to some weighted sums of auxiliary random vectors such that $(Y_k,\ Z_k = (Z_{jk})_{,j\in[\![1,J]\!]}), k = 1, ..., N$ are i.i.d. Typically, we will choose

$$X_k = \frac{n}{N}\frac{\theta' Z_k}{\frac{1}{N}\sum_{k=1}^N \theta' Z_k},\ k = 1, ..., N,$$

for some parameter $\theta \in \mathbb{R}^J$ and $p(x) = x$.In that case by exchangeability of the sum, assumption 4 is automatically satisfied. In the general case with dependent data, we only need assumptions ensuring that (5.7) converges to a limit (explaining thus assumption 5).

For sake of completeness, we also recall the following measurability assumptions adapted from [VW96] and [Ber+16].

Assumption 6. *Let $(\eta_1, \ldots, \eta_N)$ be a survey sampling plan and assume that the maps*

$$(\eta_1, \ldots, \eta_N) \mapsto \sup_{\rho_P(f,g)<\delta} \left|\sum_{i=1}^N e_i \left(\frac{1}{\sqrt{N}}\left(\frac{\eta_i}{p(X_i)} - 1\right)(f(Y_i) - g(Y_i))\right)\right|$$

and

$$(\eta_1, \ldots, \eta_N) \mapsto \sup_{\rho_P(f,g)<\delta} \left|\frac{1}{N}\sum_{i=1}^N e_i \left(\frac{\eta_i}{p(X_i)} - 1\right)^2 (f(Y_i) - g(Y_i))^2\right|$$

are measurable for every $\delta > 0$, $(e_1, \ldots, e_N) \in \{-1, 0, 1\}^N$ and $N \in \mathbb{N}$.

Apart from these measurability assumptions (ensuring that the limiting distribution will live in a measurable space), the main ingredient for ensuring convergence of the empirical process to a Gaussian process is based on the behavior of the uniform covering number. In particular we will assume the following Dudley's condition stated in

Assumption 7. *Uniform entropy condition. Let $\mathcal{D}$ be the set of all finitely discrete probability measures $\mathbb{Q}$ with $\int H^2 d\mathbb{Q} < +\infty$, then assume that*

$$\int_0^\infty \sup_{\mathbb{Q}\in\mathcal{D}} \sqrt{\log \mathcal{N}(\varepsilon \|H\|_{2,\mathbb{Q}}, \mathcal{F}, \|.\|_{2,\mathbb{Q}})}\, d\varepsilon < \infty.$$

Finally to be able to apply the CLT obtained in the preceding part to the survey sampling framework we will also need the Lindeberg-Feller type assumption:

Assumption 8. *Lindeberg-Feller type condition holds:* $\forall\, \varepsilon > 0$,

$$E_{P_{Y,X}}\left(E_{R_N}\left(\mathcal{X}_{N,i}^2 \mathbb{I}\left\{\mathcal{X}_{N,i} > \varepsilon\sqrt{N}\right\}\right)\right) \underset{N\to\infty}{\longrightarrow} 0,$$

with $\mathcal{X}_{N,i} := \left|\frac{\eta_i}{p(X_i)} - 1\right| \sup_{f\in\mathcal{F}} |f(Y_i)|$.

Recall that this assumption is satisfied if we have $E_{P_{Y,X}}\left(E_{T_N}\left(\mathcal{X}_{N,i}^{2+\varepsilon}\right)\right) < \infty$, for some $\varepsilon > 0$, which holds for instance if $E_{P_{Y,X}}\left(\frac{H(Y)}{p(X)}\right)^{2+\varepsilon} < \infty$.

We are now ready to state our main result.

Theorem 16. *Let $\mathcal{F}$ be a set of positive real functions satisfying conditions 4, 6, 7 and 8. Assume in addition that conditions (1), (2) and (3) on the survey sampling plan holds. Then, there exists a ρ_P-equicontinuous Gaussian process $\mathbb{G}^p$ in $\ell^\infty(\mathcal{F})$ with covariance operator $\Sigma^p(f,g)$ such that, almost surely along the sequence,*

$$\mathbb{G}_{R_N}^{\mathbf{p}} \Rightarrow \mathbb{G}^p \text{ weakly in } \ell^\infty(\mathcal{F}),\ \text{as } N \to \infty.$$

Proof: The proof is very similar to the proof of [Ber+16] and relies on Theorems 1.5.4 and 1.5.7 of [VW96]. We essentially need to check the two following conditions
1) the finite dimensional marginal are asymptotically Gaussians,
2) the process $\{\mathbb{G}_{R_N}^{\mathbf{p}} f :\ f \in \mathcal{F}\}$ is asymptotically tight and fulfills an asymptotic equicontinuity condition.
Let us check these two conditions:

1) Let $\mathbf{f}_K = (f_1, \ldots, f_K)$ be any vector of functions in $\mathcal{F}^K$, $K \geq 1$. Then the finite dimensional marginals $\mathbb{G}_{R_N}^{\mathbf{p}} \mathbf{f}_K := (\mathbb{G}_{R_N}^{\mathbf{p}} f_1, \ldots, \mathbb{G}_{R_N}^{\mathbf{p}} f_K)$ are asymptotically Gaussian conditionally on the data $\{(Y_1, X_1), \ldots, (Y_N, X_N)\}$ with limiting covariance matrix $\Sigma_{\mathbf{f}_K}^p := (\Sigma^p(f_k, f_\ell))_{1\leq k,\ell\leq K}$ $a.s.$ As noticed in [Ber+16], this can be checked by tacking any linear combination $a'\mathbb{G}_{R_N}^{\mathbf{p}} \mathbf{f}_K,\ a \in \mathbb{R}^{+K}$. This amounts to checking the

asymptotic normality of $\mathbb{G}^{\mathbf{p}}_{R_N} a'\mathbf{f}_K$. But under the Lindeberg-Feller assumption and assumptions (1), (2) and (3), this is a direct consequence of the negative association property of the sampling plan and of the CLT given in Theorem 15.

2) The process $\{\mathbb{G}^{\mathbf{p}}_{R_N} f : \ f \in \mathcal{F}\}$ is asymptotically tight in that it fulfills the asymptotic equicontinuity condition, for any $e > 0$, almost surely along the sequence $\mathcal{D}_N$,

$$\lim_{\delta \to 0} \limsup_{N \to \infty} P_{R_N} \left(\sup_{\rho^2_{P_X}(f,g) \leq \delta} |\mathbb{G}^{\mathbf{p}}_{R_N} f - \mathbb{G}^{\mathbf{p}}_{R_N} g| > e \right) = 0.$$

For this observe that we have for all $(f, g) \in \mathcal{F}^2$, for some constants $C_1, C_2, ...$

$$\begin{aligned} E_{R_N}\left((\mathbb{G}^{\mathbf{p}}_{R_N}(f-g))^2\right) &\leq C_1 \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} (\pi_i \pi_j - \pi_{i,j}) \left(\frac{f(Y_i) - g(Y_i)}{p(X_i)} - \frac{f(Y_j) - g(Y_j)}{p(X_j)} \right)^2 \\ &\leq C_2 ||f - g||^2_{2,P_N}. \end{aligned}$$

Now, proceed as in [Ber+16] by noticing that sums of bounded negatively associated random variables satisfy Hoeffding inequality. It follows that the process $\{\mathbb{G}^{\mathbf{p}}_{R_N} f : f \in \mathcal{F}\}$ is conditionally sub-Gaussian with respect to the semi-metric $\rho^2_{P_N}$. Then Corollary 2.2.8 in [VW96] yields, for any $\delta > 0$,

$$E_{R_N}\left(\sup_{\rho^2_P(f,g) \leq \delta} |\mathbb{G}^{\mathbf{p}}_{R_N} f - \mathbb{G}^{\mathbf{p}}_{R_N} g| \right) \leq K \int_{\epsilon=0}^{\delta} \sqrt{\log\left(\mathcal{N}(\epsilon/2, \mathcal{F}_\delta, L_2(P_N))\right)} d\epsilon \tag{5.8}$$

where $K < +\infty$ is a constant and $\mathcal{F}_\delta = \{f - g, \ \rho^2_{P_N}(f, g) \leq \delta\}$. Now conclude by recalling that (see p. 128 in [VW96]),

$$\mathcal{N}(\zeta, \mathcal{F}_\delta, L_2(P_N)) \leq \mathcal{N}(\zeta/2, \mathcal{F}, L_2(P_N))^2 < +\infty. \tag{5.9}$$

Finiteness of $\int_{\epsilon=0}^{1} \sup_Q \sqrt{\log\left(\mathcal{N}(\epsilon/2, \mathcal{F}_\delta, L_2(Q))\right)}$ implies that the right hand side of (5.8) converges to 0 as $\delta \to 0$,which in turn ensures by Markov inequality that

$$P_{R_N}\left(\sup_{\rho^2_{P_X}(f,g) \leq \delta} |\mathbb{G}^{\mathbf{p}}_{R_N} f - \mathbb{G}^{\mathbf{p}}_{R_N} g| > \eta \right) \to 0.$$

Remark: The results on the Horvitz Thompson estimator can be straightforwardly extended to the Hajek estimator. Indeed it is known that for fixed N, $P^{\boldsymbol{\pi}}_{R_N}$ is not a probability but a positive measure. The corresponding standardized estimator is the Hajek-empirical measure defined by

$$P^{\boldsymbol{H}}_{R_N} := \frac{\sum_{i=1}^{N} \frac{\eta_i}{\pi_i} \delta_{Y_i}}{\sum_{i=1}^{N} \frac{\eta_i}{\pi_i}} = \frac{P^{\boldsymbol{\pi}}_{R_N}}{\frac{1}{N}\sum_{i=1}^{N} \frac{\eta_i}{\pi_i}}$$

Since the denominator $\frac{1}{N}\sum_{i=1}^{N}\frac{\eta_i}{p(X_i)}$ converges to 1 as $N\to\infty$ (and do not depend on f), all the results remain valid with the process defined with this estimator instead of $P_{R_N}^{\boldsymbol{\pi}}$, with the same limiting distributions.

5.5 Applications and simulations

5.5.1 Linearization and asymptotics of non linear functionals

The Donsker type theorems proved before allow to establish the asymptotic normality of many statistics that can be seen as empirical version of some Hadamard differentiable functional, see [SW86]. For the sake of completeness, we recall the definition of *uniform Hadamard differentiability* used in [Ber+14], see also [PDT91]. We follow also the same arguments and generalize the results of [Ber+16] proved for Poisson and conditional Poisson plans to negatively associated sampling plans. The asymptotic normality of the Horvitz-Thompson version of the functional of interest also apply to the multivariate examples considered in [PDT91], including some functionals based on censored data. Other examples are treated in [VW96] (see Chapter 3.9 p.379).

Define $B(\mathcal{F},P_X)$ as the set of measures $\mathbb{Q}$ in $\ell^\infty(\mathcal{F})$ whose paths $f\in\mathcal{F}\mapsto\mathbb{Q}f$ are $\|.\|_{2,P_X}$-uniformly continuous and bounded. The uniform Hadamard differentiability tangentially to the subspace $B(\mathcal{F},P_X)$ weakens the notion of Hadamard differentiability and is well suited for many situations, including product integrals, a natural functional appearing in the censored data literature.

Definition 8. *A functional $T:\ell^\infty(\mathcal{F})\to\mathbb{R}^q$ is said to be uniformly* Hadamard differentiable *at P_Y tangentially to $B(\mathcal{F},P_Z)$ if and only if there exists a continuous linear mapping dT_{P_Y} such that for any sequence P_N converging to P_Y, any h_N converging to $h\in B(\mathcal{F},P_Y)$ and every t_N converging to 0 such that $P_N+t_N\,h_N\in\ell^\infty(\mathcal{F})$, we have: $\forall h_N\in\ell^\infty(\mathcal{F})$, $\forall t_N\in\mathbb{R}^*$,*

$$\frac{T(P_N+t_N\,h_N)-T(P_N)}{t_N}-dT_{P_Y}\,h_N\underset{t_N\to0}{\longrightarrow}0.$$

Notice that T may be defined not on the entire space $\ell^\infty(\mathcal{F})$ but on a subset $\mathcal{L}$ only. In this case, one must check that $P_N+t_N\,h_N\in\mathcal{L}$.

Theorem 17 (CLT for Hadamard differentiable functionals)**.** *Let R_N be a negatively associated sampling design with inclusion probabilities $\boldsymbol{\pi}_j$, $\pi_{i,j}$ determined respectively by the link functions $p:\mathcal{W}\to[p_\star,1]$ and h: $\mathcal{W}\times\mathcal{W}\to\mathbb{R}$ satisfying all the hypotheses*

of Theorem 16 . Suppose that $T : \mathcal{L} \subset \ell^\infty(\mathcal{F}) \to \mathbb{R}^q$ is Hadamard differentiable at P_Y with differential dT_{P_Y}. Then, as $N \to +\infty$, we have

$$\sqrt{N}\,\left(T(P_{R_N}^{\boldsymbol{\pi}}) - T(P_N)\right) \Rightarrow dT_{P_Y}\,\mathbb{G}^p,$$

where $\mathbb{G}^p$ is a Gaussian process with covariance operator Σ^p.

Proof: The proof simply follows by applying Hadamard differentiability to the sequence $h_N = \sqrt{N}(P_{R_N}^{\boldsymbol{\pi}(R_N)} - P_N) =: \mathbb{G}_{R_N}^{\boldsymbol{\pi}(R_N)}$, which converges to $h = \mathbb{G}$ in $\ell^\infty(\mathcal{F})$ and with $t_N = \frac{1}{\sqrt{N}} \to 0$. Following line by line, the arguments in [PDT91], we have, as $N \to +\infty$:

$$\sqrt{N}\,\left(T(P_{R_N}^{\boldsymbol{\pi}(R_N)}) - T(P_N)\right) = \sqrt{N}\,\left(T(P_N + \frac{1}{\sqrt{N}}h_N) - T(P_N)\right) \longrightarrow dT_{P_Y}.\mathbb{G}.$$

In the particular case when the functional admits a computable influence function $T^{(1)}(x, P)$ such that

$$dT_{P_Y}.Q = \int T^{(1)}(x, P_Y)Q(dx)$$

Then the covariance operator simply becomes

$$S^2_{P_{y,X}} = \frac{1}{2}E_{P_{Y,X}^{\otimes 2}}\left(h(X_i, X_j)\left(\frac{T^{(1)}(Y_i, P_Y)}{p(X_i)} - \frac{T^{(1)}(Y_j, P_Y)}{p(X_j)}\right)^2\right).$$

Assuming that

$$0 < S^2_{P_{y,X}} < \infty$$

then it is well-known that the limiting distribution is

$$dT_P.\mathbb{G} = N(0, S^2_{P_{Y,X}}),$$

a Gaussian random variable, with variance given by $S^2_{P_{y,X}}$. A natural question is then how to choose the survey sampling plan so as to minimize the asymptotic variance.

We recall that in the robustness terminology [Ham74], the influence function of the parameter $T(P_Y)$ may be calculated directly by computing the derivative of the functional taken at the contaminated distribution $(1-t)\,P_Y + t\,\delta_y$, *i.e.*

$$T^{(1)}(y, P_X) := \lim_{t \to 0} \frac{T\left((1-t)\,P_Y + t\,\delta_y\right) - T(P_Y)}{t}.$$

Hadamard differentiability is sometimes difficult to prove and it does not yield a precise control of the remainder for further approximations like Edgeworth expansions. Another approach followed by [Dud90] and [BB95] is to assume Fréchet differentia-

bility with respect to a metric $d_{\mathcal{F}}$ indexed by a class of function $\mathcal{F}$, for which some uniform-entropy conditions hold (see also [HR09]). A functional is said to be *Fréchet differentiable* at P_Y for such a metric if there exists a gradient (the influence function $T^{(1)}(y, P_Y)$, which fulfills $E_{P_Y}\left(T^{(1)}(x, P_Y)\right) = 0$) and a continuous function $\xi(.)$, null at 0, such that for any probability $\mathbb{Q}$,

$$T(\mathbb{Q}) - T(P_Y) = \int T^{(1)}(y, P_Y)\,(\mathbb{Q} - P_Y)(dy) + d_{\mathcal{F}}(\mathbb{Q}, P_Y)\,\xi(d_{\mathcal{F}}(\mathbb{Q}, P_Y)).$$

It is generally possible to choose the class of functions according to the functional of interest, see for instance [AG92] for general classes of M-estimators or [BB95] for general functionals appearing in survival analysis. Notice that in that case, by applying Fréchet differentiability twice, we have

$$\begin{aligned}\sqrt{N}\left(T(P^{\boldsymbol{\pi}}_{R_N}) - T(P_N)\right) &= \sqrt{N}\int T^{(1)}(y, P_Y)\,(P^{\boldsymbol{\pi}}_{R_N} - P_N)(dy) + r_N \\ &= \frac{1}{\sqrt{N}}\sum_{i=1}^{N}\left(\frac{\epsilon_i}{\pi_i} - 1\right) T^{(1)}(Y_i, P_Y) + r_N,\end{aligned}$$

with a remainder

$$r_N := \sqrt{N}\, d_{\mathcal{F}}(P^{\boldsymbol{\pi}}_{R_N}, P_Y)\,\xi(d_{\mathcal{F}}(P^{\boldsymbol{\pi}}_{R_N}, P_Y)) + \sqrt{N}\, d_{\mathcal{F}}(P_N, P_Y)\,\xi(d_{\mathcal{F}}(P_N, P_Y)).$$

By virtue of the results in [Háj64], it is then obvious that the linear term in this approximation is asymptotically Gaussian with known variance. Controlling the remainder essentially amounts to controlling the behavior of $\sqrt{N}\, d_{\mathcal{F}}(P^{\boldsymbol{\pi}}_{R_N}, P_X)$ or alternatively that of $\sqrt{N}\, d_{\mathcal{F}}(P^{\boldsymbol{\pi}}_{R_N}, P_N)$ (by the triangular inequality), which was the purpose of our main theorem. Of course we retrieve directly in that case that the limiting distribution of the linear part is given by $S^2_{P_{y,X}}$ under our previous assumptions.

Example 18 (Quantile)**.** *It is well-known that the functionals $T(P_X) = E_{P_X}(X)$ and $T(P_X) = \mathbb{V}_{P_X}(X)$ as well as fixed quantile are uniformly Hadamard differentiable when considering appropriate classes of function $\mathcal{F}$ (see for instance [Rie94]. For $T(P_X) = E_{P_X}(X)$, Theorem 17 reduces to the standard Central Limit Theorem for associated survey sampling.*

5.5.2 Confidence bands for a CDF : a simulation study

Consider the class of functions $\mathcal{F} := \{f_x(.) := \mathbb{I}\{. \leq x\}, x \in \mathbb{R}\}$. It respects the required uniform-entropy condition and admits an envelop equal to $H(x) = 1$ and

$\delta = 1$. Then the asymptotic uniform confidence band of level $\alpha \in (0,1)$ for the population cdf F_N will be given by

$$CB_\alpha := \left[F^{\pi}_{R_N} - \frac{q_\alpha}{\sqrt{N}} \, , \, F^{\pi}_{R_N} + \frac{q_\alpha}{\sqrt{N}} \right],$$

where $F^{\pi}_{R_N}$ is the Horvitz-Thompson estimator of the cdf based on any negatively associated sampling plan. q_α is the α-quantile of the random variable $\|\mathbb{G}^p\|_{\mathcal{F}}$. In practice, q_α can be simulated using the procedure described in [Ber+16] in their appendix section S5.

Experiment setting

We make some simulations based on a similar model to the one used in the applications in [Ber+16]:

$Y = \beta X + U$ $X \sim \mathcal{TN}(0, \sigma^2, x_{min}, x_{max})$, $U \sim \mathcal{N}(0, \sigma_U^2)$, $P(X \leq x, U \leq u) = P(X \leq x)P(U \leq u)$. Y is the variable of interest, X the auxiliary variable, U is a white noise independent from X, and $\mathcal{TN}(0, \sigma^2, x_{min}, x_{max})$ refers to the truncated Normal distribution taking values in $[x_{min}, x_{max}]$, with expectation μ and variance σ^2.

We consider a population $\mathcal{U}_N$ of size N where all the X_k and U_k are exchangeable realizations of X and U. An estimate of the cumulative distribution of Y is constructed using a sample of size $n = cN$ with $0 < c < 1$. The sampling design is a stratified sampling with a proportional allocation. The M strata are determined by the Z_k, indicators that a unit k belongs in the h-th quantile out of H. We present in figure 5.3 the confidence bands for the estimation of a cdf when $H = 2$ and $H = 25$.

The goal of this sampling design is to produce close-to-uniform confidence bands for the empirical estimates of F. As [Ber+16] showed in their application, constructing the inclusion probabilities only to minimize the variance of the estimate of $T(X)$ tends to narrow the confidence bands in a neighborhood of the mean of the distribution but does not give a good tail approximation.

We carry out K simulations with a simple algorithm. For each $k = 1 \dots K$, we randomly draw a new realization of Y and a sample of size n using the chosen sampling design. Then we compute the estimated cumulative distribution of Y on sample data. Then the simulated confidence bands at the α level are the envelope of the simulated estimates where the $1 - \alpha$ most extreme results are removed.

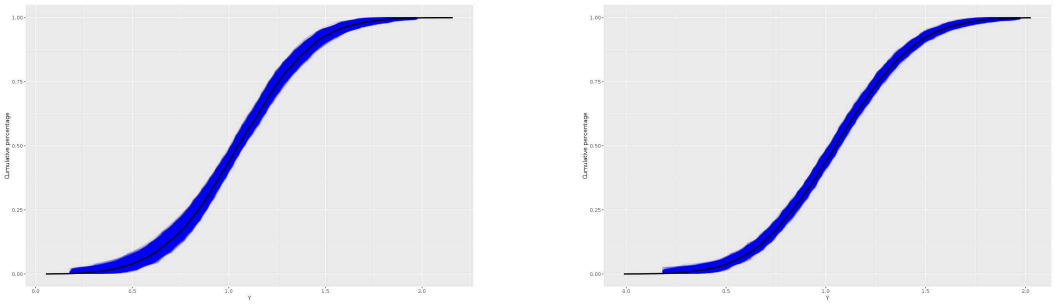
Numerical results

Numerical experiments were made with: $\beta = 1$, $\sigma_U^2 = 0.01$, $[x_{min}, x_{max}] = [0.01, 2]$, $\mu = 1, \sigma^2 = 0.09$, $\beta = 1$, $\sigma_U^2 = 0.01$, $c = 0.05$.

It is interesting to note that, given that the sampling design is constructed with a precision goal on the cdf and not directly on the on the estimate of the variable of interest, the results of the simulations won't depend much on the value of β and σ_U^2 (we checked this by simulations, but we don't report the results here for sake of clarity).

Figure 5.3 shows the 95% confidence bands obtained by simulations for the estimation of the cdf of Y, for a choice of parameters documented in paragraph 5.5.2. Figure 5.3a show the situation when only 2 strata are used ($H = 2$), and in figure 5.3b, 25 strata are used ($H = 25$).

We see that increasing the number of strata produces bands that are closer to uniform across the whole distribution (even though the precision of the estimate at the tail of the distribution is still not great).



(a) H=2 **(b)** H=25

Fig. 5.3: Asymptotic confidence bands at the 95% level with two different values for H, obtained with $\beta = 1$, $\sigma_u^2 = 0.01$, $N = 10000$, $c = 0.05$ and 10000 simulations. We observe that the bands are more uniform with a higher number of strata (i.e. quantiles in the balancing constraints).

We show a summary of the distribution of the width of the confidence bands in table 5.1. We confirm that adding more strata to the design reduces the dispersion of the width of the confidence bands. However, we see that the maximum width is always very high no matter the number of strata used. This corresponds to the last strata which contains the tail of the distribution of Y.

An other idea to improve the design would be to balance the sampling on the indicators of belonging to each one of the quantiles of X (e.g. by balanced sampling

N	1000			5000			10000		
H	2	25	100	2	25	100	2	25	100
min	0.067	0.063	0.076	0.108	0.047	0.071	0.093	0.059	0.071
mean	0.179	0.162	0.149	0.233	0.146	0.161	0.211	0.178	0.148
median	0.164	0.132	0.129	0.197	0.121	0.121	0.184	0.159	0.127
q75	0.227	0.152	0.171	0.272	0.168	0.173	0.241	0.19	0.161
q90	0.262	0.226	0.224	0.325	0.227	0.266	0.382	0.318	0.216
q95	0.37	0.402	0.33	0.548	0.311	0.377	0.394	0.355	0.284
max	0.44	0.473	0.353	0.646	0.517	0.532	0.48	0.476	0.41

N	50000			100000		
H	2	25	100	2	25	100
min	0.074	0.069	0.064	0.112	0.097	0.069
mean	0.172	0.167	0.158	0.215	0.182	0.138
median	0.151	0.129	0.129	0.182	0.163	0.107
75th centile	0.181	0.19	0.153	0.272	0.186	0.148
90th centile	0.276	0.248	0.277	0.287	0.227	0.198
95th centile	0.344	0.412	0.377	0.362	0.344	0.315
max	0.494	0.534	0.52	0.616	0.523	0.469

Tab. 5.1: Distribution of width of confidence bands for different values of N and H. Results obtained with 10000 simulations

or calibration). This would help reduce the variance of estimation for the mean of Y as well as the width of the confidence bands.

We could also imagine designs that also take care of the widest confidence bands at the tail of the distribution by specifically constructing the strata on the expected width of the confidence band instead of balancing only on the auxiliary variable X. A way to tackle this would be by allowing unequal probability sampling inside the strata (two-degree sampling design). This is left for further research.

5.6 Asymptotics of graph sampling

In this section we make a few remarks about the link between CLT for the sampling designs studied in part II and sampling designs for networks as studied in part III.

First, we discuss why negative association does not hold for snowball sampling. Then, we show that the negative association method can produce results the very simple case of adaptive sampling with Bernoulli initial design on an Erdős-Rényi graph model.

Snowball sampling is not negatively associated Section 5.2.4 shows how negative associated algorithms performed in a multistage design can still check out some NA property without the actual indicator of selection η being NA itself (this illustrated in figure 5.1). This requires that samples at all stages are two-by-two independent though.

Unfortunately, snowball sampling does not check this requirement. Figures 5.4 and 5.5 illustrates that. In figure 5.4, we consider an example graph with 7 vertices and 6 edges. We perform a snowball sample with initial sample size of 1. Possible samples (with extension) are listed in figure 5.5, depending on the unit selected at the first stage.

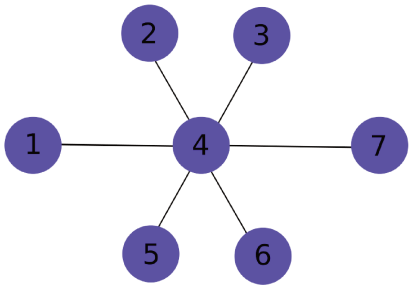
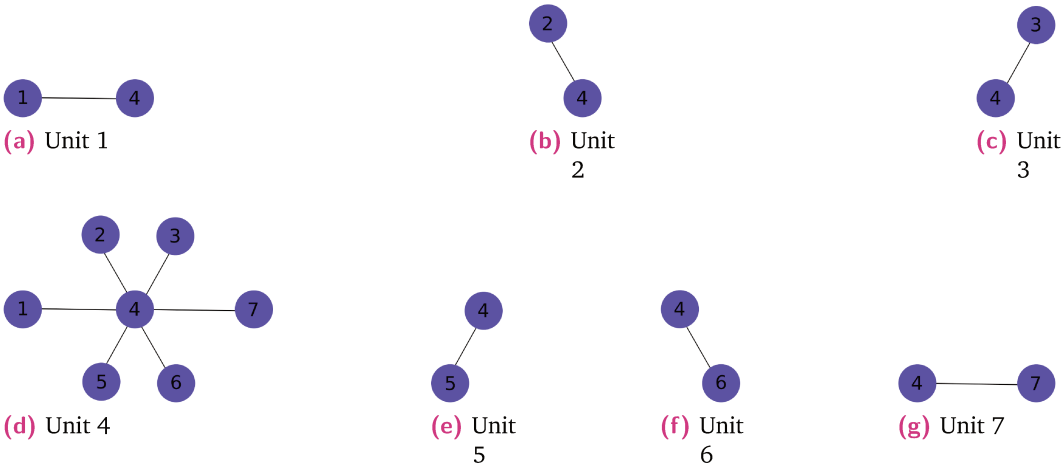


Fig. 5.4: Example of a graph with 7 vertices and 6 edges.

We observe that units that are selected in the extension phase appear in many other possible samples selected with other units at the first stage. First and second stage of the snowball sample are thus not two-by-two independent. In other terms, analogy between snowball sampling and two-stage sampling is invalid.

Adaptive sampling We consider an Erdős-Rényi $G(N, p)$ graph model with parameter $p = (1 = \epsilon) \log N / N^{1+\alpha}$, with $\epsilon, \alpha > 0$. Then with probability 1, the network is not connected, and composed of small connected portions that can be seen as clusters (figure 5.6). We also consider a Bernoulli of probability π as initial sampling design.



(a) Unit 1 (b) Unit 2 (c) Unit 3

(d) Unit 4 (e) Unit 5 (f) Unit 6 (g) Unit 7

Fig. 5.5: Total sample (with extension), for a snowball sample of initial sample size 1 on the graph of figure 5.4, depending on the unit selected at the initial stage

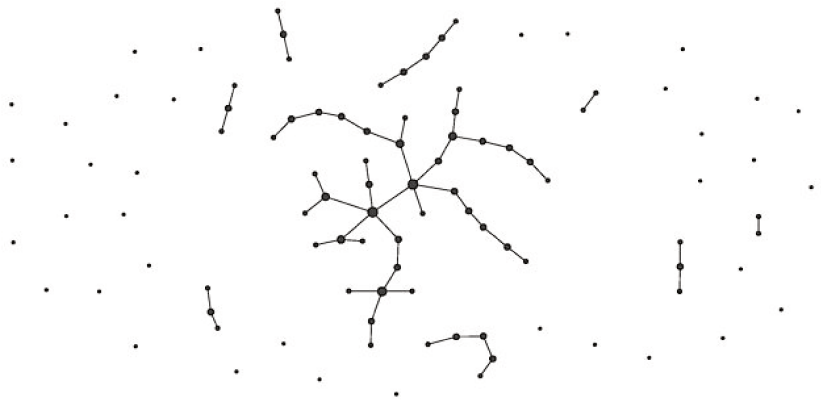


Fig. 5.6: An Erdős-Rényi model of 100 vertices with p=0.01. The graph is unconnected.

Using formula $\hat{T}(Y)_4$ of section 7.3.3, we then consider data of the K networks induced by the N units. These networks are observed with probabilities $\pi_k = 1-(1-\pi)^{n_k}, k = 1 \dots K$, with n_k the size of network k. Given that the initial sample is Bernoulli, the sampled networks are observed independently:

$$\Delta_{kk'} = \pi_k + \pi_{k'} - (1-(1-\pi)^{n_k+n_{k'}}) - (1-(1-\pi)^{n_k})(1-(1-\pi)^{n_{k'}}) = 0$$

Then with a functional checking all assumptions of theorem 16, the main result holds for this very particular sampling design.

Part III

Sampling big networks

6 Graphs: statistics and sampling

6.1 Introduction: Big graphs and sampling

In the modern Internet industry, graphs and graph databases are to be found in every product and company. In social networks like Facebook, Twitter, Instagram, etc., the users of a service compose the vertices of the graph, and the links between the users (follow, friendship) compose the edges of the graph. Those graphs can thus be directed (Twitter, the "follow" links between users are asymmetrical) or undirected (Facebook, each link between friends is symmetrical). As of 2013, theses services had to the order of a billion of users ([BW13]).

In other digital companies whose products are not social networks, graphs are still very useful and common statistical objects. A good example are recommender systems ([Kor+09]). These systems consists in providing personalized content to each user. The most famous example is the video streaming application Netflix, where famously "everything is a recommendation" [1], meaning that most of the watching time comes from a suggetion that was tailor-made for each user by an algorithm. Such systems are said to possess a very high intrinsic value, estimated to be as high as 1 billion dollars in the case of Netflix ([GUH16]).

Most of the recommendation algorithms work by taking as input the matrix of interactions between each user and each product available in the catalogue ([BL+07]). This can be seen as a graph structure ([Hua+04]). In some companies, recommendations are directly used on a social network, like for example when Twitter or Facebook recommend friends to connect with ([Han+10]).

Technically speaking, some of these graphs (and the computations on it) are handled in-memory (for example at Twitter and Google, see [Gup+13]). But this is very costly, and not accessible to smaller companies. A cheaper solution is to use parallelized algorithms and software that allow practical management of resources for parallel computation (for example [Vav+13]). One of the most famous of these frameworks is Hadoop ([Shv+10]). In the last few years, software that can

[1] https://medium.com/netflix-techblog/netflix-recommendations-beyond-the-5-stars-part-1-55838468f429

handle specifically storing and querying of graph databases efficiently have appeared ([Xin+13], [Mal+10]).

Still, many data science problems require running algorithms that are computationally very intensive (see 6.2.2). Both the time required for the algorithms to complete and size of memory are problematic, even in the parallelized case. In addition, maintaining a fully distributed database architecture creates technical debt [Kru+12] and is thus still costly. These methods are not adapted for quick analyses or small data science teams.

Besides, not all algorithms can be handled in a parallelized way. A good example is graph visualization. First, there are many different ways to visualize the same graph A good survey on graph visualization techniques can be found in [CQ07]. We present in figure 6.1 several different ways to visualize the same graph. Some of the plots emphasize units with high degree, some highlight efficiently highly clustered areas. Others make it easy to spot communities.

Given a network, a visualization has to be chosen knowing that some key properties will not show very well. Automation of the choice of a visualization is almost never possible. Overall, having to choose from a huge set of parameters and algorithms makes quick and efficient visualization methods much more practical.

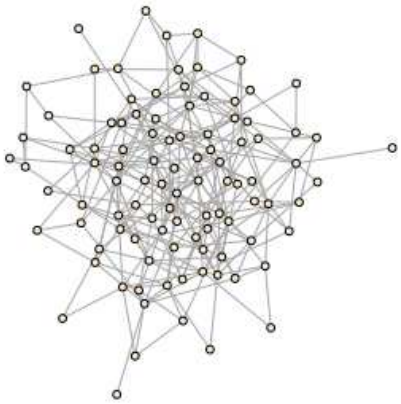
Second, each of these visualizations are tackle by algorithms that are almost all single-machine single-core software and algorithms ([Jac+14], [Jac+14]). Although distributed algorithms exist ([Mue+06]), they seem not to be practical, and they are not implemented in any popular graph data treatment software.

Graph sampling can provide solutions for all these problems. Statistical or graphical software working on network objects could fasten a lot of procedures in graph statistical analyses. Some authors recommend using model-based estimation, but they suffer a lot of drawbacks in the context of networks (see paragraph 6.3.3).

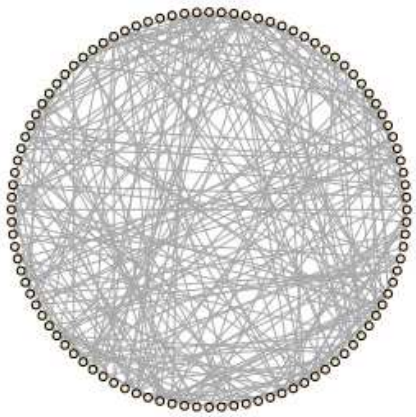
6.2 Graphs and statistics

6.2.1 Basic definitions and concepts

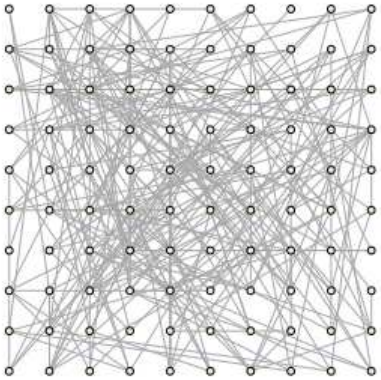
An (undirected) graph is a structure consisting of a set of vertices (or nodes) V and a set of edges (or links) E. Elements of E are (unordered) tuples $\{u, v\}$ of elements of V. The **size** of G is the number of its edges ($|E|$), the **order** of G is the



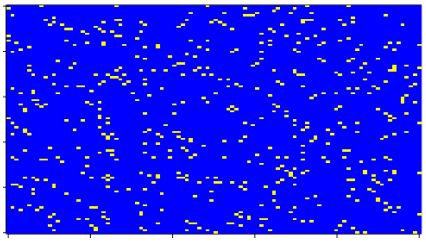
(a) Fruchterman-Reingold



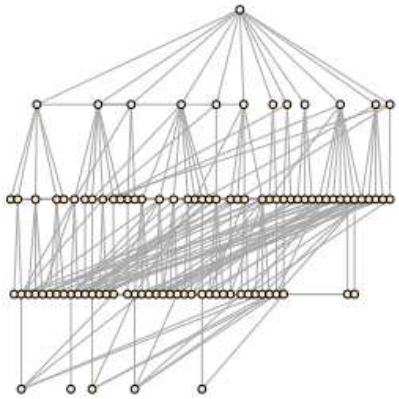
(b) Circle



(c) Grid



(d) Adjacency matrix



(e) Tree

Fig. 6.1: Different visualization algorithms for the same graph (Erdős-Rényi with 100 nodes and p=0.01).

number of its vertices ($|V|$). A graph $H = (V_H, E_H)$ is a **subgraph** of G if $V_H \subset V$ and $E_H \subset (E \cap \{\{v_1, v_2\} \in E_H \implies v_1, v_2 \in V_H\})$, that is a graph that is composed of a subset of vertices and edges of G, with the condition that all vertices that are linked by an edge of the subgraph are included in the subgraph (but it may contain additional vertices). H is a (vertex-)**induced subgraph** if E_H contains all the edges of G linking all elements of V_H together.

Edges will sometimes be referred to as links, vertices as nodes. A group of vertices linked to another vertex by some edges (distance of 1 between these vertices and a certain vertex) may be called friends of this vertex.

A **weighted graph** is a graph where each edge e is accompanied by a variable w_e named edge weight. On a weighted graph, the length of a path between two nodes will be computed as the sum of the weights of the edges traversed during the walk (see 6.2.2). We propose to define a **graph population structure** by extending graphs with two sets of variables: Y_j^V, with $j \in [\![1, J_V]\!]$, (random) vectors of size N_V and Y_j^E, with $j \in [\![1, J_E]\!]$, (random) vectors of size N_E. In this structure, every vertex and every edge has a realization of several (possibly random) variables associated to it. When $J_V = 0$ and $J_E = 1$, the graph population structure is a traditional weighted graph.

Typically, the variables Y_j^V and Y_j^E can contain:

- Weights
- Auxiliary variables
- Variables of interest
- Graph statistics (e.g degree, local clustering coefficient). They can also be considered auxiliary variables if the model for the variable of interest depends on graph geometry

For the rest of this document, we will refer to an object deriving from this "graph population structure" as a graph, except specified otherwise.

We say G is **connected** if every vertex is reachable form every other (i.e. there exists a path between every pair of vertices). If G is undirected, a **clique** is a subset C of V such that every two distinct vertices are linked by an edge.

The **adjacency matrix** of G is the square $|V| \times |V|$ matrix $A = (a_{i,j})$ where $a_{i,j} = 1$ if vertices i and j are connected, and 0 otherwise.

Example This example is inspired from a real-life example documented in [Tho98].

The goal is to study a population of drug users and to measure the amount of money each user spends on drugs each year. Drug use can be influenced by their ties to to other potential drug users.

The graph population is a realization of a random model given by the following algorithm:

- g is drawn from a forest-fire model (see paragraph 6.4.4) with order 25 and parameter 0.35
- The probability that each person in the population is a drug user is 0 when the local clustering coefficient of the corresponding vertex is lesser than 0.3, 0.35 otherwise
- For drug users, the money spent is drawn from the following linear model: $amount_k/10 = 1 + d_k + \epsilon_k$ where ϵ is a white noise and d_k the degree of vertex k

In figure 6.2, each user is colored in yellow if they're not a drug user, red if they are. The amount they spend each year on drugs is labeled on each vertex.

In the drug use example, the order of the population graph (i.e. the size of the population) is 25, the size of the graph (i.e. the number of links between people in the population) is 53. The mean amount spent on drugs is 20.8 in the population, and 52 among the drug users.

6.2.2 Some statistics and algorithms

In this section we describe a few interesting statistics used to describe graph structures, and the algorithms used to compute them.

Degree The degree d_k of a vertex k is the number of edges incident upon k. For directed graphs, we define d_k^{in} the in-degree of k, which is the number of edges directed towards k, and d_k^{out} the out-degree of k, the number of edges coming out of k.

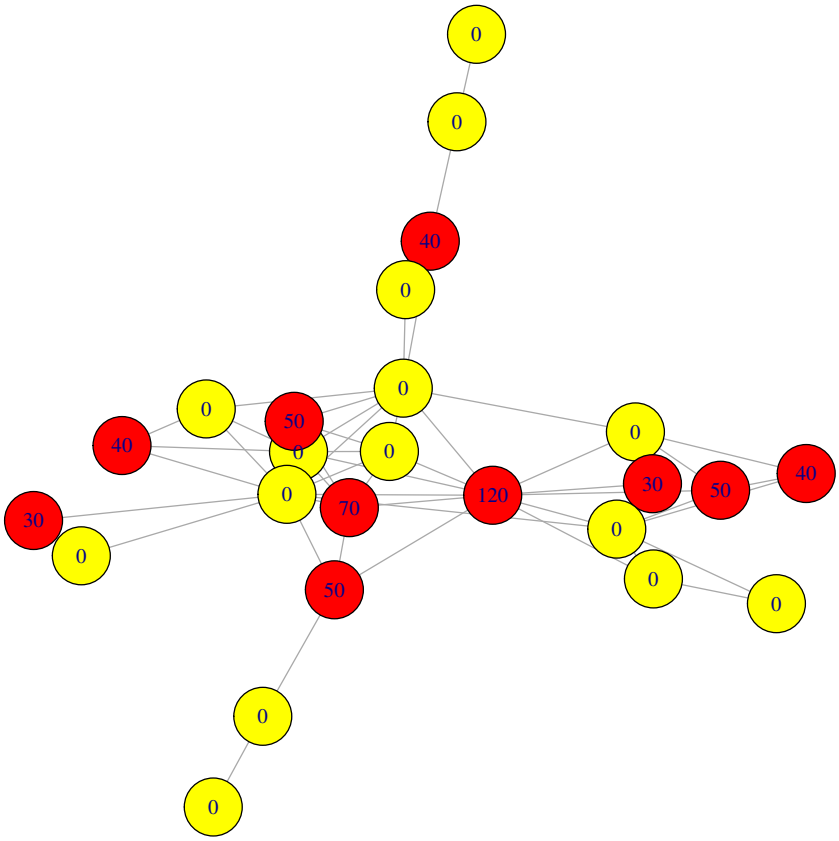


Fig. 6.2: Drug use graph population example

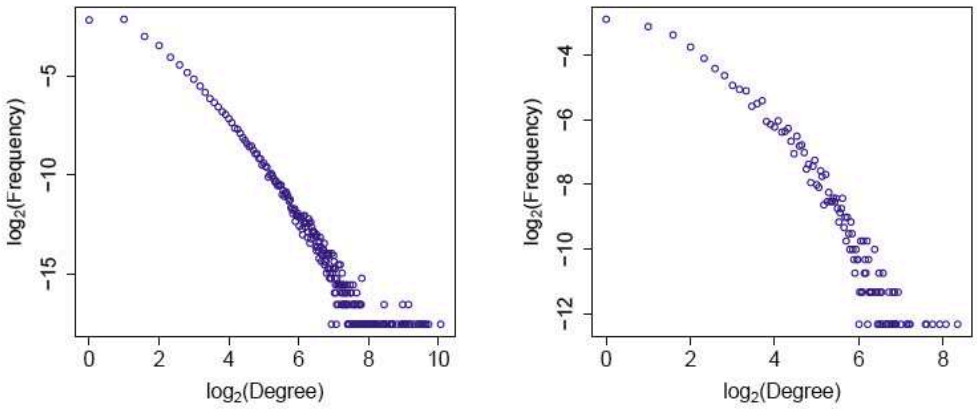


Fig. 6.3: Degree distributions. Left: the router-level Internet network graph from CAIDA data. Right: the network of measured interactions among proteins in yeast. In each plot, both x- and y-axes are in base-2 logarithmic scale. Graph excerpted from [Kol09]

An interesting statistic related to degrees is their distribution among vertices in the graph. In a lot of real-world networks, especially in applications related to Internet or the web, these distributions show a heavy tail. Power laws are often a good fit for the degree distributions of vertices in networks:

$$f_d \propto d^{-\alpha},\ \alpha > 2$$

[Kol09] gives two telling examples. The first one is the (undirected) network of Internet routers based on data collected by CAIDA [2] in August 2003. It contains 192 244 vertices and 609 066 edges. Descriptive statistics show that the degree distribution is highly skewed to the right. Degrees range from 1 to 1071. Mean degree is 6.3 while the median degree is only 3. The second one is the (undirected) graph of protein interactions in bakers' yeast. For the yeast network, degrees range from 1 to 301, mean degree is 12 and median degree is only 6. Figure 6.3 shows the degree distribution (in logarithmic scale) for these two networks. It shows that although most vertices have a very low degree, a non negligible number of vertices hold a very high degree, even 2 or 3 order of magnitudes higher than the mean. In the log-log scale, the slope of the curve appears to be close to linear, suggesting that a power law is a good fit for the degree distribution.

Another interesting example is the Twitter graph, which is directed. Similar distributions for out-degree and in-degree for Twitter users can be found in [Apa+15].

[2] Center for Applied Internet Data Analysis, http://www.caida.org

It is interesting to note that the discovery of these heavy-tailed distributions is fairly recent (see [Kol09], paragraph 4.2.1). Most graphs that were studied throughout the 20th century, like traditional random graphs (see paragraph 6.4.1) or networks of electrical circuits, showed homogeneous degree distributions.

Centrality Centrality metrics aim at measuring the "importance" of a vertex inside a graph. For example, in a social network, they give a measure of which user influence or can reach the most people. In a network of train lines, they would give a sense of how critical each station is for the operation of the whole system. Same goes for the Internet: centrality measures are supposed to identify the routers on which the stability of the global network relies the most. There are many definitions of centrality. In a 1978 survey, [Fre78] groups them in categories such as "distance", "flow", "feed-back" and "control". In this thesis, we will only use a few ones of them. They will be introduced in the next paragraphs.

First, note that **degree** is a measure of centrality itself.

Betweenness centrality of vertex k is defined as the rate of shortest paths between any tuple of vertices $l, m \neq k$ that include k:

$$\mathcal{C}_{btw,k} = \frac{2}{(N-1)(N-2)} \sum_{l \neq m \neq k \in V} \frac{\sigma(l,m|k)}{\sigma(l,m)}$$

where $\sigma(l,m|k)$ is the total number of shortest paths between l and m that pass through k and $\sigma(l,m) = \sum_{k \in V} \sigma(l,m|k)$. The factor $\frac{2}{(N-1)(N-2)}$ ensures that the metric always lies in $[0,1]$. Figure 6.4 illustrates a network with a visualization algorithm that places vertices with the highest betweenness centrality in the center of the image. The vertices are also colored depending on their betweenness centrality (red for the higher centralities, pale for the lower on figure 6.4).

Although the naive algorithm is $\mathcal{O}(N^3)$ in time and $\mathcal{O}(N^2)$ in memory, [Bra01] found an algorithm running in $\mathcal{O}(NN_e)$ time and needing $\mathcal{O}(N + N_e)$ memory.

Closeness centrality of vertex k is defined as the inverse of the total distance between k and all the other vertices:

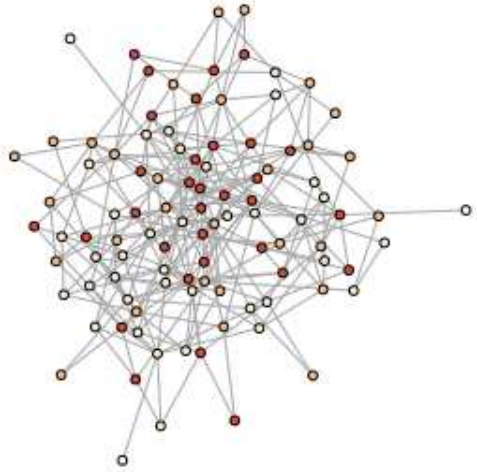


Fig. 6.4: Graph example with betweenness centrality. The darker the color, the higher the betweenness centrality.

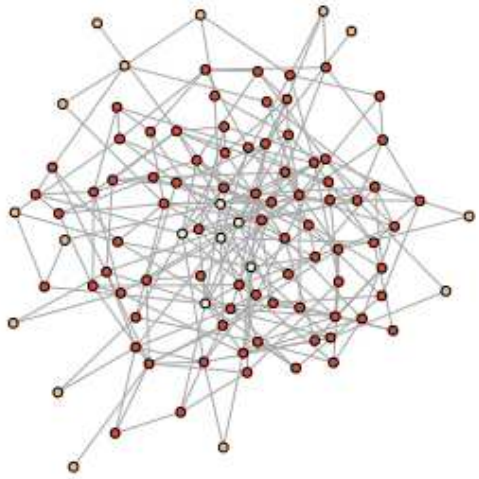


Fig. 6.5: Graph example with closeness centrality. The darker the color, the higher the centrality.

$$\mathcal{C}_{clos,k} = (N-1)\frac{1}{\sum_{l \in V} \delta(k,l)}$$

where $\delta(k,l)$ is the geodesic distance between vertices k and l. Figure 6.5 illustrates closeness centrality on an example network.

Using Dijkstra's algorithm, closeness centrality can be computed in $\mathcal{O}(N^2 \log N + NN_e)$ time.

Eigenvector centralities are also popular. The idea of this centrality is to assess the "status" (or "rank") of a given unit in a network. The more central the neighbors of

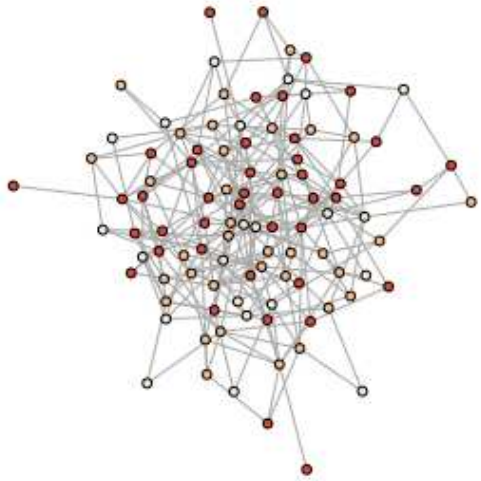


Fig. 6.6: Graph example with pagerank centrality. The darker the color, the higher the centrality.

a given vertex is, the more central this particular vertex has to be. This definition of centrality can be expressed as a system of linear equations, that can typically be solved using eigenvector decomposition. Writing $\mathbf{A} = (a_{k,l}), k, l \in V$ the adjacency matrix of graph G(V,E), the eigenvector centrality of vertex k writes:

$$\mathcal{C}_{eigen,k} = \frac{1}{\alpha} \sum_{l \in V} a_{k,l} x_l$$

$\mathcal{C}_{eigen,k}$ is thus the solution of the matrix equation $\mathbf{Ax} = \alpha\mathbf{x}$. In general, there exists an infinity of possible α. [Bon72] proposes to chose the largest eigenvalue of $\mathbf{A}$ for α. The results are taken in absolute value and normalized so they always lie in $[0; 1]$. Google's pagerank ([Pag+99]) is a particular case of eigenvector centrality and is a key component of the Google search engine. Figure 6.6 shows an illustration of pagerank centrality.

Clustering coefficients It can be interesting to try and quantify a notion of density of a graph. Do friends of a user of a social network tend to be friends of one another? [WS98] propose a **local clustering coefficient** (also called "transitivity") that measures density in the neighborhood of each vertex of the graph. This coefficient is the proportion of links between the vertices within its neighborhood divided by the maximum number of possible links between them. For example, for undirected graphs, denote d_k the degree of unit k. Then there are $d_k(d_k - 1)/2$ possible links between the vertices that are friends with vertex k so the clustering coefficient writes:

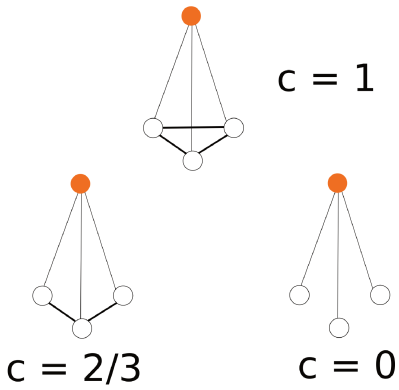


Fig. 6.7: Examples of possible values of local clustering coefficient for the vertex in orange, depending on the configuration of the network for its friends.

$$c_{loc,k} = \frac{2|N_k|}{d_k(d_k - 1)}$$

where N_k is the set of vertices that are linked with vertex k.

The algorithm to compute the local clustering coefficient for all vertices of the graph runs in $\mathcal{O}(N\bar{d_o}^2)$([Pei14]). Figure 6.7 illustrates the concept of local clustering coefficient.

The mean of the local clustering coefficient is an indicator of density of the graph. Another way to measure the density of the graph is the **global clustering coefficient** defined in [New03]:

$$c_{glob}(G) = \frac{3N_{\triangle}(G)}{N_3(G)}$$

where $N_{\triangle}(G)$ is the number of triangles in the graph G and $N_3(G)$.

Statistics of higher order A path between vertices i and j is a sequence of edges that connects them. The graph is connected if and only if a path exists between any pairs of vertices in the graph. The **distance** between i and j is the length of the shortest path between them. The **diameter** of a (connected) graph is the maximum distance between a two vertices.

Path lengths are a statistic of order 2 because they work with pairs of vertices. Other higher order statistics exist. It is for example interesting to count the number of

triangles (3-cliques) in the graph (for the global clustering coefficient, for example), which is a statistic of order 3. Some more complicated motifs are also an area of research.

6.3 Graph sampling

6.3.1 Definition

We say that a graph $G_s = (V_s, E_s, Y_s^V, Y_s^E)$ is a (vertex-)sample from $G = (V, E, Y^V, Y^E)$ if (V_s, E_s) is a subgraph of G, selected by a random sampling design p among all induced subgraphs of G and Y_s^V, Y_s^E are the restriction of Y^V, Y^E Then G is called the population graph.

Note that with this definition, any randomly selected subgraph can be a graph sample. Very simple properties such as connectivity can be lost in the sampled graph (see figure 6.8). Some sampling algorithms will guarantee that a (vertex-)induced subgraph is selected (for example extension sampling designs, see paragraph 6.3.4), preserving connectivity in the sample for connected population graphs.

Just like in traditional sampling of integer set populations, it is not practically possible to browse the set of all subgraphs and select one at random. Next parts will thus present algorithms designed to select subgraphs at random by browsing elements of the graph (vertices or edges).

In this thesis, we chose to make the vertices the population of interest, so in the rest of this thesis we will mostly discuss vertex-induced sampling designs, although it would be possible to present edge-induced graph sampling designs with some adjustments.

Efficient estimators will almost always require weights of some sort. Thus, the set of variables Y_s^V will almost always include a column $w \in \mathbb{R}^{|V_s|}$ (see paragraph 7 for discussions on estimators).

6.3.2 A note on graph observability

We start this section by making some remarks on observability of nodes and edges in a sampled graph. Our sampling designs will require browsing the population graph, thus the question of what edges and vertices are visible given that we see other vertices and edges is important.

Observability of links

Lots of interesting vertex-level graph statistics depend on the edges coming in and out of the said vertex and on the vertices that are linked to these edges. Some sampling designs also require some browsing of the graph, and will work efficiently only if all friends of a certain node are reachable (see 6.3.4).

So, a desirable property when browsing a graph, is that all links are observable given that we observe one vertex. This is the situation of **full link observability**.

However, this observability property does not always hold. Particularly in pioneering graph sampling studies, very few of the vertices connected to a given vertex were effectively observable.

Often, the motivation behind a graph sampling design in social science or official statistics institutes was to estimate rare statistics such as drug use or the number of people infected with HIV ([Hec+99]). Another famous study that involved graph sampling tried to estimate some characteristic of the population of jazz musicians in New York ([HJ01]).

For all these studies, the sampling algorithm consisted in an extension sampling design (see 7) of a particular form. A first sample was drawn with no prior information on the rare characteristic. It could be a simple random sampling (in official statsitics) or opportunistic equal-probability sampling if no reliable frame was available for the population. The sample size was calibrated so as to obtain a significant number of people bearing the characteristic of interest. Those people were incentivized to give a way to contact other people they knew (i.e in their network) bearing the same characterstic. Thus, only a fraction of someone's actual network was observable and included in the sample.

Compared to the situation of full link observability, estimation is more complicated. This is discussed for example in [Fra77]. Among other estimation methods in this case, we can use a Markov-Chain method ([Hec97]) or an Hansen-Hurwitz type estimator ([VH08]). [Lav12] gives a good survey on respondant-driven sampling.

For the rest of this article, unless mentioned otherwise, we consider that all links are observable (full link observability). When dealing with "big data" problems, this will almost always be the case (a counter-example would be when doing inference from a third-party API that does not allow for exhaustive browsing of all the links coming to or from nodes with particularly high degrees. This case is discussed in chapter 7).

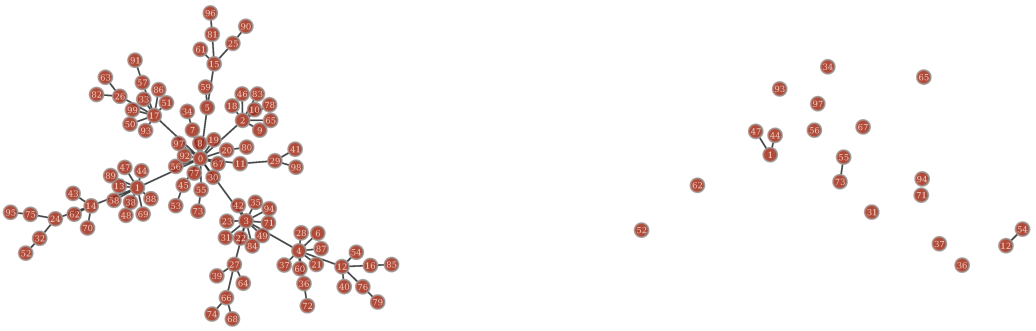
Different types of estimators

Given that a sample graph is selected, different ways exist to perform estimation.

The most simple form is **plugin estimation**. It consists in only considering the sample graph, and considering stats of the sample graph as estimators for values on the population graph. Plugin estimation is the most used form of estimation in computer science literature.

A sample graph can be any subgraph of the population graph, which means it is not necessarily an induced subgraph: all edges linking one vertice of the sample graph to another are not necessarily present in the sample.

Let's show on a very simple example that naive plugin sampling cannot produce sensible estimates of stats of the whole population, even for very simple properties. Figure 6.8 shows a connected graph with 100 vertices, and a sampled graph obtained with an induced subgraph sampling algorithm with a sampling fraction of 0.2 (the sampled graph has 20 vertices).



(a) A labeled graph with 100 nodes

(b) Graph obtained as a result of induced subgraph sampling algorithm with 20 nodes (sampling fraction of 0.2) from the population graph on the left

Fig. 6.8: Illustration of a sampled graph by the induced subgraph sampling method. The population graph is connected but the sampled graph is not. Thus plugin estimation of connectivity fails.

We observe that connectivity of the population graph is not preserved in the sample graph. Consequently, it will not be possible to compute any statistic depending on paths (betweenness centrality, path lengths, etc.) with this plugin sampling method.

Realistically, we cannot expect plugin estimation to produce good quality estimates, especially for small sample sizes and/or complex statistics. A rule of thumb proposed by [LF06] is that a sampling fraction of at least 15% is necessary to produce good estimates. However, the plugin estimates will always be biased. We show in section 8.2 that biases don't fade out sample size and cannot be ignored even for very high sampling fractions.

An alternative to plugin estimation is **observability-based estimators**.

We attach weights to every vertex and define estimators of a vertex-based statistic $S = \sum_{v \in V_G} f(v)$ with f some function as $\hat{S} = \sum_{v \in V_G} f(v, w_v)$ with w_v the weight attached to vertex v. Weight computation is discussed in 7. Several methods exist to obtain the most efficient estimators. The weights will often depend on the graph structure in the vicinity of each vertex of the sample graph.

With this estimation method, it is possible to construct estimators of any graph statistic without requiring specific properties on the sampled graph. Asymptotic properties of these estimators will be analyzed in paragraph 5. Properties of the sample graph do not matter anymore. In particular, it is possible to construct good estimators of path lengths without ensuring connectivity of the sample graph.

In terms of visualization, it is possible to plot observability-based samples as weighted graphs, with the vertices bearing the weights. Software such as Gephi ([Bas+09]) make this easy. However, it is hard to use weights as a visual cue to emphasize clusters or communities. Some weighted graphs might mathematically be very accurate estimates of very large populations, but be of no help when trying to convey information visually.

In fact, for visualization (and some other graph problems), plugin sampling might be a more adequate solution than its weighted counterpart. In these specific cases, defining efficient sampling designs for plugin estimation so as to minimize the size of the sample graphs used under constraints of precision could be useful. This is beyond the scope of this thesis.

6.3.3 Simple sampling algorithms

In this section we present simple random sampling algorithms for graphs. Algorithms such as Bernoulli / Poisson sampling or fixed size simple random sampling for integer populations cannot be directly used, because a graph sampling algorithm has to be able to capture both edges and vertices.

Induced subgraph sampling Induced subgraph sampling consists in taking a random sample of vertices in the population graph G and selecting their induced subgraph as the sample (using simple random sampling without replacement). An illustration of this method is presented in figure 6.9.

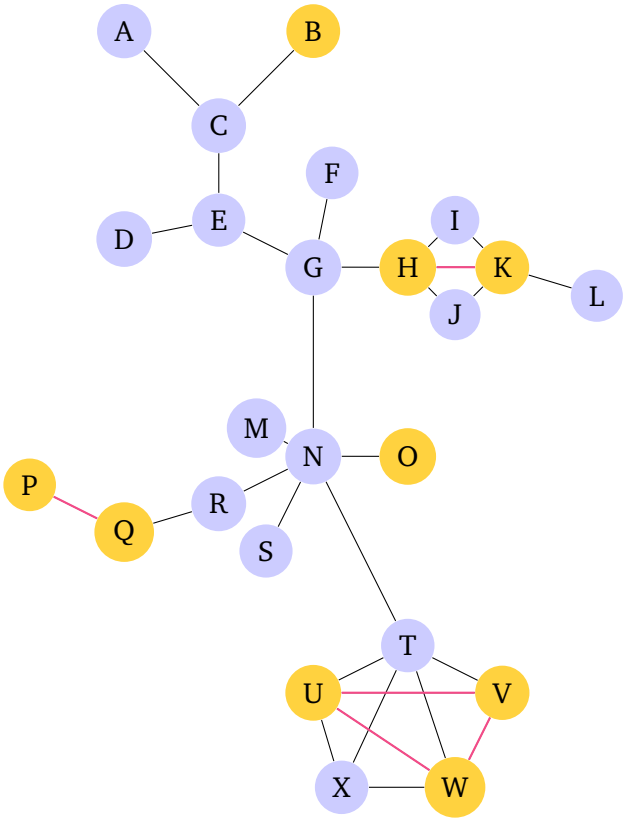


Fig. 6.9: Example of “induced subgraph sampling". 9 nodes are selected (in yellow) among the 24 in the graph population. The edges observed (in red) are those connecting two nodes of the sample.

The selection probabilities for vertex i and edge $\{i, j\}$ are:

$$\pi_i = \frac{n}{N_v}$$
$$\pi_{\{i,j\}} = \frac{n(n-1)}{N_v(N_v-1)}$$

with N_v the number of vertices. Note that N_v has to be known for the inclusion probabilities to be exactly computed. If N_v is not known, inclusion probabilities can still be estimated by replacing N_v with an estimate. Among other solutions, [Kol09] proposes capture-recapture methods.

Incident subgraph sampling Incident subgraph sampling is the edge counterpart of induced subgraph sampling. We select n edges using simple random sampling without replacement, and observe all vertices that are incident to the edges selected in the sample. Figure 6.10 illustrates this type of sampling.

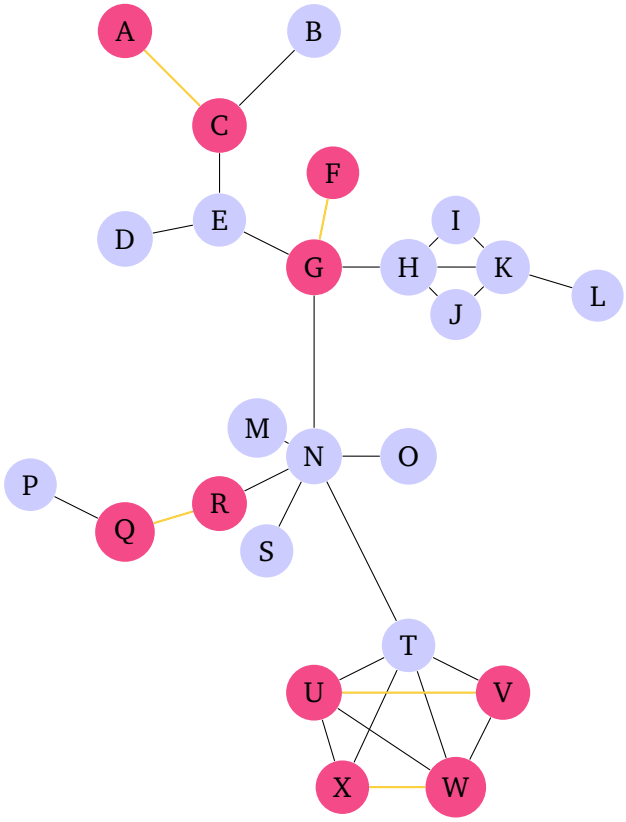


Fig. 6.10: Example of “incident subgraph sampling". 5 edges are selected (in yellow) among the 29 edges in the population graph. The observed nodes (in red) are those that are linked by at least one edge in the sample.

For incident subgraph sampling, inclusion probabilities for the edges are:

$$\pi_{i,j} = \frac{n}{N_e}$$

with N_e the number of edges (which, just like N_v in the case of induced subgraph sampling, can be estimated). A vertex is included in the final sample if and only if one of the edges it is attached to is selected in the initial sample. Thus the vertex inclusion probabilities are:

$$\pi_i = 1 - \frac{\binom{N_e - d_i}{n}}{\binom{N_e}{n}} \text{ if } n \leq N_e - d_i$$

with d_i the degree of vertex i and $\pi_i = 1$ if $n > N_e - d_i$.

For both incident and induced subgraph sampling designs, it is possible to substitute simple random sampling with other more complicated designs. Inclusion probabilities are easily computed as long as the sampling algorithm chosen allows for computation of second-order inclusion probabilities (with either a closed form expression or an algorithm for numerical computation).

As it was noted in example of figure 6.8, such sampling designs are often not able to capture simple properties with good quality, even when using weighted estimates. Hence the idea of using extension sampling designs (see section 6.3.4).

6.3.4 Extension sampling algorithms

Starting from this section we concentrate on vertex-based sampling designs, although equivalent in edge-based sampling designs could be easily constructed. We describe "extension sampling designs", which consist in selecting a first-stage sample of vertices, which serves as a basis to select other vertices in a second stage.

The case of full link observability makes it easier to select second-stage units. If links can only be discovered partially, then inclusion probabilities will have to be approximated, which lowers the quality of the estimates (see paragraph 6.3.2).

Link tracing sampling Link tracing sampling consists in selecting two samples of vertices: the "source vertices" S and the "target vertices" T (the two samples being disjointed). Then for each pair of vertices $(s_i, t_j) \in S \times T$, each edge and vertex in the path between s_i and t_j is included in the final sample. An illustration is shown in figure 6.11.

Inclusion probabilities for link tracing design are hard to compute, and several methods exist. [Dal+06] show that inclusion probabilities can be approximated by:

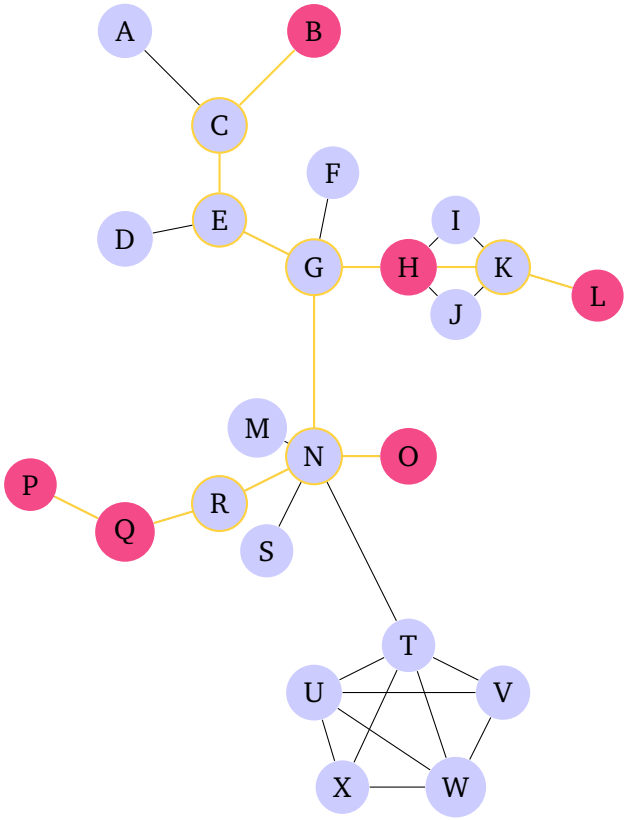


Fig. 6.11: Example of “link tracing sampling". Sampled source nodes (in yellow): B,H,P. Sampled target nodes (in yellow): O,L,Q. Observed nodes (in red) along the paths from sources to targets: C,E,G,N,K,R. Observed edges colored in red.

$$\pi_i \approx 1 - (1 - f_s - f_t)\exp(-f_s f_t b_i)$$
$$\pi_{\{i,j\}} \approx 1 - \exp(-f_s f_t b_{i,j})$$

where f_s and f_t are source and target sampling fractions, b_i is the vertex betweenness centrality of vertex i and b_{ij} is the edge betwenness centrality of edge $\{i, j\}$.

[VT17] use Markov-Chain methods to obtain approximations of the Rao-blackwellized version of the sampling estimates.

Adaptive and snowball sampling

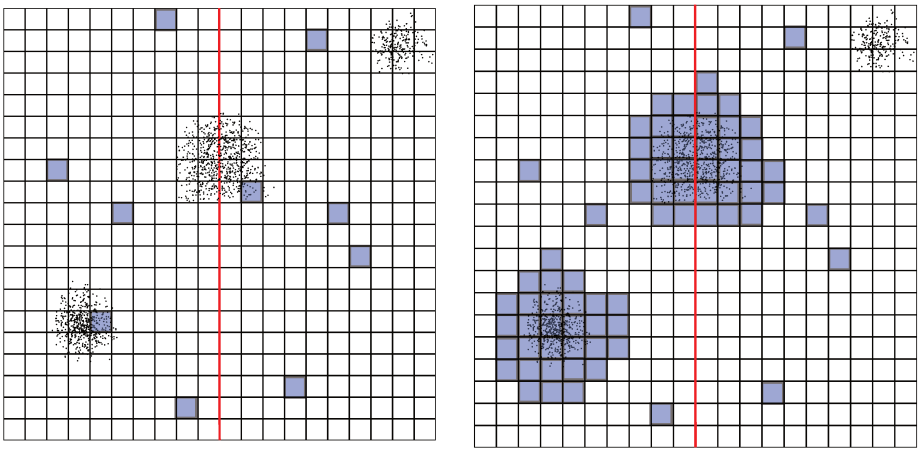
Adaptive and snowball sampling both consist in selecting at the second stage units that are linked to units at the first stage. These sampling techniques have been used for several years in social sciences, in the case of partial link observability ([Hec97], [HJ01], [Lav12]). We describe these designs more precisely in chapter 7.

Adaptive sampling consists in selecting in the second stage all elements linked to the elements of the first stage that possess a certain characteristic (typically, belonging to a certain community). For this reason, it is often illustrated using visuals that remind the video game "minesweeper" (see figure 6.12).

k-snowball sampling consists in adding in the second stage all units that are located within a distance smaller than k of a unit selected in the first stage. An illustration of 1-snowball sampling is shown in figure 6.13.

6.4 Random graph models

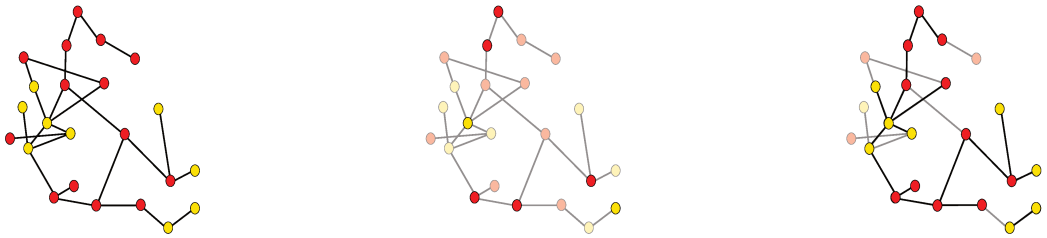
In this section we describe very shortly some probabilistic frameworks to model graphs as statistical objects. We focus on what properties of real-world graphs we can hope to replicate with these different frameworks.



(a) First sampling phase for the stratified adaptive design

(b) Extension sampling phase for the stratified adaptive design

Fig. 6.12: Illustration of stratified adaptive sampling. The population is described as a 20 $\times$ 20 grid (and two strata partitioning the population in two, marked with the vertical red line). The networks of units are represented by aggregated little dots. In the first sampling phase (a), five squares are selected in each stratum (marked in light blue). In the second phase (b), the sample is extended with the networks (and their sides) that could be reached using the first phase sample.



(a) An example graph population (units bearing different characteristics of interest are depicted with different colors)

(b) Sample selected at the first stage

(c) Addition of 1-stage snowball extension (note that units of both colors can be included in the second stage)

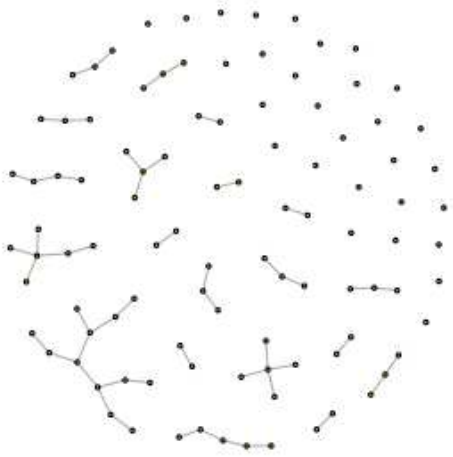
Fig. 6.13: Illustration of snowball sampling

6.4.1 The Erdős-Rényi (random graph) model

[ER59], [ER60] and [ER61] introduced the first random graph model. For this reason, "random graph model" traditionally refers to the Erdős-Rényi model, although the most widely used formalization of the model is due to [Gil59][3].

A collection $G(n, p)$ is composed of all graphs G of order N that may be obtained by assigning an edge independently to each pair of distinct vertices with probability p. Figure 6.14 shows examples of Erdős-Rényi graphs with different parameters.

(a) Erdős-Rényi graph with 100 vertices and parameter p=0.05. The graph is fully connected, mean degree is 5.12



(b) Erdős-Rényi graph with 100 vertices and parameter p=0.01. The graph has 49 different connected components, mean degree is 1.02

Fig. 6.14: Example of Erdős-Rényi graphs with different parameters

Although these random graphs are very important in mathematical theory, they are not really useful in practical graph modeling, because they are not able to replicate the heavy tailed distribution of degrees illustrated in figure 6.3.

6.4.2 The Barabási-Albert (preferential attachment) model

In order to correct this, [BA99] introduced a model where new units are preferentially attached to units that have a high degree. The Barabási-Albert graphs thus present heavy-tailed degree distributions, which is closer to real graphs. Figure 6.15 shows an example of a Barabási-Albert graph.

Unfortunately, the density of Barabási-Albert graphs are much lower than what we observe in most real life graphs.

[3] Although Erdős-Rényi model and Gilbert's framework are not exactly the same, they are equivalent for large values of N

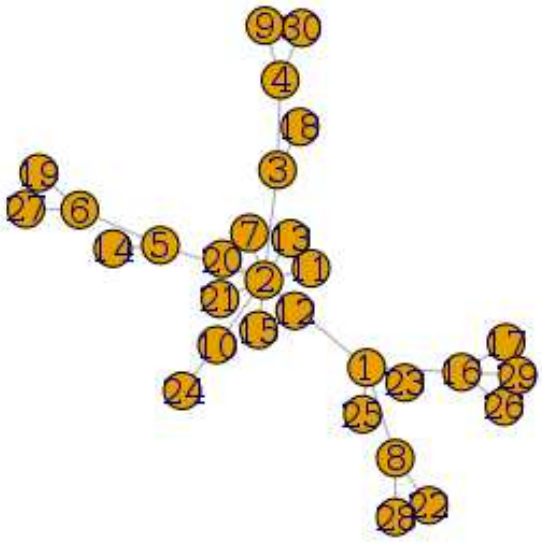


Fig. 6.15: Example of an undirected Barabási-Albert graph with 30 vertices. Vertices that appeared earlier (indicated by a low vertex index) have a higher attachment probability (and thus generally a higher degree). Mean degree for the graph is 1.93, kurtosis of the degree distribution is 10.4.

6.4.3 The Watts-Strogatz (small world) model

Most real-world graphs exhibit a high density (small world property). In other terms, most nodes are not neighbors of one another, but the neighbors of every node are likely to be neighbors of each other. More precisely, the path length between two randomly chosen nodes has to grow proportionally to the logarithm of the order of the graph N.

[Bac+12] conducted a study on modern social networks (using Facebook and Twitter data), showing that the average maximum path length for most units is lesser than 4, although a small number of isolated (and somewhat inactive) users can show maximum path lengths greater than 10. The degree distributions of small-world networks are also heavy-tailed.

[WS98] created a random model (the small-world model) that is able to recreate this property. Figure 6.16 shows an example of a small-world graph generated with this model.

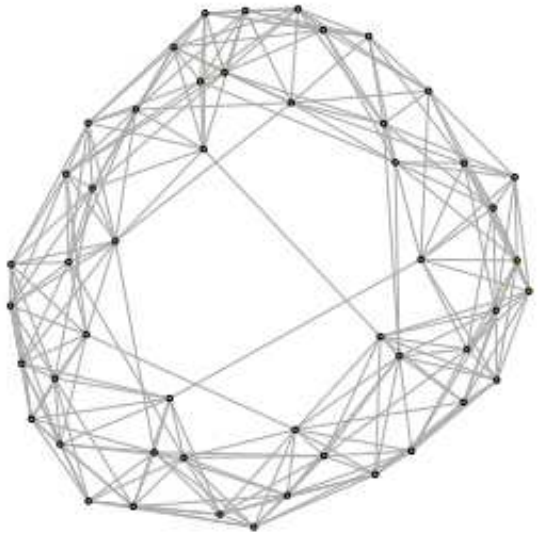


Fig. 6.16: Example of a small-world graph generated with the Watts-Strogatz model, with 50 vertices. Mean degree is 10. Global clustering coefficient is 0.56.

6.4.4 The forest-fire model

The forest-fire model ([Les+05]) intends to reproduce small world properties, heavy-tailed degree distributions, but also densification (clustering coefficient networks tend to increase in time), shrinking diameters, and apparition of communities.

The model is implemented in the R package [CN06], and we will take advantage of its capacity to replicate many real-life properties in chapter 8.

6.4.5 Other graph models

Some other models are out of the scope of this article. Among other very useful models are Stochastic Block models (see [Dec+11] for a discussion on the asymptotics of such models), a generalization of Erdős-Rényi models which allow to construct communities of different shapes, and Exponential Random Graph Models (see [WF94]), which are very important for foundations of statistical inference on graph models.

7 Application: the Twitter graph

> *This work has appeard in the Proceedings of the Third Conference of the International Society for Nonparametric Statistics (ISNPS)*

—

With the rise of big data, more and more attention is paid to statistical network analysis. However, exact computation of many statistics of interest (such as clustering or centrality) is of prohibitive cost for big graphs. A way to solve this problem is to use statistical estimators instead of exact ones. A broad literature on model-based estimation exists, but these estimates cannot be used for quick computation of statistics of interest. Therefore, design-based estimates relying on sampling methods were developed specifically for use on graph populations. Reference libraries for statistical network analysis now implement variations of sampling designs.

In this paper, we test some sampling designs used by official statistics institutes to estimate quantities when characteristics of interest on the population can be modeled as networks. These sampling methods can be described as “extension" sampling designs. Unit selection happens in two phases: in the first phase, simple design such as Bernoulli sampling are used, and in the second phase, some units are selected among those that are somehow linked to the units in the first-phase sample.

We test these methods on Twitter data because its size and structure typically match the structure of big social networks for which such methods would be very useful. Also, statistics on the Twitter graph are used in many papers in computer and social science.

7.1 Introduction

7.1.1 Problem

With more and more businesses and public administrations producing larger raw datasets every day, statistical analysis of so-called “big data" has risen. Consenquently, more research in computer science and statistics have focused on methods to tackle such problems. However, a significant part of datasets that fall under the general “big data" framework are actually graphs. Graph-specific data analysis has applications in domains as diverse as social networks, biology, finance, etc. Since the rise of the web, statistical literature for networks has been growing rapidly, especially in the field of model-based estimation. In the past 20 years, models such as Barabási-Albert ([BA99]), Watts-Strogatz ([WS98]), stochastic block models ([NS01]) and many others have induced huge progress in understanding probabilities and statistics for various cases of networks. Yet, model-based estimation is sometimes inconvenient. First, models cannot possibly perform well on all statistics of real-life graphs. Second, model-based estimation obviously requires a fine tuned choice of model before being able to produce statistics. Finally, even when a specific model exists and fits the analysis of a specific graph, computation can be cumbersome. This is a motivation for developping design-based estimates, which have received little attention from a purely statistical point of view. One of the most efficient tools for statistical graph analysis, snap - which has a very convenient Python interface ([LS14]), uses sampling methods to compute various statistics of interest ([LF06]).

There are some domains in computer science and social sciences that focus exclusively on social networks analysis. A large part of this literature analyses published content rather than estimates regarding quantification or qualification of accounts engaged in the analyzed content. On Twitter, this means the focus is more on tweets than on who tweets. Most recent analysis and inference based on Twitter data used the Streaming API[1] ([Bur+15]). Tweets matching a research criterion were collected in real time for a few days, and then analyzed. Many studies perform sentiment analysis on the harvested tweets. Some studies try to unravel political sentiment based on their Twitter data ([Con+11], [Tum+10]), for example to try and predict election outcomes. Very recently, election prediction based on Twitter data was proven not more accurate than traditional quota sampling, when trying to predict the outcome of the UK 2015 general election ([Bur+15]). Other topics are very diverse and include for example stock market prediction ([Bol+11]). Non-academic studies based on tweets data are also numerous. Most are made by market research companies, often to measure “engagement" by users to a brand.

[1] `https://dev.twitter.com/streaming/overview`

However, such analyses are based on biased estimates. First, the Streaming API is often preferred by researchers over the Rest API[2]. The Streaming API provides tweets in real-time matching a certain query, thus allowing collection of a huge amount of data. However, when the data size exceeds a certain threshold, only a fraction of tweets are selected, but Twitter does not disclose the sampling design used to select these tweets. The Rest API allows the collection of a much more limited number of tweets, but the selection is made by account and not by a query on tweets. A statistical analysis using the Rest API is thus much closer to the survey sampling paradigm. In fact, it is possible to derive unbiased estimators using the Rest API. Using the Streaming API to perform a global analysis on the scope of the Twitter users (or *a fortiori* if estimates are to be extrapolated to a larger population, such as in the case of the prediction of election outcomes) can lead to unbalanced profiles of users. When the selection is made by tweets, users tweeting much less about a certain subject have a much lower probability of being selected than users who tweet a lot about the same subject. This is even more telling when so-called “bots" (automated accounts set up to regularly write tweets about a pre-defined subject) account for a non-negligible fraction of the total tweets ([Fer+14]). When selection is made using a search query, it is very difficult to assess the selection probability, and when not all the tweets are output a large number of probabilities can be equal to 0. As [Mus+11] noticed, in many cases the users that precisely have a low selection probability account for the greater variability in estimates. [Slo+13] studies precisely these difference in user profiles used in Twitter-based estimations.

The goal of our work is to perform an estimate of the number of accounts tweeting on a specific subject, focusing on the accounts behind the tweets instead of the tweets themselves. We chose to work on the release of the trailer of the movie “Star Wars : the Force awakens" that occurred on October 19, 2015. According to “Twitter Reverb” data, this event generated 390 000 tweets in less than 3 hours[3].

This study is meant to be a proof of concept for further study. Regarding specific analysis on social networks, we think our analysis could be used joinlty with analyses based on tweets to shed a new light on data structures. For example, in the case of election prediction, it could be used to balance selection probabilities that are naturally skewed away from people who’re less likely to participate in Internet debates (which, unfortunately, is correlated to political opinions). In any other marketing context, this method could be used to weight tweets according to methods extending the “generalized weights share method" such as proposed by Rivers ([RB09], [LC01]). More generally, understanding how different sampling schemes

[2] `https://dev.twitter.com/rest/public`
[3] `http://reverb.guru/view/597295533668271595)`

behave on a huge graph that exhibits many of the particular properties of web and social graphs (see for example 7.1.2) could help us understand the benefits and disadvantages of these schemes. Estimation by sampling could be used to improve computation time and precision for descriptive statistics or even machine learning algorithms on graphs.

Here, we test two different methods of sampling adapted to graph populations. These two methods are in fact extensions of simple sampling designs. Each of these methods yield unbiased estimates. Theoretically, the precision of the estimates can be spectacularly improved with these extension designs. However, their practical use can be limited by our ability to collect the data on the graph. In this paper, no inference is made on any population outside the scope of Twitter users (ie the vertices of the Twitter Graph).

7.1.2 The Twitter graph

In computer science, the Twitter graph is said to show both “information network" and “social network" properties (see for instance [Mye+14]) :

- The degree distributions (both inbound and outbound) are heavy-tailed, resembling a power law distribution

- The average path length between two users is short

Thus, in the context of model-based estimation, the Twitter graph should be modelized by both a scale-free network (Barabási-Albert , [BA99]) and a small-world network (Watts-Strogatz, [WS98])

Let’s denote :

$$
\begin{aligned}
\mathcal{U} &= \text{population of interest} = \text{vertices of the graph} \\
N &= \#\mathcal{U} = \text{population size} \\
n &= \text{sample size} \\
y_k &= \text{number of times user k tweeted about the Star Wars trailer} \\
z_k &= \mathbb{1}\{k \in \mathcal{U}, y_k \geq 1\} \\
N_C &= \#\{k \in \mathcal{U}, y_k \geq 1\} \\
n_C &= \#\{k \in s, y_k \geq 1\} = \text{number of people in } s \text{ who tweeted about Star Wars} \\
T(Y) &= \sum_{k \in \mathcal{U}} y_k
\end{aligned}
$$

We also use graph-specific notations to make some formulas clearer

$$
\begin{aligned}
V &= \text{set of vertices } (= \mathcal{U}) \\
E_{ij} &= \text{set of edges linking vertices i and j}
\end{aligned}
$$

The goal is to estimate $N_C = T(Z)$, the number of users who have tweeted about the Star Wars trailer during the time span of the study.

For the sake of simplicity, we chose a very simple definition of our sub-population of interest, the Twitter users that tweeted about the Star Wars trailer. In our study, the "Star Wars fans" are detected because the tweets they wrote during the selected time span match a pre-defined list of words, mentions to other accounts, or hashtags. However, a more realistic setting would be to imagine that we have at our disposal a complex classifier f able to identify a broader definition of a "Star Wars fan" (for example a machine learning classifier based on Natural Language Processing or a community detection algorithm). Therefore our goal would be to estimate $N_C = \mathbb{1}\{k, f(k) = 1\}$, which is statistically speaking identical to the problem we deal with here. It is also worth noting that exhaustive computation of such a classifier would be very computationally intensive, even with a full access to the Twitter graph. Thus the use of a sample also makes practical computation of such statistical objects easier.

7.2 Sampling designs

7.2.1 Notations

Notations for sets and networks :

$$
\begin{aligned}
s_0 &= \text{initial sample} \\
C &= \bigcup_i r_i = \text{sub-population of units bearing the characteristic of interest (denoted)} r_i \\
s_c &= s \cap C \\
s &= \text{total sample i.e. } s_0 \text{ inflated with units of } C \text{ that can be reached} \\
s_{ri} &= s \cap r_i \\
C_k &= \forall k \in s, C_k = \mathbb{1}\{k \in C\} \\
\delta &= \text{Graph geodesic distance between two units in the network} \\
r_i^0 &= \{k, \delta(k, r_i)\}, \text{“side" of } r_i \\
s^0 &= \bigcup_i r_i^0 = \{k \in s, \delta(k, s_C) = 1\} \\
s_{ex} &= \text{elements which are neither in a network nor a side} = \{k \in s, \delta(k, s_C) \geq 2\}
\end{aligned}
$$

In this article, s_0 is selected using stratified Bernoulli sampling.

7.2.2 Bernoulli sampling

Poisson sampling consists in selecting in the sample s each unit $k \in \mathcal{U}$ with a Bernoulli experiment of parameter π_k, the first-order inclusion probability of unit k:

$$\forall k \in \mathcal{U}, \mathbb{P}(k \in s) = \pi_k$$

Second-order inclusion probabilities thus have a very simple expression:

$$\forall k \neq l \in \mathcal{U}, \pi_{kl} = \mathbb{P}(k, l \in s) = \pi_k \pi_l$$

Bernoulli sampling is Poisson sampling with equal probabilities ($\forall k \in \mathcal{U}, \pi_k = p$). Bernoulli sampling design is the most simple design that can be used and thus gives very simple formulae for estimators and variance estimators. One can refer to Särndal ([Sär+03]) for a more detailed presentation of Poisson and Bernoulli sampling. For the rest of this article, we'll use Bernoulli sampling as the primary sampling design on which we'll base other more complicated (and more efficient in terms of precision) designs: a stratified design, an adaptive design and a one-stage snowball design. We write:

$p =$ selection probability for each unit in the frame (each twitter user)

$q = 1 - p$

Poisson sampling is seldom used in survey sampling (at least in national statistics institutes), because it yields samples of variable size ($\#s$ is a random variable). A variable size can be problematic when data collection is costly (in most surveys by official statistics institutes, interviewers set up meetings with selected individuals either by phone or face-to-face). In our present case, it doesn't matter because the final goal is to use adaptive or snowball sampling, which will eventually yield a random final sample size. More, the cost of data collection is uniform across all units. We thus chose Poisson designs over simple random sampling, because the expressions for adaptive estimation and Rao-Blackwell estimates are much simpler.

On Twitter, users are assigned an id, ranging from 1 to $N \approx 3.3 \cdot 10^9$[4]. Some of the ids in this range are not assigned, but every new user is given an id greater than the last. In result, when selecting a Poisson sample using the ids in $[\![1, N]\!]$, part of the ids selected ($\approx 30\%$) will not be assigned to a Twitter user. Our sample is thus selected in a set greater than the population. Moreover, we are interested in only people who are *active* on twitter, i.e. users who tweeted in the last month. There are two reasons for which we consider these units: first we have a population margin to calibrate on (see 7.3.4) and second, it is the scope that is generally considered when stats on Twitter users are discussed for business purposes. Consequently, our frame over-covers our scope. But as out-of-scope units are perfectly identified, the over-coverage can be treated very simply (see [Sau15]) by using the restriction to a domain of the usual estimators. Two strategies can be used to produce estimates. If the total population on the domain $\mathcal{U}_d$ is unknown, we use Hájek estimators:

[4] as of december, 2015

$$\hat{T}(Y)_d = \sum_{k \in \mathcal{U}} y_{kd}$$
$$\hat{\bar{y}}_d = \frac{\sum_{k \in \mathcal{U}} y_{kd}}{\sum_{k \in \mathcal{U}} z_{kd}}$$

where:

$$y_{kd} = y_k \cdot \mathbb{1}\{k \in \mathcal{U}_d\}$$
$$z_{kd} = \mathbb{1}\{k \in \mathcal{U}_d\}$$

In our case, we know the total of the population on the domain N_d (see paragraph 7.3.4), so it is preferable to use:

$$\hat{T}(Y)_d = \sum_{k \in \mathcal{U}_d} y_k \tag{7.1}$$

Using estimator (7.1), all the estimators will work just like a restriction of the sample data frame on the scope. From now on, all the estimators are implicitly restricted to the scope domain unless stated otherwise.

The fact that some of the units sampled are finally out of scope leads to a final true sample size that can vary, which is another argument for not using fixed-size sampling design. For the simple Bernoulli design, our initial expected sample size (i.e. not taking the scope restriction into account) is **20000**:

$$p = \frac{20000}{3300000000} \approx 6.1 \cdot 10^{-6}$$

7.2.3 Stratified Bernoulli sampling

A usual way to improve the precision of the estimates is to stratify the population. We write:

$$\mathcal{U} = \mathcal{U}_1 \oplus \mathcal{U}_2$$

and draw two independant samples in $\mathcal{U}_h,\ h = 1, 2$ (in our case, two Bernoulli samples). With this design, the variance of the estimators writes:

$$\mathbb{V}(\hat{T}) = \sum_h f_h(S_h^2)$$

If strata are wisely selected, they should be homogeneous so that the S_h^2 are much reduced in comparison to the S^2 which determines the variance of the Horvitz-Thompson estimator under the simple Bernoulli design (see paragraph 7.4.1). Here, we try to estimate a proportion (alternatively a sub-population size), so we have:

$$f_h(S_h^2) = (1 - \frac{n_h}{N_h})\frac{N_h}{N_h - 1}\frac{p_h(1 - p_h)}{n_h}$$

The goal is thus to constitute strata where the probability of tweeting about the “Star Wars : The force awakens" trailer is more or less equal among users.

We divide the population of twitter users into two strata:

1. The users who follow the official Star Wars account (“@starwars") (1654874 followers as of October, 21, 2015)

2. The users who don’t

We use Bernoulli sampling on each stratum independently, but with different inclusion probabilities, so that users who follow the official Star Wars account are “over-represented" in the final sample. By doing this, we hope to increase n_C the

number of units who tweeted about Star Wars in s, because users who follow the official Star Wars account are more likely to be interested in tweeting about the new trailer.

In order to be able to compare the precision of the stratified design with the simple Bernoulli, we keep an expected final sample size of $n = n_1 + n_2 = 20000$. But we have to think about how to allocate these 20000 units between the two strata. We choose (again, in expected sample sizes):

1. 9700 units in stratum 1

2. 10300 units in stratum 2

which corresponds to a Neyman allocation ([Ney34]) supposing that approximately half of the people who tweeted were following the official @starwars account and that each person who tweeted about “The force awakens" did it three times:

$$n_{sw1} = 50000$$
$$n_{sw2} = 50000$$

which gives:

$$p_1 = 3.3\%$$
$$p_2 = 0.15\%$$

and then, the Neyman allocation writes:

$$n_1 = n\frac{N_1 S_1}{N_1 S_1 + N_2 S_2} \approx 9700 \tag{7.2}$$
$$n_2 = n\frac{N_2 S_2}{N_1 S_1 + N_2 S_2} \approx 10300 \tag{7.3}$$

The quantities in equations (7.2) and (7.3) are approximations, but this does not matter much, as it is well known that the Neyman optimum is flat (see for example

[MAR] for an illustration of this property). Thus, even if the approximated allocation is slightly shifted from the optimal one, the variance of the Horvitz-Thompson estimator will still be very close from the optimal variance.

7.2.4 One stage snowball sampling

For a vertex i, let's denote:

$$B_i = \{i\} \cup \{j \in V, E_{ji} \neq \emptyset\}$$
$$A_i = \{i\} \cup \{j \in V, E_{ij} \neq \emptyset\}$$

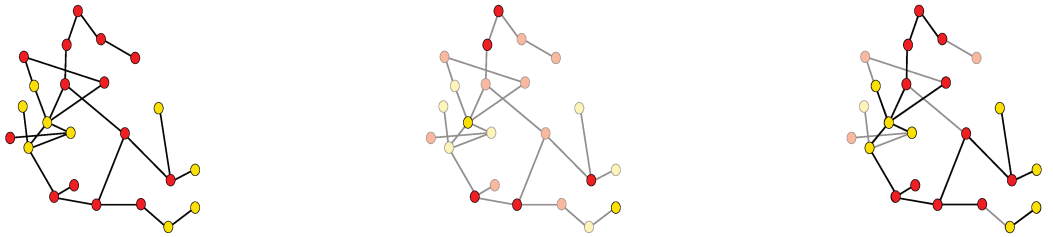
B_i is called the set of vertices adjacent before i, and A_i the set of vertices adjacent after i. In the case of the Twitter graph, B_i is the set of the users *following user* i and A_i is the set of users *followed by user* i.

The snowball design consists in selecting:

$$s = A(s_0)$$

Contrary to adaptive sampling (see paragraph 7.2.5), the sample is enhanced with the friends of every user $k \in s_0$, even if $k \notin C$. In this design, we also do not care about symmetric relationships. Another difference with the adaptive design is that we cannot remove out-of-scope units prior to the sample enhancement, as out-of-scope units may have in-scope units in their friends. Figures 7.1a, 7.1b and 7.1c describe the snowball sampling design.

As decribed in paragraph 7.1.2 and in [Mye+14], the Twitter graph is highly central and clustered. Thus, the number of reachable units *via* the edges of the units is s_0 is huge. Mean number of friends for every unit is estimated from the stratified sample to be approximately 75 for units in stratum 2 and a little more than 400 for units in stratum 1. In order to obtain final samples s of comparable sizes, we base the snowball design on a smaller stratified Bernoulli sample of expected size **1000 units**, with an allocation proportional to the Neyman described in paragraph 7.2.3 : $p_{S1} = 485/1654874 \approx 2.9 \cdot 10^{-4}$ and $p_{S2} = 515/33000000 \approx 1.6 \cdot 10^{-8}$. The final sample size is **159957**, which is much higher than we expected, probably due to



(a) Graph population $\mathcal{U}$. Elements of community C are depicted in yellow

(b) Sample s_0.

(c) Addition of 1-stage snowball $A(s_0)$

Fig. 7.1: Illustration of snowball sampling

the large tail distribution of the degree for units of stratum 1. With such a degree distribution, mean number of contacts is not very informative about the distribution of the sample size. Also, sample size has a very high variability.

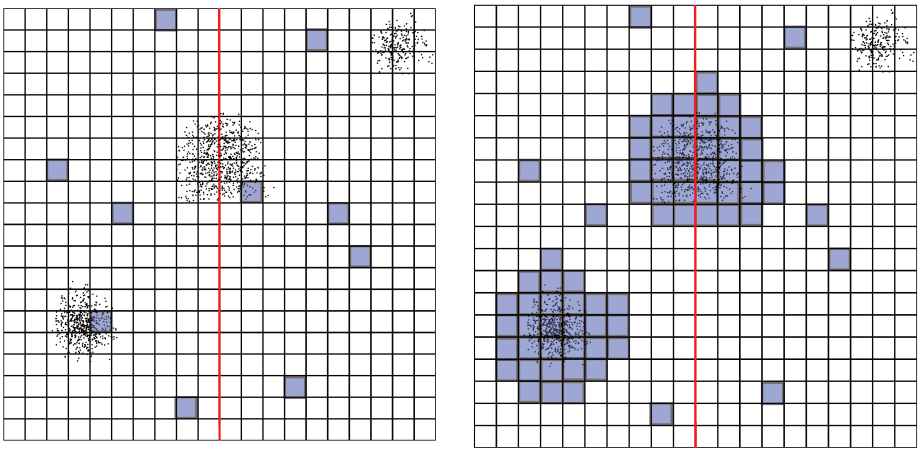
This definition of one-stage snowball sampling can easily be generalized to n-stage snowball sampling by including in the sample all units that have a shorter path than n to any unit in s_0. n-stage snowball sampling, although fairly simple to implement, can lead to very complex estimators for $n \geq 2$ (see [Kol09] for a more general discussion). But even more problematic in our case is that we saw in paragraph 7.1.2 that the average path length for the Twitter is very small (≈ 4.5). A precise analysis of characteristics of the Twitter graph ([Mye+14]) even shows that the distribution of path length is somewhat platykurtic. Thus, selecting a n-snowball sample could lead to huge sample sizes, reaching almost every units for very small values of n. For these reasons, we limit ourselves to one-stage snowball sampling (see 7.3.2).

7.2.5 Stratified adaptive cluster sampling

In our problem, the population is made of the vertices of a graph bearing a dummy characteristic. One modality of the characteristic is supposed to be rather rare in the population, and the units bearing it should be more likely to be linked to one another. In other terms, we suppose that users who follow people who tweet about Star Wars have a higher propensity of also tweeting about Star Wars.

Adaptive cluster sampling (first described in [Tho90]) consists in enhancing the initial sample s_0 with all units for which $y > 0$ (i.e who tweeted about the Star Wars Trailer) among the units who are connected to the units of s_0. As all units of interest

who are connected with to a unit in s_0 will be added to the final sample s, adaptive sampling is a particular case cluster sampling, the clusters being the networks of units having tweeted about the Star Wars trailer. Units of s_0 who didn't tweet about Star Wars won't have any other unit from their network added to the sample, but they can be seen as a 1-unit cluster. Once a person who tweeted about star wars is found in the initial sample s_0, many more can be discovered, which resembles the gameplay of the famous "minesweeper" videogame, and is often depicted as such in the literature (see figure 7.2).



(a) First sampling phase for the stratified adaptive design

(b) Extension sampling phase for the stratified adaptive design

Fig. 7.2: Illustration of stratified adaptive sampling. The population is described as a 20 $\times$ 20 grid (and two strata partitioning the population in two, marked with the vertical red line). The networks of units are represented by aggregated little dots. In the first sampling phase (a), five squares are selected in each stratum (marked in light blue). In the second phase (b), the sample is extended with the networks (and their sides) that could be reached using the first phase sample.

The Twitter network is a directed graph. Let us consider a unit $i \in s_0, y_i = 1$. Following the logic of adaptive sampling explained in the previous paragraph, we should look for units who also tweeted about “The force awakens" by searching the friends and followers of i, and if such units were found, look for other units among the friends and followers of these units and so on till the entire network is discovered. But if we did this, the inclusion probabilities for any unit $k \in s$ would not only depend on the units of k's network but also on other units with edges leading to k. This typically cannot be estimated from sample data ([Tho98]), as there is no reason that any of these unit be included in the sample. Thus, we only select units who show symmetric relationships with units in s_0 or in their networks.

For huge networks, we could expect the final sample size to be much greater than the initial sample size, especially if the network is highly clustered (which is the case of the Twitter graph, see paragraph 7.1.2). However, the fact that the Twitter graph is directed imposes us to only look for symmetric relationship, which will limit the size of the networks s_{ri} added to s_0. Finally, in addition to the s_{ri}, we also include in s units who have symmetric relationships to units in s_C, but who are not in C themselves. These units are called the “sides" s^0 of the networks s_r. These units can be described as: $\{k \in s, \delta(k, C) = 1\}$. In general, the sides of the networks also contain valuable information and should be included in the estimations to improve precision (see [Dev12]).

In our case, we'll rely on the Bernoulli stratified design of paragraph 7.2.3 to select s_0: the final design is thus stratified adaptive cluster sampling ([Tho91]). Estimators for this design are developped in paragraph 7.3.3.

7.3 Estimates

7.3.1 Horvitz-Thompsom estimator

For the simple designs, the privileged estimator is Horvitz and Thompson's ([HT52]), which weighs the observations with the inverse of the inclusion probabilities:

$$\hat{T}(Y)_{HT} = \sum_{k \in s} \frac{y_k}{\pi_k}$$
$$\hat{\bar{y}}_{HT} = \frac{1}{N} \sum_{k \in s} \frac{y_k}{\pi_k}$$

For the Bernoulli design, it simply writes:

$$\hat{T}(Y)_1 = \frac{1}{p}\sum_{k\in s} y_k$$
$$\hat{N}_{C1} = \frac{n_C}{p}$$

For the stratified Bernoulli, we get:

$$\hat{T}(Y)_2 = \sum_{s\cap\mathcal{U}_{d1}} \frac{y_k}{p_1} + \sum_{s\cap\mathcal{U}_{d2}} \frac{y_k}{p_2}$$
$$\hat{N}_{C2} = \frac{N_1}{n_1} n_{C1} + \frac{N - N_1}{n_2} n_{C2}$$

7.3.2 One stage snowball sampling

Horvitz-Thompson estimator

Estimation is developped in Frank ([Fra77]). The Horvitz-Thompson estimator writes:

$$\hat{T}(Y)_3 = \sum_{k\in s} \frac{y_i}{1 - \bar{\pi}(B_i)}$$
$$\hat{N}_{C3} = \sum_{k\in s} \frac{z_i}{1 - \bar{\pi}(B_i)}$$

where $\bar{\pi}(B_i) = \mathbb{P}(B_i \subset \bar{s})$, the probability that no unit of B_i is included in s. As the sampling design for s_0 is stratified Bernoulli (in particular, the event of belonging or not to s is independant for each unit of $\mathcal{U}$):

$$\begin{aligned}\bar{\pi}(B_i) &= \prod_{k\in B_i} (1 - \mathbb{P}(k \in s)) \\ &= q_{S1}^{\#(B_i\cap\mathcal{U}_1)} \cdot q_{S2}^{\#(B_i\cap\mathcal{U}_2)}\end{aligned} \tag{7.4}$$

The computation of this probability does not use any particular knowledge of the shape of the graph.

Just like the classic Horvitz-Thompson estimator, the estimator for the one-stage snowball sampling is linear homogeneous, with weights $d_i = \frac{1}{1-\bar{\pi}(B_i)}$. This property will be particularly useful for calibration on margins (7.3.4).

Rao-Blackwell estimator

In the case of one-stage snowball sampling, [Fra77] shows that a Rao-Blackwell type estimator can be written in closed-form:

$$\hat{N}^*_{C3} = \sum_{k \in s} \frac{y_k}{\pi_k} \left[1 - \frac{\sum_{L \subset s} (-1)^{\#L} \bar{\pi}(\{k\} \cup B(L \cup \bar{s}))}{\sum_{L \subset s} (-1)^{\#L} \bar{\pi}(B(L \cup \bar{s})} \right] \tag{7.5}$$

where the summation is done over all the subsamples L of s.

This Rao-Blackwell estimator is unbiased and as long as selection probabilities are not ill-defined (meaning that $\forall k \in \mathcal{U}, \bar{\pi}(B_k) < 1$), it performs better than the Horvitz-Thompson (7.4). However, there are $2^{\#s} - 1$ terms in each of two sums in equation 7.5. Computing each term of the sums implies computing the exclusion probabilities for as many B_k, which means at least knowing their sizes. This is extremely costly in time, and in our particular case in number of calls to the Twitter API. Thus, we prefer using $\hat{N}_{C3}$ over $\hat{N}^*_{C3}$.

7.3.3 Estimators for adaptive sampling

Horvitz-Thompson

As explained in 7.2.5, the adaptive sample is selected using only units who have symmetric relationships (so that the graph induced by a unit $k \in s_0 \cap C$ can be considered undirected).

$$\hat{T}(Y)_4 = \sum_{k=1}^{K} \frac{y_k^* J_k}{\pi_{gk}}$$
$$\hat{N}_{C4} = \sum_{k=1}^{K} \frac{n_{Ck}^* J_k}{\pi_{gk}}$$

where $k =$ networks in the population , y_k^* is the total of Y in the network k, n_{Ck}^* the number of people with $y_k \geq 1$ in the network k, $J_k = \mathbb{1}\{k \in C\}$ (ie intersection with sample) and π_{gk} is the probability that the initial sample intersects network k:

$$\pi_{gk} = 1 - (1 - p_k)^{n_g}$$

where $n_g = \#\{k \in g\}$.

Hansen-Hurwitz

Other design-unbiased estimators can be used. For example, Hansen-Hurvitz - like ([HH43]) estimators in this case would write:

$$\hat{T}(Y)_{HH} = \sum_{h=1,2} \frac{1}{p_h} \sum_{n=1}^{N_h} \frac{y_i f_i}{m_i}$$

where m_i is the number of units in the network that includes unit i and f_i is the number of units from that network included in the initial sample. This re-writes:

$$\hat{T}(Y)_{HH} = \sum_{h=1,2} \sum_{i=1}^{n_h} \frac{w_i}{p_h}$$

where the sum is over the elements of $s_0 \cap \mathcal{U}_h$ and with w_i the the average of y in the network r_i, which would give:

$$\hat{N}_{HH} = \sum_{h=1,2} \frac{1}{p_h} \sum_{i=1}^{n_h} n_{Ci}$$

Rao-Blackwell

In the case of adaptive sampling, it is very common to use the Rao-Blackwell estimator. This means using the conditional design $\mathbb{P}(s|s_0)$ (see for example [Dev12] or [Tho06] for more details on exhaustivity in sampling). In most cases, computing the Rao-Blackwell is computationally intensive and is achieved using Markov-Chain Monte-Carlo ([Tho06]). However, with Bernoulli stratified sampling it is possible to derive a closed form of the Rao-Blackwell ([Dev12]). Following [Tho98], we write:

$$\hat{T}(Y)_{RB} = \sum_{k \in s_{ex}} \frac{y_k}{\pi_k} + \sum_{k \in s^0} y_k + \sum_{k \in s_r} \frac{Y_r}{\pi_r}$$

where $Y_r = \sum_{k \in r} y_k$ denotes is the total of the variable of interest y in the network r. In our case, this leads to:

$$\hat{N}_{C5} = \sum_{s^0} \mathbb{1}\{yk \geq 1\} + \sum_{s_r} \frac{n_r}{\pi_r} = n^0 + \sum_{k=1}^{K} \frac{n_r}{1-(1-p)^{n_r}}$$

with $n^0 = \#s^0$.

The Rao-Blackwell for the Hansen-Hurvitz estimator also has a simple closed form that can be found in [Dev12].

7.3.4 Calibration on margins

The calibration estimator

Let us denote X a matrix of J auxiliary variables:

$$\forall j \in [\![1, J]\!], X_j = (X_{j1} \dots X_{jn})$$

whose values can be computed at least for all units of s. Let's suppose we also know the totals of these auxiliary variables on the population: $T(X_j) = \sum_{k \in \mathcal{U}} X_{jk}$. We write $T(X_j)$ the total of the j-th auxiliary variable and $T(X)$ the column vector of all the J totals $T(X) = (T(X_1) \dots T(X_J))'$.

In the general case, there is no reason that the estimated totals $\hat{T}(X)_{j\pi}$ using the Horvitz-Thompson weights match the actual totals $T(X_j)$. For two main reasons, one may want to use a linear homogeneous estimator that is calibrated on some auxiliary variables. First, often in official statistics, a single set of weights is used to compute estimated totals for many variables. Thus, it is often required that these weights are calibrated on some margins for sake of consistency. For example, $X = 1$ is often chosen as a calibration variable so that the size of the population N is estimated with no variance. Other typical calibration variables include sizes of subpopulations divided in sex and age groups so that the demographic structure of estimates is similar to the demographic structure of the population or quantitative variables linked to revenue - which is a main parameter of interest in most sociological studies, etc. Second, when estimating a variable Y using the calibrated weights w_k, the precision of the estimator will be increased if Y is correlated to one or more of the auxiliary variables X_j (see 7.4.4). This is the most interesting feature of calibration in the context of this study.

Let us write: X_s the $n \times j$ matrix of the values of the auxiliary variables and w a set of weights of a linear homogeneous estimator. Calibration on margins $X_1 \dots X_j$ consists in a searching a linear homogeneous estimator such that :

$$X_s' w = T(X) \tag{7.6}$$

supposing, of course, that this system has at least one solution. In general, when this linear systems has solutions, their number is infinite. To choose among these solutions, we look for weights w_k that are the closest to the Horvitz-Thompson

weights d_k wih respect to some distance G. Finally, finding the calibrated weights is equivalent to solving a linear optimization problem ([DS92]):

$$\begin{cases} \min_{w_k} \sum_{k \in s} d_k G(\frac{w_k}{d_k}) \\ \text{under constraint: } X_s' w = T(X) \end{cases}$$

Property 1 (Approximately Design Unbiasedness (ADU))**.** *The calibration estimator is approximately design-unbiased. Its bias tends to 0 as* $N \to +\infty$

The Horvitz-Thompson estimator is the most used estimator in survey sampling. One of its key properties is that it is design-unbiased. The calibration estimator is the closest estimator to the Horvitz-Thompson estimator (w.r.t a certain distance) that ensures the calibration equation (7.6). We can thus hope that it shares some properties with the Horvitz-Thompson estimator, in particular its unbiasedness. In fact, the unbiasedness of the calibration estimator holds asymptotically, i.e. when N goes with n to infinity (precise definition of the superpopulation model used to prove this asymptotic property is detailed in Isaki-Fuller, [IF82]).

Margins used for Twitter

For the calibration of our estimators $\hat{N}_C$, we use the following margins (i.e the columns of matrix X):

- $N =$ Total number of accounts who tweeted in the last month (quantitative):
- $T(Y) =$Total number of tweets about “Star Wars : The Force Awakens" between 10/25, 7:48 PM EST and 10/25, 10:48 PM EST (quantitative): 390000
- Number of verified accounts[5]
- Structure of users in sample in terms of number of followers

Many other margins could be added to the calibration process. Improvement in terms of variance of $\hat{N}_C$ is achieved as long as the margin X_j correlates with N_C and the calibration algorithm has a solution. Other calibration margins could include the number of (active) verified users, the number of tweets about “Star Wars: The Force awakens" per hour, etc. The geographical origin of the tweets is another variable

[5]https://support.twitter.com/articles/119135

that might strongly correlate to N_C. Twitter does dispose of such a variable, but it is seldom available. Imputation is not worth considering for this particular variable, because the number of accounts for which the variable can be used is very low. However, we could use Time Zones as a proxy for geographical origin, as the Time Zones are disclosed for every account. In general, in order to facilitate sampling estimates, Twitter could release calibration margins on a few characteristics.

Finally, we could calibrate on variables accounting for the graph structure, which probably correlate highly with N_C as well as a lot of other characteristics of interest of the Twitter graph. This could be achieved by finding a method for calibration on non-linear totals (see 7.6.2).

7.4 Variance and precision estimation

7.4.1 Simple designs

$$\mathbb{V}(\hat{T}(Y)_1) = \frac{1}{p}(\frac{1}{p} - 1)\sum_{k \in \mathcal{U}} y_k^2$$

which can be easily estimated by:

$$\hat{\mathbb{V}}_1(\hat{T}(Y)_1) = \frac{1}{p}(\frac{1}{p} - 1)\sum_{k \in s} y_k^2$$

Once n (which is random) has been drawn, it is common to work with the variance conditional to the sample size ([Til01]). This gives another variance estimator:

$$\hat{\mathbb{V}}_2(\hat{T}(Y)_1) = \hat{\mathbb{V}}_1(\hat{T}(Y) \mid \#s = n) = \frac{1 - \frac{n}{N}}{n}\hat{\sigma}(Y)$$

where $\hat{\sigma}(Y) = \frac{1}{n-1}\sum_{k \in s} (y_k - \bar{y})^2$.

For stratified Bernoulli, the total variance is the sum of the variance of the two independant designs in the strata. If the strata are built correctly, the dispersions $\hat{\Sigma}(Y)$ will be lower among each strata than in the total population, yielding a lower variance for the Horvitz-Thompson estimator. For example, using the simple plugin variance estimator, this writes:

$$\hat{\mathbb{V}}(\hat{T}(Y)_2) = \sum_{h=1}^{2} \frac{1}{p_h}(\frac{1}{p_h} - 1) \sum_{k \in s_h} y_k^2$$

In our study we chose, the variance estimators $\hat{\mathbb{V}}_2$, which finally write:

$$\hat{\mathbb{V}}_2(\hat{N}_{C1}) = \frac{N(N-n)(n-n_C)}{n^3(n-1)}$$
$$\hat{\mathbb{V}}_2(\hat{N}_{C2}) = N^2 \left[\frac{(N_1-n_1)(n_1-n_{C1})}{n_1^3(n_1-1)N_1} + \frac{(N_2-n_2)(n_2-n_{C2})}{n_2^3(n_2-1)N_2} \right]$$

7.4.2 Snowball sampling

An unbiased estimator of the variance of the Horvitz-Thompson estimator is given in [HT52]. One can easily rewrite the general case formula in the case of snowball sampling:

$$\hat{\mathbb{V}}(\hat{N}_{C3}) = \sum_{i \in s} \sum_{j \in s} \frac{z_i z_j}{\bar{\pi}(B_i \cup B_j)} \gamma'_{ij}$$

where:

$$\gamma'_{ij} = \frac{\bar{\pi}(B_i \cup B_j) - \bar{\pi}(B_i)\bar{\pi}(B_j)}{[1-\bar{\pi}(B_i)][1-\bar{\pi}(B_j)]} \tag{7.7}$$

The probabilities in (7.7) are theoretically easily computed using the formula (7.4). However, practically, it means browsing the whole set of vertices $B(s)$, which can be huge for highly central and/or clustered graphs (see 7.5.2). This variance estimator

is thus potentially highly instensive in time and computation power. In the case of the Twitter graph, this means a high number of calls to the API, which is unfortunately limited in number of calls. For too big sample sizes, we might have trouble computing the estimation.

7.4.3 Adaptive designs

Horvitz-Thompson

We use the variance estimator proposed by Särndal ([Sär+03]):

$$\hat{\mathbb{V}}(\hat{N}_{C4}) = \sum_{k=1}^{K} \sum_{k'=1}^{K} \frac{y_k y_{k'}}{\pi_{gkk'}} \left(\frac{\pi_{gkk'}}{\pi_{gk}\pi_{gk'}} - 1 \right)$$

where:

$$\pi_{gkk'} = 1 - \pi_{gk} - \pi_{gk'} + (1-p)^{n_{gk}+n_{gk'}}$$

Computing this variance estimator only requires the sizes of each network, which can be stored during the data collection process. In terms of number of calls, the variance estimator is thus much less demanding than the variance estimator in the case of the one-stage snowball (7.4.2).

Rao-Blackwell

Although the Bernoulli scheme conveniently yields a closed-form Rao-Blackwell estimation, it's not the case for the variance estimator. We have:

$$\mathbb{V}(\hat{N}_{C4}) = \mathbb{V}(\hat{N}_{C5}) + \mathbb{E}(\hat{N}_{C4} - \hat{N}_{C5})^2$$

The second term can be estimated without bias by selecting m samples. An unbiased estimator for $\mathbb{V}(\hat{N}_{C5})$ then writes:

$$\hat{\mathbb{V}}(\hat{N}_{C5}) = \hat{\mathbb{V}}(\hat{N}_{C4}) - \frac{1}{m-1}\sum_{i=1}^{m}(\hat{N}_{C5i} - \overline{\hat{N}}_{C4})$$

Of course, this method increases the number of calls needed to the graph API. In the case of Star Wars, we were already unable to get to the end of s^0 (see 7.5.2). If we'd had to generate several samples to estimate the variance of the estimation, it would obviously have been even harder.

7.4.4 Adjustment for calibration

Following [DS92], the variance for a calibrated Horvitz-Thompson for total Y writes:

$$\hat{\mathbb{V}}_c(\hat{T}(Y)) = \sum_{k\in s}\sum_{l\in s}\frac{\Delta_{kl}}{\pi_{kl}}(g_k d_k e_k)(g_l d_l e_l)$$

where: w_k are the weights of the calibrated estimator, d_k the weights of the non-calibrated estimator, $g_k = \frac{w_k}{d_k}$ and $e_k = y_k - \hat{b}' x_k$, residuals of the weighted regression (weights: d_k) of Y on $X_1 \dots X_j \dots X_J$ in s.

We recognize the general form of the Horvitz-Thompson estimator of variance (see [HT52]) which is used to construct variance estimators of paragraphs 7.4.1, 7.4.2 and 7.4.3, applied to the linear variable $g_k e_k$. This means that $\hat{\mathbb{V}}(\hat{N}_{Ci}), i = 1...5$ can be easily computed by re-using the formulae from 7.4.1, 7.4.2 and 7.4.3, and replacing the z_k by $g_k \cdot e_k$.

7.5 Results

7.5.1 Estimates

The variance of all estimators used in this paper is bounded by $\mathcal{O}(\frac{1}{n})$. It is thus always possible to reduce variance by increasing the sample size. In order to compare the sampling designs for their respective merits in reducing the variance of the estimation, we define the **design effect** Deff as:

$$Deff = \frac{\mathbb{V}(\hat{Y}_{design})}{\mathbb{V}(\hat{Y}_{design})} = \frac{\mathbb{V}(\hat{Y}_{design})}{(1 - \frac{n}{N}\Sigma(Y)^2)}$$
$$\text{with: } \Sigma(Y) = \frac{1}{N-1}\sum_{k \in \mathcal{U}} (y_k - \bar{y})^2$$

which compares the variance of the estimator to the variance of the estimator under a simple random sampling design of same size. The Deff is greater than 1 for designs that yield worse precision than the simple design (typically cluster or two-degree sampling), and lesser than 1 for designs that are more precise (typically stratified designs). Of course, real variance is impossible to compute, so the Deff can be estimated by:

$$\hat{Deff} = \frac{\hat{\mathbb{V}}(\hat{Y}_{design})}{\hat{\mathbb{V}}(\hat{Y}_{SAS})} = \frac{\hat{\mathbb{V}}(\hat{Y}_{design})}{(1 - \frac{n}{N}\sigma(Y)^2)}$$
$$\text{with: } \sigma(Y) = \frac{1}{n-1}\sum_{k \in s} (y_k - \bar{y})^2$$

The results of the estimators are presented in table 7.1:

Design	n	n_{scope}	n_0	$\hat{N}_C$	$\hat{CV}$	$\hat{Deff}$
Bernoulli	20013	3946		354121	0.231	1.04
Stratified	20094	9832		316889	0.097	0.68
1-snowball	159957	73570	1000	331097	0.031	0.60

Tab. 7.1: Estimates and characteristics for three different sampling designs

We can also note that the mean number of tweets about Star Wars according to the one-stage snowball sampling design is 1.18 ± 0.07. This low number suggests that automatic accounts are responsible for a very small amount, if any, of the total number of tweets on this subject ([Fer15]).

7.5.2 Size of the extensions

Because the Twitter network is highly central and clustered (a large number of users follow the a few number of highly-influential units), the friends of the units in s_0 have a huge number of followers. In our example, the file listing the ids of the followers of s weighed more than 250 GB (remember that the initial sample size, $\#s_0$, was less than 1000)!

It is important to note that the final sample size is random, just like for any clustered sampling design with non-constant sizes of clusters. When a unit is selected in s_0, there is no way to know in advance what the size of the clusters (networks) that includes it will be. Therefore, the final sample size $\#s$ can be highly variable. In the case of adaptive sampling, the subject studied featured highly clustered networks of Star Wars fans. This led to a very large number of users reached by the adaptive procedure. Due to the limits in number of calls imposed by the Twitter API, we were not able to finish collecting the stratified adaptive sample in less than a month. In order to prevent such issues, we could imagine sampling designs adjusting the selection probabilities as the collection of the units of s^0 goes along. Sampling design and estimation with such sampling designs is one of the future developments of our research we consider on this subject (see section 7.6.2).

Snowball sampling is unlikely prone to the same flaw, as number of units in final sample depends on the degree distribution. This distribution is often known *a priori*, and well modeled by a power law for all web and social networks graphs. Contrary to adaptive sampling, we are guaranteed that the extension will be complete after only one browse of the list of users followed by the units in s_0.

7.6 Conclusion and future work

7.6.1 Summary

This paper describes a design-based statistical method to estimate linear quantities on the Twitter graph based on users rather than queries. The method relies on sampling theory and in particular on developments of sampling theory for graphs. We tried two so-called “extension" designs: snowball sampling and adaptive sampling, which were first designed in official statistics to measure rare characteristics on a given population. We use them to try and measure the number of accounts having tweeted about the trailer of the Star Wars movie on the day of its release. Despite the event generating 390000 tweets in approximately 3 hours, the users responsible for these tweets are rare among the 1 billion Twitter users. Despite being unable to get through the whole adaptive sample because of the number of calls allowed by the Twitter API, the stratified snowball proved rather precise. Variance estimators were also computed, although they required a much higher number of calls.

7.6.2 Future work

Design effects The results obtained proved the extension sampling designs to be efficient methods to measure rare quantities with decent accuracy. Though the design effects shown in table 7.1 are smaller than 1, they are very similar to those of the stratified sampling design. In other terms, these extension sampling designs seem successful as a way to reach rare individuals ; however, in general, it appears that neither snowball nor adaptive sampling are generic methods that can help reduce variance by extending any sampling design. In other terms, variance for snowball and adaptive sampling cannot by bounded by something lower than $\mathcal{O}(\frac{1}{n})$, which is the order of magnitude of the variance of the Horvitz-Thompson estimator under a simple random sampling design. It is also interesting to note that we used an opportunity stratification, thus such performance cannot be guaranteed in general, even for networks presenting characteristics similar to the Twitter graph.

Relations to graph models Our future work on this aspect consists in studying correlations for different models commonly used in model-based estimation. This will hopefully lead to specifications determining cases when extension designs are most worth using. More generally, specific needs in specific cases require specific sampling designs. On graph, most useful statistics relate to degree, clustering, centrality, path length, etc. We suspect that optimal sampling designs depend very much on the graph model and will try characterizing the most suitable design for each statistic under each graph model. An interesting subject to investigate would be optimal sampling designs for graph that are accurately modeled using a mix of several models (like the Twitter graph).

More specific adaptive designs As we saw in 7.5.2, in the case of adaptive sampling, the final sample size can be huge if units bearing the characteristic of interest are highly clustered. In order to avoid being unable to collect the whole sample in this case, it might be useful to introduce a second-degree sampling design to select only a part of each network $k \in [\![1, K]\!]$. Estimation using such designs has been studied in [Tho06].

Estimation of graph totals Finally, in this article, we only dealt with linear estimators of totals and means. Estimation of totals of higher order are possible under the snowball and the adaptive sampling design. Such variables can include degree, clustering, centrality, etc. Following and idea developped by [Del+02] and [Les09], we will also try calibration on such totals when they are known.

8 Practical efficiency of snowball designs

8.1 Introduction

Snowball sampling has been long proven efficient to estimate size of communities in networks, particularly when these communities are ”hidden” and the extension design is used as a mean to recruit more units in the sample (see chapter 7).

In the context of big networks, unweighted estimates are often used (for example in [LS14]). We start by showing that these “plugin” estimates perform poorly and that weighted estimates are much preferable in terms of precision.

Then we try to assess whether snowball sampling can be more efficient than simple random sampling with the same sample size, for various variables and graph populations. We consider only the case of **undirected graphs**, weighted estimators and full link observability (paragraph 6.3.2).

We use Poisson designs as the initial sampling designs. In a first part, we use Bernoulli designs (equal inclusion probabilities for all vertices) and focus on the conditions for the extension sampling part to be efficient. In a second part, we fine-tune the initial sampling design by including proportional-to-size inclusion probabilities, hoping to yield more efficient estimators.

Efficiency of the snowball design very much depends on which variable is estimated and on the topology of the graph (see for example theorem 4 in [Fra77] for an illustration). We propose simple metrics that depend on the shape of the graph, and show that they are good proxies to decide whether a performing the extension phase is worth the effort.

We observe that snowball is often inefficient in terms of precision relative to sample size. Sometimes the relative inefficiency outweighs other benefits of the snowball sampling design: its computational efficiency, and its capacity to recruit sub-domains of interest that would otherwise be hard to capture.

We keep the same notations as in chapter 7 for this whole chapter.

8.2 Inefficiency of plugin sampling

Besides its popularity in social sciences, snowball sampling is also one of the most popular graph sampling methods in computer science and in programs used in the analytics industry (see [LS14]). In most cases, estimates are obtained either by reconstructing a graph from selected vertices and observed edges coming in and out of these vertices (induced subgraph sampling) and taking stats on the reconstructed graph ; or by directly estimating totals and means of graph stats with observed data on each vertex included in the sample. Observations are thus never weighted. [LF06] recommends a sampling fraction (i.e. size of sample relative to the size of the population) of at least 15% for these types of estimates to start being consistent.

Sample graphs with such sampling fractions are still huge, and we believe it is possible to do much better by using other methods. More, sampling theory tells us that plugin estimators are biased, and we expect the estimation errors to be huge for small sampling fractions. The goal of this paragraph is to show that very efficient estimates can be obtained simply by weighting the observations.

We use simulations to do so. We generate a superpopulation graph using a forest-fire model with forward burning probability of 0.25, backward burning probability of 1, and an ambassadors parameter of 1. A forest-fire model (described more precisely in [Les+05]) produces a directed graph with the following procedure: at each step, one vertex is added, and connects to a number of vertices already present in the graph equal to the ambassadors parameter. Then, for each vertex v already present in the graph, x additional edges from v to other vertices and y additional edges from other vertices pointing to v are added. x and y follow a geometrical distribution of parameters respectively equal to the forward burning probability and the backward burning probability. It is one of the best random models to replicate some properties of real-life networks (Heavy-tailed in- and out-degree distributions, existence of communities, densification in time according to a power-law rule, and shrinking diameter over time), but one of its main drawbacks is that most statistical quantities, even the simplest, are intractable in this model.

The number of vertices of the graph population is 100 000. Its number of edges is 178 097 (see paragraphs 8.3 and 6.2.2 for more details on why we chose a forest-fire model for this application). We select snowball samples of initial sample sizes varying from 0.01% to 7.5% of the population total. The resulting sample

sizes (including the extensions) range from 25 to 17700 (corresponding to sampling fractions between 0.00025% and 17%).

We then try to estimate the mean of some usual graph variables of interest denoted y (betweenness , closeness and eigenvector centralities, degree and page rank) using sample data (set of sampled vertices is denoted s). The two estimation methods are plugin (estimates denoted $\hat{\bar{y}}_1$) with unweighted data, and weighted estimates (denoted $\hat{\bar{y}}_2$) using weights w_k which are computed using the formulas developed in chapter 7 (Hajek-type estimates):

$$\hat{\bar{y}}_1 = \frac{1}{\#s} \sum_{k \in s} y_k$$
$$\hat{\bar{y}}_2 = \frac{1}{\sum_{k \in s} w_k} \sum_{k \in s} w_k y_k$$

We plot the Monte-Carlo Mean Squared Error of estimation for these variables $MSE(\hat{\bar{y}}_1)$ and $MSE(\hat{\bar{y}}_2)$ in figure 8.1:

$$MSE(\hat{\bar{y}}_1) = \frac{1}{100000} \sum_{k=1}^{100000} (\hat{\bar{y}}_{1,k} - \hat{\bar{y}})^2$$
$$MSE(\hat{\bar{y}}_2) = \frac{1}{100000} \sum_{k=1}^{100000} (\hat{\bar{y}}_{2,k} - \hat{\bar{y}})^2$$

We see that weighted estimates are always better in terms in terms of MSE than plugin estimates. Interestingly, weighted estimates with very small sample sizes almost always outperform plugin estimtors with the biggest sample sizes tested in our simulations. Contrary to what is suggested in the computer science literature, the estimation error for the plugin method does not decrease with sample size.

We also focus on the biases of these estimates:

$$Bias(\hat{\bar{y}}_1) = \frac{1}{100000} \left| \sum_{k=1}^{100000} (\hat{\bar{y}}_{1,k} - \hat{\bar{y}}) \right|$$
$$Bias(\hat{\bar{y}}_2) = \frac{1}{100000} \left| \sum_{k=1}^{100000} (\hat{\bar{y}}_{2,k} - \hat{\bar{y}}) \right|$$

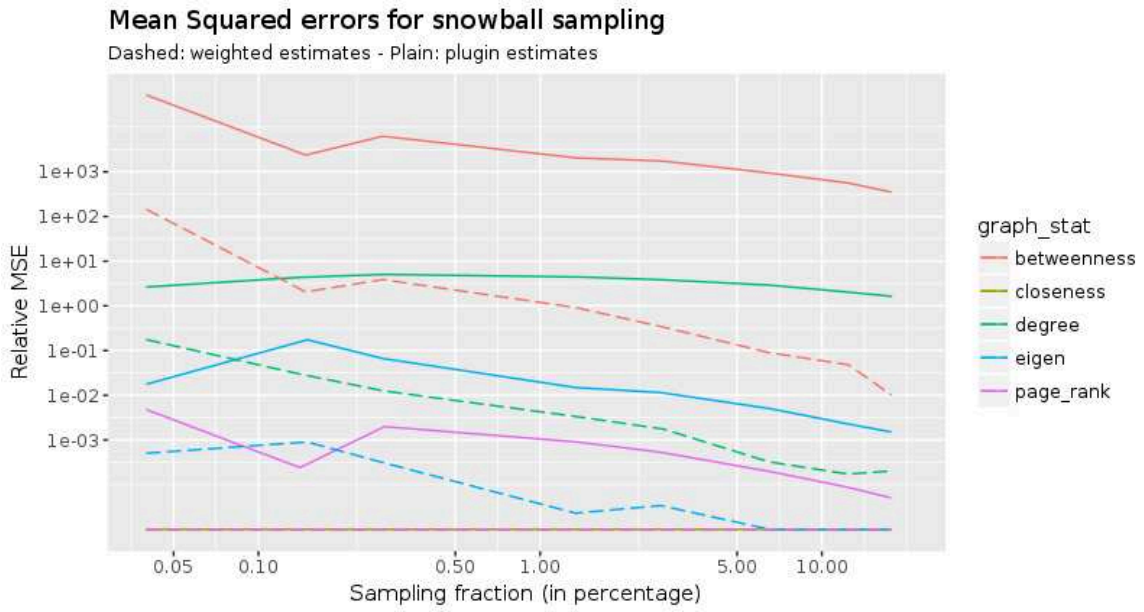


Fig. 8.1: Mean squared errors for plugin and weighted estimates (logarithmic scale)

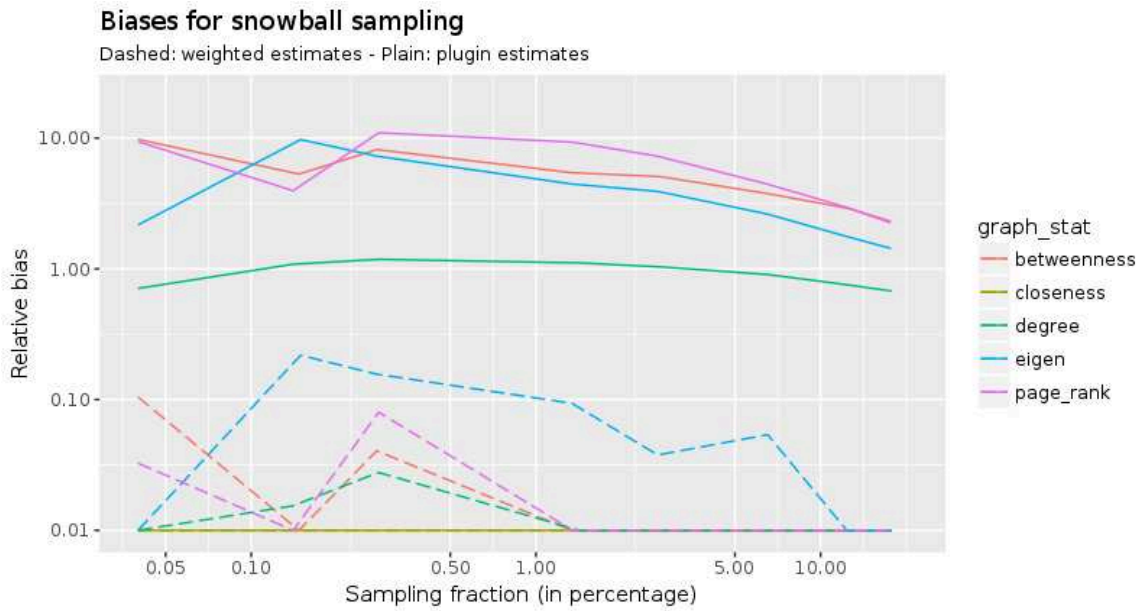


Fig. 8.2: Absolute value of bias for plugin and weighted estimates (logarithmic scale)

As expected, the graph shows that bias is order of magnitudes bigger for plugin sampling than in the case of weighted sampling, and does not decrease with size. Bias is also, in proportion, the bigger component of mean squared error, which makes it the primary cause of the inefficiency of the plugin estimates.

Finally, weighted estimates allows to select only a fraction of vertices of the graph and still offer better precision than plugin estimates with sampling fractions much closer to 1. Weighting in snowball sampling is thus an efficient paradigm in terms of

precision. We don't discuss computing efficiency in this section (refer to chapter 7) for practical computing considerations.

8.3 Measuring the efficiency of snowball extension

8.3.1 Design effect and estimators

We consider a graph $G = (V, E)$ and a (vertex-) variable of interest $Y = (Y_k)$ for $k \in V$. We denote N the order of the graph (the number of vertices in G). Y can be any kind of variable, for example a graph variable like described in chapter 6.2.2: the degree, a centrality, etc. (see further paragraph 8.3.2). The sampling design is the same as in chapter 7: we draw an initial sample using a Bernoulli design of expected size $n_0 = 50$, and then we extend the initial sample by adding the friends of all vertices that were initially selected (i.e the vertices that are at distance 1 of any vertex in the initial sample).

Estimators for $T(Y)$ and $\bar{Y}$ are computed by using formulas by [Fra77]:

$$\hat{T}(Y) = \sum_{k \in s} \frac{y_k}{1 - (1-p)^{d_k}} \tag{8.1}$$

$$\hat{\bar{Y}} = \frac{1}{\sum_{k \in s} \frac{1}{1 - (1-p)^{d_k}}} \frac{y_k}{1 - (1-p)^{d_k}} \tag{8.2}$$

where d_k is the degree of vertex k.

We write $V_{snow} = \mathbb{V}(\hat{T}(Y))$ and define:

$$Deff(\hat{(}T(Y))) = \frac{V_{snow}}{(1 - \frac{n}{N})\frac{1}{n}\sum_{k \in V} y_k^2}$$

which is the **design effect**, equal to the gain ratio in variance between estimating the total of Y on data collected with the snowball sample versus on data collected by simple random sampling of the same size. The idea is that collecting each unit has a

cost, and that simple random sampling algorithms are easy to implement. So, if a snowball design performs significantly poorer than a simple random design for the variables of interest, the simpler design might be a better option.

Efficiency of snowball sampling in [Fra77] Frank notices that graph population topology has a great influence on precision of sampling estimates:

Theorem 19 (Theorem 4, [Fra77]). *Denote B_k the set of vertices with an edge pointing towards vertex k, and π_k the inclusion probability of vertex k in the sample. If there exists a pair of distinct vertices i and j such that the graph and the initial sampling design satisfy:*

$$B_i = \{i\} \subset B_j and$$
$$\bar{\pi}(B_j)\pi_{ij} \neq \pi_{ij} - \pi_i\pi_j$$

then neither the snowball estimate with all sampling data (including the extension) neither the estimate using only the initial sample data dominates the other

This condition depends much more on the characteristics of the population graph than it depends on the initial sampling design. As noted before, it is thus difficult to tackle and we chose to do so using simulated graph populations (see paragraph 8.3.2).

Analogy with cluster sampling Snowball sampling can be seen as analog to cluster sampling with the clusters begin the subnetworks captured in the extension phase. The analogy is not exact because units selected in the extension phase can overlap. This depends on the local clustering coefficient, which is hard to compute for most random graph models. For example, in forest-fire models, the behavior of the local clustering coefficient is only known by simulation.

Still, the analogy could be useful to get insights on performance of the extension phase of the snowball design. Let's consider a formula for design effect of cluster sampling that can be found in [Ard06]. For a population of size N divided in M clusters, a sample is selected by drawing all units lying in m out of M clusters using simple random sampling. Then the design effect for the Horvitz-Thompson estimator of the total of the variable of interest writes:

$$Deff(\hat{T}(Y)_\pi) = 1 + \frac{N-M}{M-1}(1 - \frac{S^2_{intra}}{S^2}) + \frac{1}{\bar{N}(M-1)S^2}\sum_{i\in C}(N_i - \bar{N})N_i\bar{Y_i}^2 \quad (8.3)$$

where $S_{intra^2} = \frac{1}{N-M}\sum_{i\in C}\sum_{k\in\mathcal{U}_i}(y_k - \bar{Y}_i)^2$, $S^2 = \frac{N-M}{N-1}S^2_{intra} + \sum_{i\in C}\frac{N_i}{N-1}(\bar{Y}_i - \bar{Y})^2$

We name $\hat{Deff}_{cluster}$ the estimator by assimilating each set C_k of friends of k (ie $\forall k, C_k = \{v, \delta(k,v) = 2\}$) as clusters. It is important to note that this is not the expression of the true design effect of the snowball design in general, because the assimilation of friends of a vertex to clusters in inaccurate. In fact, if there is overlap between the friends of two units selected in the initial sample, this unit will be selected only once (sampling algorithms we consider are without replacement), and the clusters yielded by the two units of the initial sample would overlap. The correct "cluster" analogy would thus be random clusters, depending on the initial sample (i.e., non-random conditional to the initial sampling phase). Replacing with graph values in (8.3):

$$\hat{Deff}_{cluster} = 1 + \frac{\sum_{k\in V}(d_k - \bar{d})d_k\bar{Y_{(k)}}^2}{\bar{d}\sum_{k\in V} d_k(\bar{Y_{(k)}} - \bar{Y})^2} \tag{8.4}$$

with $\bar{Y_{(k)}}$ the mean of the statistic of interest among the friends of vertex k, d_k the degree of vertex k, $\bar{d}$ the mean degree of the graph.

Interpretation This anticipated design effect allows us to describe variables and topology that could correspond to an efficient snowball extension phase.

Let us examine what terms in this equation could help make the design effect the lowest possible. First, what range of values would mean that our design is good? Obviously, a design effect lower than 1 would be great, but such values are rare in practice. For example, in official statistics applications, geographical clusters are often used so that face-to-face interviews are less costly. This allows for a lower cost per unit compared to simple random sampling, and makes the cluster sampling design efficient, even if the design effect is greater than 1.

The situation with our graph framework is a little different: the real cost per vertex included in the sample is much flatter (even if we could imagine differences arising from implementations, particularly in the case of distributed algorithms, for example using Spark RDDs). But the real strength of snowball sampling lies in its ability to help detect communities or yield efficient estimators that are specific to some

communities (see paragraph 8.1). Similarly to what we observe in official statistics, we will thus consider designs that show Deffs slightly greater than 1 as efficient.

First, we notice that the anticipated design effect can be lower than 1 only if $\sum_{k \in V} (d_k - \bar{d}) d_k \bar{Y_{(k)}}^2$ is negative. This implies negative correlation between degree and high values of $\bar{Y_{(k)}}^2$. A sufficient condition for this to be realized is that the absolute value of the variable of interest and the degree are negatively correlated. This would allow "clusters" to be really homogeneous (small values of $(\bar{Y_{(k)}} - \bar{Y})^2$) in order to lower even more the design effect.

More realistically, we hope for small values of the numerator and high values for the denominator. On the denominator, it means that $(\bar{Y_{(k)}} - \bar{Y})^2$ has to be high, which means group of vertices with a geodesic distance of 2 have to be the most heterogeneous possible across the graph. Another way to say this is that there must be a lot of information contained in the friends of each vertex.

Finally, we notice that hoping for small values of $d_k - \bar{d}$ is not realistic because in real graphs, degree distribution are heavy tailed (see 6.2.2). So a small design effect will be obtained only for variables that are at least un-correlated to degree.

We can sum up what we're trying to find in those two conditions:

1. Weak (or negative) correlation of variable of interest with degree

2. High dispersion of the variable of interest among units with distance 2

We propose two metrics derived from $\hat{Deff}_{cluster}$ to assess whether snowball estimation for a graph variable can be efficient.

$$\hat{Deff}_1 = 1 + \frac{\sum_{k \in V} (d_k - \bar{d}) \bar{Y_{(k)}}^2}{\sum_{k \in V} d_k (\bar{Y_{(k)}} - \bar{Y})^2} \tag{8.5}$$

$$\hat{Deff}_2 = 1 + \frac{\bar{d} \sum_{k \in V} (d_k - \bar{d}) \bar{Y_{(k)}}^2}{\sum_{k \in V} (\bar{Y_{(k)}} - \bar{Y})^2} \tag{8.6}$$

We hope that these formulas are somewhat simpler to use than $\hat{Deff}_{cluster}$, closer to the conditions on variables and graph topology we listed earlier, and make it quicker to assess the quality of the snowball design than computing $\hat{Deff}_{cluster}$ or making simulations. In paragraph 8.3.2, we will use them to try to find variables compatible with good performance of snowball, and we will check their efficiency by doing simulations on different graphs - random and real. Note that both $\hat{Deff}_1$ and $\hat{Deff}_2$ can take negative values.

8.3.2 Variables of interest and population networks

Variables of interest We consider some well-known network statistics as variables of interest: degree, betweenness centrality, closeness centrality, local clustering coefficient, eigenvector centrality, page rank, maximum path length. For more precise definitions of these graph statistics, refer to paragraph 6.2.2. We could also consider blends of these network statistics, or variables that partly depend on a network statistic, for instance via a linear relationship (see example of figure 6.2).

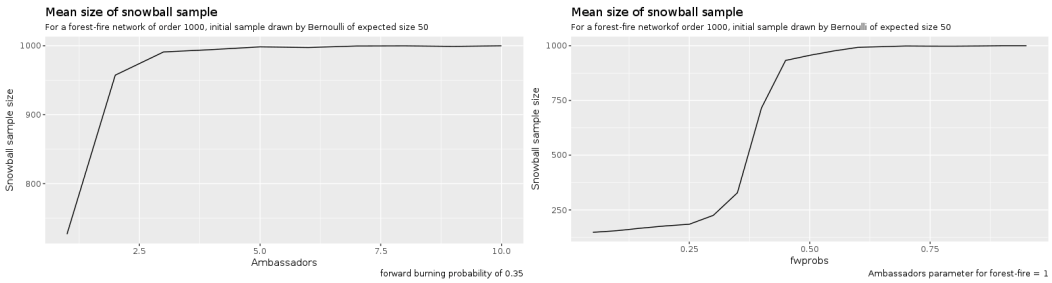
Let us now describe the graph populations that we consider. For all the applications, the initial sample size is $n_0 = 50$.

The forest-fire random graph model We chose first to run simulations on a forest-fire model (for various values of parameters of the model), because forest-fire models are able to replicate some essential properties from real graphs, like heavy-tailed degree distributions and short path lengths (see paragraph 6.4.4 and [LF06] for a more precise definition of this random graph model).

Nevertheless, snowball sampling captures too many units when the graph is too dense (see also chapter 7). In the case of forest-fire sampling, graphs with higher forward burning probabilities and higher number of ambassadors will be more dense. We start by making simulations to see what range of parameters yield acceptable snowball sample size (figure 8.3).

We observe two very steep portions on these curves - the snowball sample size suddenly increases and almost immediately the whole graph is included in the sample. Finally, we decide to only consider a number of ambassadors of 1 and forward burning probabilities from 0.01 to 0.30.

Then, we look at the values of the variables of interest for forest-fire models with parameters in this range. There are no closed-form expressions for most graph statistics in the forest-fire model. Some empirical studies have been conducted (for



(a) Ambassadors varying from 1 to 10

(b) Forward burning probabilities varying from 0.01 to 0.99

Fig. 8.3: Size of snowball sample for forest-fire model in function of forward burning probabilities and number of ambassadors. Initial sampling design: Bernoulli of expected size 50

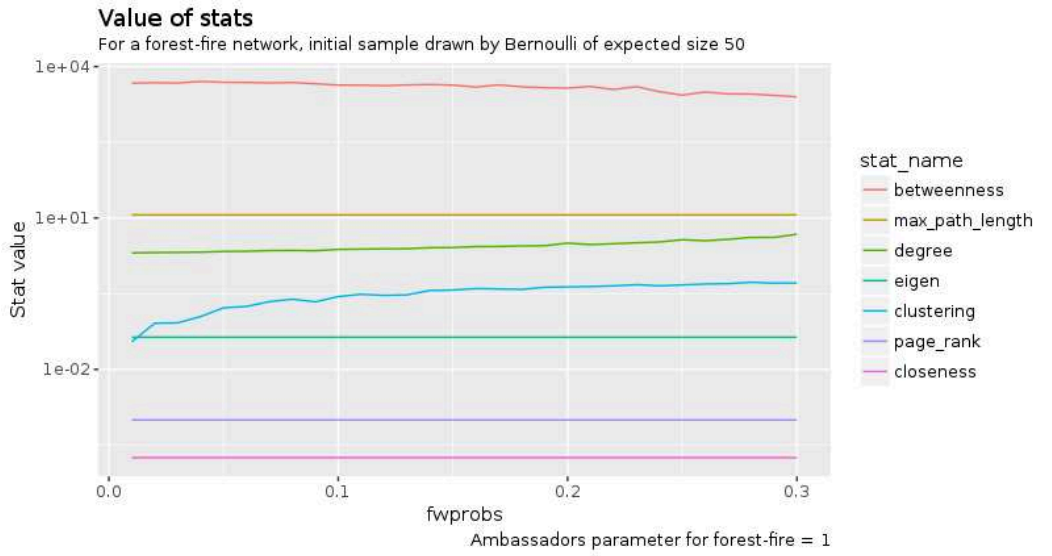


Fig. 8.4: Value of statistics of interest for the forest-fire model with forward burning probabilities between 0.01 and 0.30.

example [KM16], [Sax+17]). We proceed by simulations, our results are plotted in figure 8.4.

This graph shows that in this range of forward burning probabilities, the value of our graph stats of interest do not vary very much, except for the local clustering coefficient.

The Barabási-Albert model Then, we consider a Barabási-Albert model (see paragraph 6.4.2). This model is able to replicate scale-free distribution of degrees, but the graphs obtained are not dense, contrary to the forest-fire model.

Real graphs We also consider three real-life networks included in the R package [Csa15]. The *yeast* network consists of comprehensive protein-protein interaction

compiled by [VM+02] when studying regulatory network in cellular function. The *USairports* dataset is the network of passanger flights between airports in the United States (flights that happened in december 2010). This network was originally directed with edge directions corresponding to flight directions, but we only consider the undirected version of the graph where an edge indicates a connexion between two airports. Each edge is specific to a single carrier aircraft type. Multiple carriers between the same two airports are denoted by multiple edges. Finally, *immuno* is the undirected and connected network of interactions in the immunoglobulin protein. Its vertices represent amino-acids and an edge is drawn between two amino-acids if the shortest distance between their "C alpha atoms" is smaller than the threshold value of 8 angstroms ([Gfe07]).

Some descriptive statistics for these networks are listed in table 8.1.

Graph name	Vertices	Edges	Mean degree	Mean local clustering	Global clustering coefficient	Number of connected components	Power law fit
yeast	2617	11855	9,06	0,39	0,47	92,00	2,45
USairports	755	23473	62,18	0,09	0,34	6,00	1,86
immuno	1316	6300	9,57	0,53	0,49	1,00	30,30

Tab. 8.1: Descriptive statistics for the 3 networks considered

Two of these graphs, *immuno* and *yeast* are dense (very high mean local and global clustering coefficiens) and show both heavy-tailed degree distributions (power law fit for degree distribution is greater than 2). The graph of the US airports *USairports* is also quite dense (very high global clustering coefficient, mean local coefficient somewhat high).

Only *immuno* is fully connected (only 1 connected component). Thus, it is the only one of the three for which path lengths can be computed.

As expected, these graphs show heavy-tailed degree distributions (Power law fit for degree distribution is always greater than 2) and are dense (gloabl clustering coefficients are high - the local clustering coefficients for *yeast* and *immuno* are very high). But, despite the high density of these three graphs, the size of the snowball sample does not explode (table 8.2)

Metrics Table 8.4 in annex references all values of $\hat{Deff}_1$ and $\hat{Deff}_2$ for all variables of interest and all graphs considered (forest-fire and Barabási-Albert models plus the three real graph datasets). We consider that the estimation for a variable is efficient when the Deff estimate is strictly below 1, and inefficient when it is

Graph name	Snowball sample size
BA	138
yeast	390
Usairports	284
immuno	438

Tab. 8.2: Size of full snowball sample for initial design Bernoulli of size 50

strictly greater than 2. All values between 1 and 2 could be considered acceptable, as explained in the previous section. We shouldn't attach much importance to magnitude of estimates, as we expect them to be somewhat imprecise.

Results are fairly similar across all populations tested. We find that degree and clustering almost always yield inefficient estimates (for 37 graph populations among the 40), whereas page rank and max path length show the most promises for efficient estimates (inefficient estimates for only 3 out 40 populations). Other variables are more undecided: in the cases of betweenness, closeness and eigenvector centralities, the design effect estimates are almost equally divided between the efficiency categories.

8.3.3 Simulations

In this section, we try to assess the predictive power of the previous formulas in terms of efficiency of the snowball design. We do this using Monte-Carlo simulations for all the graph populations considered.

Monte-Carlo estimation of precision To do so, we use the following algorithm:

1. Select a graph population

2. Draw a snowball sample (with an expected size of the initial sample of $n_0 = 50$) and compute the estimates of means and totals given in section 8.3 for all variables of interest

3. Loop with $N_s = 1000$ different samples, denoted $s_i, i = 1 \dots N_s$.
   For graph populations sampled from a graph model (Barabási-Albert and forest-fire), also replicate M=500 different populations, with $N_s = 1000$ different samples for each population. Samples are denoted $s_{ij}, i = 1 \dots N_s, j = 1 \dots M$.
   The order of the graph populations (number of vertices) is $N = 10000$. For each simulation j from a graph model, the variable of interest changes : we denote it $T(Y)_j$, and still denote $\hat{(}T(Y))$ its Horvitz-Thompson estimator computed from sample data for sake of clarity

Then we can obtain Monte-Carlo estimates for the Mean Squared Error of $\hat{(}T(Y))$. For the real graph datasets, only the N=1000 simulated samples need be considered.

$$MSE(T(\hat{Y}))_{MC} = \frac{1}{N_s}\sum_{i=1}^{N_s}(\hat{T}(Y)_i - T(Y))^2$$

For the Barabási-Albert and forest-fire models, we also need to average across the simulated populations:

$$MSE(T(\hat{Y}))_{MC} = \frac{1}{N_s M}\sum_{j=1}^{M}\sum_{i=1}^{N_s}(\hat{T}(Y)_i - T(Y)_j)^2$$

In the same fashion, we define a Monte-Carlo estimate of the bias of the estimate using sample data ($Bias(T(\hat{Y}))_{MC}$). These two quantities yield Monte-Carlo estimates of the variance $\mathbb{V}(T(\hat{Y}))_{MC} = MSE(T(\hat{Y}))_{MC} - Bias(T(\hat{Y}))^2_{MC}$.

Then, we use the simulations to obtain expected sample sizes for the total sample with extension, respectively:

$$n_{MC} = \frac{1}{N_s}\sum_{i=1}^{N_s}\#(s_i)$$

and:

$$n_{MC} = \frac{1}{N_s M}\sum_{j=1}^{M}\sum_{i=1}^{N_s}\#(s_{ij})$$

Finally we have a way to compute the design effect using simulated data:

$$Deff_{MC} = \frac{\mathbb{V}(T(\hat{Y}))_{MC}}{(1/n_{MC} - 1/N)S^2}$$

with S^2 the empirical dispersion for variable Y in the population graph: $S^2 = \frac{1}{N-1}\sum_{k=1}^{N}(y_k - \bar{y})^2$ for real-life datasets and $S^2 = \frac{1}{M(N-1)}\sum_{j=1}^{M}\sum_{k=1}^{N}(y_k - \bar{y}_j)^2$.

Results The results are presented in figure 8.5 for forest-fire graph superpopulations with varying forward burning probabilities, and in figure 8.6 for the Barabasi-Albert model and the three datasets from the *igraphdata* package.

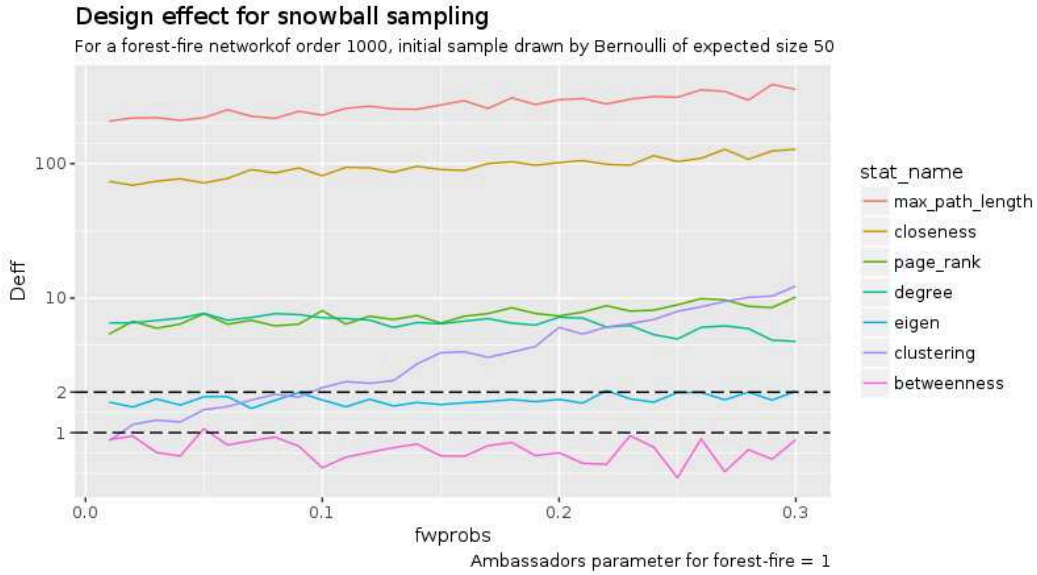


Fig. 8.5: Design effect for some variables of interest for forest-fire model

Again, results are fairly consistent among all graph populations and models, which makes it possible to establish general guidelines. Unfortunately, results rarely coincide with the estimates considered in section 8.3. More variables of interest yield inefficient snowball extension than expected.

Page rank and maximum path length, which were expected to be well estimated using formulas from section 8.3, are in fact almost always inefficient for snowball estimates. The order of magnitude of the design effect are even impressively high, often way above 10 (with the exception of the USairports dataset). Conversely, degree and clustering coefficient are as inefficient as expected. Closeness is inefficient while the measures of section 8.3 were undecided. Eigenvector centrality is acceptable for forest-fire models but a little less so for other models (and was deemed undecided in the predictions). Only betweenness centrality is consistently efficient for almost every graph population considered (the only exception being the *immuno* dataset, with a Deff value of 6.2).

However, the poor performances of the predictive metric makes it hard to understand why this is the case and if this would hold for a graph population with a very different topology. We wish we were able to give a simple criterion based on correlation with

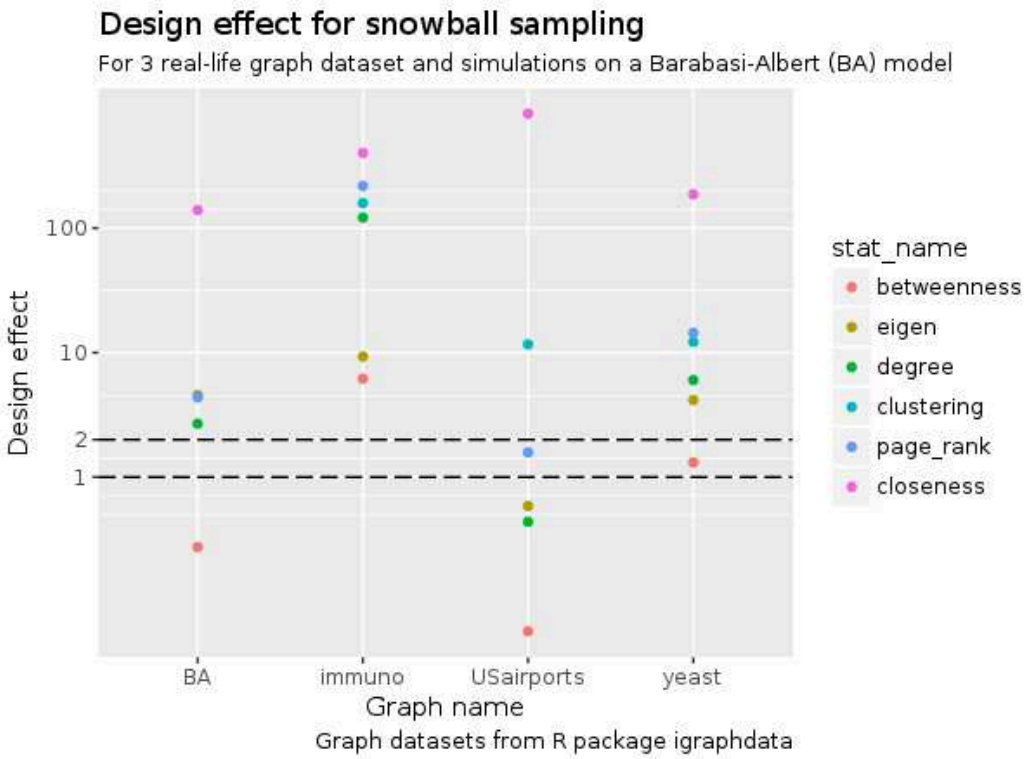


Fig. 8.6: Design effect for some variables of interest for Barabási-Albert model and three real graph models (logarithmic scale)

degree or dispersion among units who are linked to one another, but both conditions unfortunately seem to be neither necessary nor sufficient to ensure decent values of design effects.

These results are summed up in table 8.3. In this table, variables are organized in the three efficiency categories relative to the design effect predictions of section 8.3 (in black), and design effects simulated by Monte-Carlo in section 8.3.3 (in orange).

Efficient	Medium / Undecided	Inefficient
Page rank	Betweenness	Degree
Max path length	Closeness	Clustering
Betweenness	Eigen	Degree
	Eigen	Clustering
		Closeness
		Page rank
		Max path length

Tab. 8.3: Graph variables of interest classified relatively to the efficiency (design effect < 1: efficient, design effect > 2: inefficient, else: medium/undecided)
In black: using $\hat{Deff}_1$ and $\hat{Deff}_2$ estimates. In orange: using Monte-Carlo estimates

It is interesting to note that (see paragraph 8.2), that in terms of design effects, the order is almost reversed compared to the efficiency of sampling estimates. Stats with high MSEs (difficult to estimate with sampling) tend to have low design effects and vice versa.

Finally, efficiency of snowball sampling appears not only hardly predictable using an analogy to cluster sampling, but also very often inefficient. The exact behaviour may differ widely relative to the growth of graph superpopulation, and most random graph models are not very well understood yet, and only tractable using simulations.

8.4 Proportional-to-size sampling

Proportional-to-size sampling (pps sampling) consists in modifying selection probabilities to reduce variance of the estimates. We chose inclusion probabilities propotional to the variable of interest: $\pi_k \propto y_k, k = 1, \dots N$. It is easy to see that for a fixed size design, pps sampling on Y yields a null variance for the horvitz-Thompson estimates of T(Y) and $\bar{Y}$. In practice, values of y_k are unknown for $k = 1, \dots, N$, so we select another variable X (called the auxiliary variable) for which the $X_k, k = 1, \dots, N$ are all known. When X is well correlated with Y, precision is much improved compared to equal selection probabilities.

In the case of snowball sampling, this only changes the selection of the initial sample. In section 8.3.3, the initial sample was selected using equal probabilities (Bernoulli design). The second stage of the snowball sampling design (the extension phase) always consists in selecting all units that are linked to the units selected in the initial sample.

The goal of this paragraph is to examine if pps sampling can improve the design effect of snowball extension, compared to one-stage pps sampling. We don't consider the case of absolute efficiency of pps sampling.

We run simulations similar to those of section 8.3.3 for the same variables of interest. Each variable of interest also acts as a possible auxiliary variable to compute the inclusion probabilities.The graph populations considered are the real datasets *immuno* and *yeast* already described in section 8.3.3, a Barabási-Albert model and two forest-fire models with an ambassadors parameter of 1 and forward burning probabilities of 0.03 and 0.25.

For each auxiliary variable, we plot in figure 8.7 the ratio between the design effect for the estimate that uses only the initial sample, selected using pps sampling (one-stage pps) and the one that uses the sample data with extension. If the resulting value is greater than 1 (above black dashed line in figure 8.7), then it is more efficient in terms of precision to just increase the size of the initial sample and not run the extension phase.

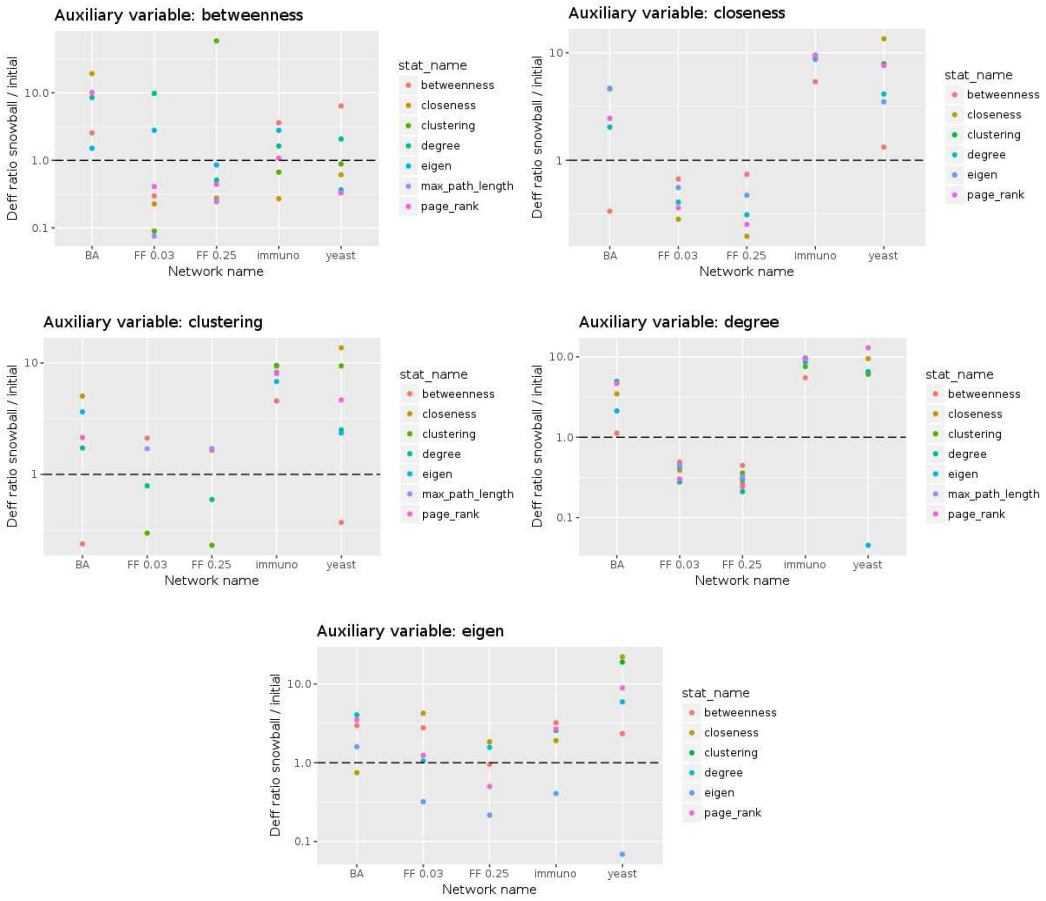


Fig. 8.7: Relative design effect between snowball pps sampling and pps sampling with initial sample for 6 different (graph) auxiliary variables. A point below the dashed line shows that snowball pps sampling is more effective cost-wise than one-stage pps sampling

We also observe that results are rather inconsistent between forest-fire and real-life networks. For example, using degree or closeness as auxiliary variable always yields more efficient snowball samples than one-stage samples, but the result is the opposite for all other networks considered.

Finally, pps sampling is not a spectacular improvement in terms of efficiency of extension sampling designs. Like in section 8.3.3, only betweenness centrality seems to be a consistent improvement over the equal probabilities.

8.5 Conclusion

With the exception of betweenness centrality, snowball extension often comes at a loss of efficiency in comparison of equal probability sampling with no extension phase. This is consistent with what we observed on Twitter data (see chapter 7). Introducing auxiliary information does not make the extension phases more efficient in general.

We have to keep in mind some limitation of the study. First, some results obtained with simulated graphs from a forest-fire model are not always consistent with real-life networks. Second, due to the form of our sampling design, we were not able to conduct snowball sampling for networks too highly clustered. We expect many real-life graphs to be much more clustered globally and locally. In the case of forest-fire networks, we limited the forward burning probabilities to 0.35. Further work should concentrate on methods using a second sampling phase to limit the sampling fractions even in the case of very highly clustered graph. Forest-fire sampling ([LF06]) was designed precisely for this reason. Finally, we should also study the efficiency of extension designs for distributions.

Finally, this study only looked at the statistical efficiency of snowball sampling. Extension designs have a lot of advantages besides their purely statistical properties. Snowball sampling can be algorithmically much more efficient than one-stage samples. It is also the only way to recruit vertices if the subject of the analysis is the units linked to some vertices of interest. Finally, it is a very efficient tool to spot communities ([Les+05]).

Annex - Table of design effect estimations

Graph name	Variable	Metric	Value
Forest-fire 0.05	eigen	deff1	-3.12
Forest-fire 0.05	page_rank	deff1	-0.67
Forest-fire 0.05	max_path_length	deff1	0.07
Forest-fire 0.05	closeness	deff2	1.00
Forest-fire 0.05	page_rank	deff2	1.00
Forest-fire 0.05	degree	deff2	1.00
Forest-fire 0.05	clustering	deff2	1.00
Forest-fire 0.05	max_path_length	deff2	1.00
Forest-fire 0.05	eigen	deff2	1.00
Forest-fire 0.05	closeness	deff1	2.10
Forest-fire 0.05	betweenness	deff2	2.43
Forest-fire 0.05	clustering	deff1	3.46
Forest-fire 0.05	betweenness	deff1	4.93
Forest-fire 0.05	degree	deff1	9.90
Forest-fire 0.1	eigen	deff1	-6.84
Forest-fire 0.1	page_rank	deff1	-1.81
Forest-fire 0.1	max_path_length	deff1	0.32
Forest-fire 0.1	closeness	deff2	1.00
Forest-fire 0.1	page_rank	deff2	1.00
Forest-fire 0.1	degree	deff2	1.00
Forest-fire 0.1	clustering	deff2	1.00
Forest-fire 0.1	eigen	deff2	1.00
Forest-fire 0.1	max_path_length	deff2	1.00
Forest-fire 0.1	closeness	deff1	1.86
Forest-fire 0.1	clustering	deff1	2.82
Forest-fire 0.1	degree	deff1	6.52
Forest-fire 0.1	betweenness	deff2	14.37
Forest-fire 0.1	betweenness	deff1	31.75
Forest-fire 0.15	eigen	deff1	-1098.28
Forest-fire 0.15	page_rank	deff1	-55.35
Forest-fire 0.15	max_path_length	deff1	0.43
Forest-fire 0.15	closeness	deff2	1.00
Forest-fire 0.15	page_rank	deff2	1.00
Forest-fire 0.15	degree	deff2	1.00
Forest-fire 0.15	clustering	deff2	1.00
Forest-fire 0.15	max_path_length	deff2	1.00
Forest-fire 0.15	eigen	deff2	1.00
Forest-fire 0.15	closeness	deff1	1.71

Forest-fire 0.15	clustering	deff1	2.46
Forest-fire 0.15	betweenness	deff2	2.97
Forest-fire 0.15	degree	deff1	4.50
Forest-fire 0.15	betweenness	deff1	6.85
Forest-fire 0.2	page_rank	deff1	-10.72
Forest-fire 0.2	max_path_length	deff1	0.56
Forest-fire 0.2	closeness	deff2	1.00
Forest-fire 0.2	page_rank	deff2	1.00
Forest-fire 0.2	degree	deff2	1.00
Forest-fire 0.2	clustering	deff2	1.00
Forest-fire 0.2	max_path_length	deff2	1.00
Forest-fire 0.2	eigen	deff2	1.00
Forest-fire 0.2	closeness	deff1	1.61
Forest-fire 0.2	clustering	deff1	2.25
Forest-fire 0.2	degree	deff1	3.93
Forest-fire 0.2	betweenness	deff2	4.09
Forest-fire 0.2	eigen	deff1	4.36
Forest-fire 0.2	betweenness	deff1	10.60
Forest-fire 0.25	betweenness	deff1	-1.37
Forest-fire 0.25	betweenness	deff2	0.13
Forest-fire 0.25	max_path_length	deff1	0.69
Forest-fire 0.25	eigen	deff2	1.00
Forest-fire 0.25	max_path_length	deff2	1.00
Forest-fire 0.25	clustering	deff2	1.00
Forest-fire 0.25	degree	deff2	1.00
Forest-fire 0.25	page_rank	deff2	1.00
Forest-fire 0.25	closeness	deff2	1.00
Forest-fire 0.25	closeness	deff1	1.46
Forest-fire 0.25	clustering	deff1	2.26
Forest-fire 0.25	eigen	deff1	3.22
Forest-fire 0.25	degree	deff1	3.50
Forest-fire 0.25	page_rank	deff1	4.82
Forest-fire 0.3	max_path_length	deff1	0.79
Forest-fire 0.3	closeness	deff2	1.00
Forest-fire 0.3	page_rank	deff2	1.00
Forest-fire 0.3	degree	deff2	1.00
Forest-fire 0.3	clustering	deff2	1.00
Forest-fire 0.3	max_path_length	deff2	1.00
Forest-fire 0.3	eigen	deff2	1.00
Forest-fire 0.3	closeness	deff1	1.35
Forest-fire 0.3	betweenness	deff2	1.89
Forest-fire 0.3	eigen	deff1	2.57

Forest-fire 0.3	page_rank	deff1	2.65
Forest-fire 0.3	degree	deff1	2.91
Forest-fire 0.3	clustering	deff1	4.50
Forest-fire 0.3	betweenness	deff1	5.64
BA	max_path_length	deff1	-19.03
BA	closeness	deff1	-11.25
BA	degree	deff1	-1.13
BA	betweenness	deff1	-0.81
BA	betweenness	deff2	-0.40
BA	eigen	deff1	-0.40
BA	page_rank	deff1	-0.29
BA	eigen	deff2	1.00
BA	max_path_length	deff2	1.00
BA	degree	deff2	1.00
BA	page_rank	deff2	1.00
BA	closeness	deff2	1.00
BA	clustering	deff2	1.00
immuno	closeness	deff1	-1.10
immuno	betweenness	deff1	-0.94
immuno	clustering	deff1	0.46
immuno	betweenness	deff2	0.70
immuno	max_path_length	deff2	1.00
immuno	eigen	deff2	1.00
immuno	degree	deff2	1.00
immuno	clustering	deff2	1.00
immuno	page_rank	deff2	1.00
immuno	closeness	deff2	1.00
immuno	page_rank	deff1	1.33
immuno	eigen	deff1	1.46
immuno	max_path_length	deff1	3.15
immuno	degree	deff1	6.11
USairports	page_rank	deff1	-19.11
USairports	clustering	deff1	-0.47
USairports	betweenness	deff1	0.27
USairports	betweenness	deff2	0.97
USairports	eigen	deff2	1.00
USairports	degree	deff2	1.00
USairports	clustering	deff2	1.00
USairports	page_rank	deff2	1.00
USairports	closeness	deff2	1.00
USairports	closeness	deff1	1.08
USairports	degree	deff1	6.00

USairports	eigen	deff1	13.51
yeast	page_rank	deff1	0.46
yeast	betweenness	deff1	0.74
yeast	betweenness	deff2	0.86
yeast	eigen	deff2	1.00
yeast	clustering	deff2	1.00
yeast	degree	deff2	1.00
yeast	page_rank	deff2	1.00
yeast	closeness	deff2	1.00
yeast	closeness	deff1	1.21
yeast	clustering	deff1	1.43
yeast	eigen	deff1	2.12
yeast	degree	deff1	2.14

Tab. 8.4: Table of estimated design effects

Bibliography

[AG92] Miguel A Arcones and Evarist Giné. „On the bootstrap of M-estimators and other statistical functionals“. In: *Exploring the Limits of Bootstrap, ed. by R. LePage and L. Billard, Wiley* (1992), pp. 13–47 (cit. on p. 101).

[Ale89] Kenneth S Alexander. „A counterexample to a correlation inequality in finite sampling“. In: *The Annals of Statistics* 17.1 (1989), pp. 436–439 (cit. on p. 77).

[Apa+15] Sofía Aparicio, Javier Villazón-Terrazas, and Gonzalo Álvarez. „A model for scale-free networks: application to twitter“. In: *Entropy* 17.8 (2015), pp. 5848–5867 (cit. on p. 117).

[Ard06] Pascal Ardilly. *Les techniques de sondage*. Editions Technip, 2006 (cit. on pp. 20, 24, 168).

[BA99] Albert-László Barabási and Réka Albert. „Emergence of scaling in random networks“. In: *science* 286.5439 (1999), pp. 509–512 (cit. on pp. 132, 136, 138).

[Bac+12] Lars Backstrom, Paolo Boldi, Marco Rosa, Johan Ugander, and Sebastiano Vigna. „Four degrees of separation“. In: *Proceedings of the 4th Annual ACM Web Science Conference*. ACM. 2012, pp. 33–42 (cit. on p. 133).

[Bai+07] Sophie Baillargeon, Louis-Paul Rivest, and Michel Ferland. „Stratification en enquêtes entreprises: une revue et quelques avancées“. In: *Proceedings of the Survey Methods Section, Statistical Society of Canada*. 2007 (cit. on p. 21).

[Ban88] Michael D Bankier. „Power allocations: determining sample sizes for subnational areas“. In: *The American Statistician* 42.3 (1988), pp. 174–177 (cit. on pp. 3, 21, 22, 30, 32).

[Bas+09] Mathieu Bastian, Sebastien Heymann, Mathieu Jacomy, et al. „Gephi: an open source software for exploring and manipulating networks.“ In: *Icwsm* 8 (2009), pp. 361–362 (cit. on p. 125).

[BB08] Cynthia Bocci and Jean-François Beaumont. „Another look at ridge calibration“. In: *Metron* 66.1 (2008), pp. 5–20 (cit. on pp. 61, 62).

[BB09] Julius Borcea and Petter Brändén. „The Lee-Yang and Pólya-Schur programs. I. linear operators preserving stability“. In: *Inventiones mathematicae* 177.3 (2009), pp. 541–569 (cit. on pp. 71, 72).

[BB95] Philippe Barbe and Patrice Bertail. „The Weighted Bootstrap, vol. 98 of“. In: *Lecture Notes in Statistics* (1995) (cit. on pp. 71, 73, 85, 100, 101).

[BC10] Patrice Bertail and Stéphan Clémençon. „Sharp bounds for the tails of functionals of Markov chains“. In: *Theory of Probability & Its Applications* 54.3 (2010), pp. 505–515 (cit. on p. 72).

[BC94] Patrice Bertail and Pierre Combris. „Bootstrap généralisé d'un sondage, une application aux données Secodip“. In: *Annales d'Economie et de Statistiques* (1994) (cit. on p. 71).

[Ber+14] Patrice Bertail, Emmanuelle Gautherat, and Hugo Harari-Kermadec. „Empirical φ^*-Divergence Minimizers for Hadamard Differentiable Functionals“. In: *Topics in Nonparametric Statistics*. Springer, 2014, pp. 21–32 (cit. on pp. 6, 99).

[Ber+16] Patrice Bertail, Emilie Chautru, and Stéphan Clémençon. „Empirical processes in survey sampling with (conditional) poisson designs“. In: *Scandinavian Journal of Statistics* (2016) (cit. on pp. 10, 70–72, 78, 92, 94, 96–99, 102).

[Ber98] Yves G Berger. „Rate of convergence to normal distribution for the Horvitz-Thompson estimator“. In: *Journal of Statistical Planning and Inference* 67.2 (1998), pp. 209–226 (cit. on pp. 70, 77).

[BJ12] Petter Brändén and Johan Jonasson. „Negative dependence in sampling“. In: *Scandinavian Journal of Statistics* 39.4 (2012), pp. 830–838 (cit. on pp. 10, 71, 72, 74, 76, 78, 79, 84).

[BL+07] James Bennett, Stan Lanning, et al. „The Netflix prize“. In: *Proceedings of KDD cup and workshop*. Vol. 2007. New York, NY, USA. 2007, p. 35 (cit. on p. 111).

[Boi+17] Hélène Boistard, Hendrik P Lopuhaä, and Anne Ruiz-Gazen. „Functional central limit theorems for single-stage sampling designs“. In: *The Annals of Statistics* 45.4 (2017), pp. 1728–1758 (cit. on pp. 71, 79, 91, 92).

[Bol+11] Johan Bollen, Huina Mao, and Xiaojun Zeng. „Twitter mood predicts the stock market“. In: *Journal of Computational Science* 2.1 (2011), pp. 1–8 (cit. on p. 136).

[Bon72] Phillip Bonacich. „Factoring and weighting approaches to status scores and clique identification“. In: *Journal of mathematical sociology* 2.1 (1972), pp. 113–120 (cit. on p. 120).

[Bor+03] Jonathan Borwein, Rustum Choksi, and Pierre Maréchal. „Probability distributions of assets inferred from option prices via the principle of maximum entropy“. In: *SIAM Journal on Optimization* 14.2 (2003), pp. 464–478 (cit. on p. 6).

[Bor+09] Julius Borcea, Petter Brändén, and Thomas Liggett. „Negative dependence and the geometry of polynomials“. In: *Journal of the American Mathematical Society* 22.2 (2009), pp. 521–567 (cit. on p. 76).

[Bra01] Ulrik Brandes. „A faster algorithm for betweenness centrality“. In: *Journal of mathematical sociology* 25.2 (2001), pp. 163–177 (cit. on p. 118).

[Bri11] Philippe Brion. „Esane, Le dispositif rénové de production des statistiques structurelles d'entreprises“. In: *Courrier des statistiques N 130, Insee* (2011) (cit. on p. 43).

[Bur+15] Pete Burnap, Rachel Gibson, Luke Sloan, Rosalynd Southern, and Matthew Williams. „140 Characters to Victory? Using Twitter to Predict the UK 2015 General Election“. In: *arXiv preprint arXiv:1505.01511* (2015) (cit. on p. 136).

[BW07] Norman E. Breslow and Jon A Wellner. „Weighted likelihood for semiparametric models and two-phase stratified samples, with application to Cox regression“. In: *Scandinavian Journal of Statistics* 35 (2007), pp. 186–192 (cit. on p. 70).

[BW13] Paul Burkhardt and Chris Waring. „An NSA big graph experiment“. In: *presentation at the Carnegie Mellon University SDI/ISTC Seminar, Pittsburgh, Pa*. 2013 (cit. on p. 111).

[BW14] Stefan Milton Bache and Hadley Wickham. *magrittr: A Forward-Pipe Operator for R*. R package version 1.5. 2014 (cit. on p. 9).

[Car+13a] Hervé Cardot, Alain Dessertaine, Camelia Goga, Étienne Josserand, and Pauline Lardin. „Comparison of different sample designs and construction of confidence bands to estimate the mean of functional data: An illustration on electricity consumption“. In: *Survey Methodology* 39 (2013), pp. 283–301 (cit. on p. 70).

[Car+13b] Hervé Cardot, David Degras, and Étienne Josserand. „Confidence bands for Horvitz-Thompson estimators using sampled noisy functional data“. In: *Bernoulli* 19.5A (2013), pp. 2067–2097 (cit. on p. 70).

[CF09] Marc Christine and Sebastien Faivre. „Le projet OCTOPUSSE de nouvel Echantillon-Maître de l'INSEE.“ In: *JMS* 2009 (2009), p. 24 (cit. on p. 21).

[Cha+11] Guillaume Chauvet, Daniel Bonnéry, and J-C Deville. „Optimal inclusion probabilities for balanced sampling“. In: *Journal of Statistical Planning and Inference* 141.2 (2011), pp. 984–994 (cit. on p. 70).

[Cha11] Guillaume Chauvet. „On variance estimation for the French master sample“. In: *Journal of Official Statistics* 27.4 (2011), p. 651 (cit. on p. 76).

[Cha67] Samprit Chatterjee. „A note on optimum allocation“. In: *Scandinavian Actuarial Journal* 1967.1-2 (1967), pp. 40–44 (cit. on pp. 3, 20).

[Che+94] Xiang-Hui Chen, Arthur P Dempster, and Jun S Liu. „Weighted finite population sampling to maximize entropy“. In: *Biometrika* 81.3 (1994), pp. 457–469 (cit. on pp. 77, 78).

[Chi+10a] Paola Maddalena Chiodini, Bianca Maria Martelli, Giancarlo Manzi, and Flavio Verrecchia. „Between theoretical and applied approach: which compromise for unit allocation in business surveys?“ In: *SIS Conference*. Società italiana di statistica. 2010 (cit. on pp. 3, 20).

[Chi+10b] Paola Maddalena Chiodini, Bianca Maria Martelli, Giancarlo Manzi, and Flavio Verrecchia. „The ISAE manufacturing survey sample: validating the Nace Rev. 2 sectorial allocation“. In: *Economic tendency surveys and the services sector*. CIRET. 2010 (cit. on pp. 20, 23).

[Chi+17] Paola Maddalena Chiodini, Giancarlo Manzi, Bianca Maria Martelli, and Flavio Verrecchia. *Sampling Allocation Strategies: A Simulation-Based Comparison*. 2017 (cit. on pp. 3, 20).

[CN06] Gabor Csardi and Tamas Nepusz. „The igraph software package for complex network research“. In: *InterJournal* Complex Systems (2006), p. 1695 (cit. on p. 134).

[Coc63] William Cochran. *Sampling techniques*. 1963 (cit. on pp. 3, 20).

[Col58] James S Coleman. „Relational analysis: the study of social organizations with survey methods“. In: *Human organization* 17.4 (1958), pp. 28–36 (cit. on p. 11).

[Con+11] Michael Conover, Jacob Ratkiewicz, Matthew Francisco, et al. „Political polarization on twitter.“ In: *ICWSM*. 2011 (cit. on p. 136).

[CQ07] Weiwei Cui and Huamin Qu. „A survey on graph visualization“. In: *PhD Qualifying Exam (PQE) Report, Computer Science Department, Hong Kong University of Science and Technology, Kowloon, Hong Kong* (2007) (cit. on p. 112).

[Csa15] Gabor Csardi. „igraphdata: A collection of network data sets for the ‘igraph’package“. In: *R package version 1.0* 1 (2015) (cit. on p. 172).

[CVS12] Marie Cordier-Villoing and Olivier Sautory. „Redressement de la non-réponse et calage dans les enquêtes couplées“. In: *Actes des Journées de Méthodologie Statistique, Paris* (2012) (cit. on p. 66).

[Dal+06] Luca Dall’Asta, Ignacio Alvarez-Hamelin, Alain Barrat, Alexei Vázquez, and Alessandro Vespignani. „Exploring networks with traceroute-like probes: Theory and simulations“. In: *Theoretical Computer Science* 355.1 (2006), pp. 6–24 (cit. on p. 128).

[Dan+55] George B Dantzig, Alex Orden, Philip Wolfe, et al. „The generalized simplex method for minimizing a linear form under linear inequality restraints“. In: *Pacific Journal of Mathematics* 5.2 (1955), pp. 183–195 (cit. on pp. 7, 42).

[DD09] Laurent Davezies and Xavier D’Haultfœuille. „Faut-il pondérer?... Ou l’éternelle question de l’économètre confronté à un problème de sondage“. In: *Document de travail de l’Insee* (2009) (cit. on p. 21).

[Dec+11] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. „Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications“. In: *Physical Review E* 84.6 (2011), p. 066106 (cit. on p. 134).

[Del+02] Fabien Dell, Xavier d’Haultfoeuille, Philippe Février, and Emmanuel Massé. „Mise en oeuvre du calcul de variance par linéarisation“. In: *Actes des Journées de Méthodologie Statistique, http://jms. insee. fr/site/index. php* (2002) (cit. on p. 161).

[Dev12] Jean-Claude Deville. „Échantillonnage de réseaux, une relecture de S.K Thompson avec une nouvelle présentation et quelques nouveautés“. In: *Actes des Journées Méthodologie Statistique* (2012). [Online; accessed 7-April-2017] (cit. on pp. 148, 152).

[DHJ59] Tore Dalenius and Joseph L Hodges Jr. „Minimum variance stratification“. In: *Journal of the American Statistical Association* 54.285 (1959), pp. 88–101 (cit. on p. 21).

[DS92] Jean-Claude Deville and Carl-Erik Särndal. „Calibration estimators in survey sampling“. In: *Journal of the American statistical Association* 87.418 (1992), pp. 376–382 (cit. on pp. 5, 6, 38, 57, 58, 154, 158).

[DT04] Jean-Claude Deville and Yves Tillé. „Efficient balanced sampling: the cube method“. In: *Biometrika* 91.4 (2004), pp. 893–912 (cit. on p. 76).

[DT98] Jean-Claude Deville and Yves Tillé. „Unequal probability sampling without replacement through a splitting method“. In: *Biometrika* 85.1 (1998), pp. 89–101 (cit. on pp. 71, 79).

[Dud90] Richard M Dudley. „Nonlinear functionals of empirical measures and the bootstrap“. In: *Probability in Banach Spaces 7*. Springer, 1990, pp. 63–82 (cit. on p. 100).

[ER59] Paul Erdős and Alfréd Rényi. „On random graphs“. In: *Publ. Math. Debrecen* 6.290 (1959), pp. 290–297 (cit. on pp. 12, 132).

[ER60] Paul Erdős and Alfréd Rényi. „On the evolution of random graphs“. In: *Publ. Math. Inst. Hung. Acad. Sci.* 5 (1960), pp. 17–61 (cit. on p. 132).

[ER61] Paul Erdős and Alfréd Rényi. „On the strength of connectedness of a random graph“. In: *Acta Mathematica Hungarica* 12.1 (1961), pp. 261–267 (cit. on p. 132).

[Fer+14] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. „The rise of social bots“. In: *arXiv preprint arXiv:1407.5225* (2014) (cit. on p. 137).

[Fer15] Emilio Ferrara. „Manipulation and Abuse on Social Media“. In: *SIGWEB Newsl.* Spring (Apr. 2015), 4:1–4:9 (cit. on p. 159).

[Fra77] Ove Frank. „Survey sampling in graphs“. In: *Journal of Statistical Planning and Inference* 1.3 (1977), pp. 235–264 (cit. on pp. 15, 123, 149, 150, 163, 167, 168).

[Fre78] Linton C Freeman. „Centrality in social networks conceptual clarification“. In: *Social networks* 1.3 (1978), pp. 215–239 (cit. on p. 118).

[Gfe07] David Gfeller. „Simplifying complex networks: from a clustering to a coarse graining strategy“. In: *These École polytechnique Fédérale de Lausanne EPFL* 3888 (2007) (cit. on p. 173).

[Gil+88] Richard D Gill, Yehuda Vardi, and Jon A Wellner. „Large sample theory of empirical distributions in biased sampling models“. In: *The Annals of Statistics* 16.3 (1988), pp. 1069–1112 (cit. on p. 70).

[Gil59] Edgar N Gilbert. „Random graphs“. In: *The Annals of Mathematical Statistics* (1959), pp. 1141–1144 (cit. on p. 132).

[Goo61] Leo A Goodman. „Snowball sampling“. In: *The annals of mathematical statistics* (1961), pp. 148–170 (cit. on p. 11).

[GR16] Emmanuel Gros and Antoine Rebecq. „Étude du calage sur bornes minmales“. In: *Actes du 9ème colloque francophone sondages* (2016) (cit. on pp. 61, 65).

[Gra14] Monique Graf. „Calage serré des poids d'enquête“. In: *Actes 8eme colloque francophone Sondages* (2014) (cit. on pp. 6, 7, 39).

[Gro15] Emmanuel Gros. „Note de bilan relative aux calculs de pondérations dans l'enquête nationale sur les ressources des jeunes“. In: *Document de travail INSEE* (2015) (cit. on p. 62).

[Gro16] Emmanuel Gros. „Les différentes méthodes de calage – Le choix des « bornes » de calage“. In: *Présentation au Séminaire de Méthodologie Statistique du département des méthodes statistiques de l'Insee* (2016) (cit. on pp. 45, 48).

[GT10] Fabien Guggemos and Yves Tillé. „Penalized calibration in survey sampling: Design-based estimation assisted by mixed models“. In: *Journal of statistical planning and inference* 140.11 (2010), pp. 3199–3212 (cit. on p. 61).

[GUH16] Carlos A Gomez-Uribe and Neil Hunt. „The Netflix recommender system: Algorithms, business value, and innovation“. In: *ACM Transactions on Management Information Systems (TMIS)* 6.4 (2016), p. 13 (cit. on p. 111).

[Gup+13] Pankaj Gupta, Ashish Goel, Jimmy Lin, et al. „Wtf: The who to follow service at twitter“. In: *Proceedings of the 22nd international conference on World Wide Web*. ACM. 2013, pp. 505–514 (cit. on p. 111).

[Ham74] Frank R Hampel. „The influence curve and its role in robust estimation“. In: *Journal of the American Statistical Association* 69.346 (1974), pp. 383–393 (cit. on p. 100).

[Han+10] John Hannon, Mike Bennett, and Barry Smyth. „Recommending twitter users to follow using content and collaborative filtering approaches“. In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM. 2010, pp. 199–206 (cit. on p. 111).

[Hec+99] Douglas D Heckathorn, Robert S Broadhead, Denise L Anthony, and David L Weakliem. „AIDS and social networks: HIV prevention through network mobilization“. In: *Sociological Focus* 32.2 (1999), pp. 159–179 (cit. on pp. 11, 123).

[Hec97] Douglas D Heckathorn. „Respondent-driven sampling: a new approach to the study of hidden populations“. In: *Social problems* (1997), pp. 174–199 (cit. on pp. 123, 130).

[HH43] Morris H Hansen and William N Hurwitz. „On the theory of sampling from finite populations“. In: *The Annals of Mathematical Statistics* 14.4 (1943), pp. 333–362 (cit. on p. 151).

[HJ01] Douglas D Heckathorn and Joan Jeffri. „Finding the beat: Using respondent-driven sampling to study jazz musicians“. In: *Poetics* 28.4 (2001), pp. 307–329 (cit. on pp. 11, 123, 130).

[HJ91] Jørgen Hoffmann-Jørgensen. *Stochastic processes on Polish spaces*. 39. Aarhus Universitet. Matematisk Institut, 1991 (cit. on p. 93).

[Hor+15] Kurt Hornik, David Meyer, and Christian Buchta. „slam: Sparse Lightweight Arrays and Matrices“. In: *R package version 0.1-32, URL http://CRAN. R-project. org/package= slam* (2015) (cit. on p. 42).

[Hou+06] J. Ben Hough, Manjunath Krishnapur, Yuval Peres, and Bálint Virág. „Determinantal processes and independence“. In: *Probability surveys* 3.206-229 (2006), p. 9 (cit. on p. 79).

[HR09] Peter J Huber and Elvezio M Ronchetti. „Robust statistics“. In: *NJ: Wiley. doi* 10.1002 (2009), p. 9780470434697 (cit. on p. 101).

[HT52] Daniel G Horvitz and Donovan J Thompson. „A generalization of sampling without replacement from a finite universe“. In: *Journal of the American Statistical Association* 47.260 (1952), pp. 663–685 (cit. on pp. 75, 148, 156, 158).

[Hua+04] Zan Huang, Wingyan Chung, and Hsinchun Chen. „A graph model for E-commerce recommender systems“. In: *Journal of the Association for Information Science and Technology* 55.3 (2004), pp. 259–274 (cit. on p. 111).

[Háj59] Jaroslav Hájek. „Optimal strategy and other problems in probability sampling“. In: *Časopis pro pěstování matematiky* 84.4 (1959), pp. 387–423 (cit. on p. 77).

[Háj64] Jaroslav Hájek. „Asymptotic theory of rejective sampling with varying probabilities from a finite population“. In: *The Annals of Mathematical Statistics* (1964), pp. 1491–1523 (cit. on pp. 70, 71, 77, 78, 85, 101).

[Háj81] Jaroslav Hájek. *Sampling from a finite population*. Marcel Dekker, New York, 1981 (cit. on p. 78).

[IF82] Cary T Isaki and Wayne A Fuller. „Survey design under the regression superpopulation model“. In: *Journal of the American Statistical Association* 77.377 (1982), pp. 89–96 (cit. on p. 154).

[Jac+14] Mathieu Jacomy, Tommaso Venturini, Sebastien Heymann, and Mathieu Bastian. „ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software“. In: *PloS one* 9.6 (2014), e98679 (cit. on p. 112).

[JDP83] Kumar Joag-Dev and Frank Proschan. „Negative association of random variables with applications“. In: *The Annals of Statistics* (1983), pp. 286–295 (cit. on pp. 71–73).

[KM09] Malik Koubi and Sandrine Mathern. *Résolution d'une des limites de l'allocation de neyman*. 2009 (cit. on pp. 3, 21, 31).

[KM16] Tomasz Kajdanowicz and Mikołaj Morzy. „Using graph and vertex entropy to compare empirical graphs with theoretical graph models“. In: *Entropy* 18.9 (2016), p. 320 (cit. on p. 172).

[Kol09] Eric D Kolaczyk. *Statistical analysis of network data*. Springer, 2009 (cit. on pp. 117, 118, 126, 146).

[Kor+09] Yehuda Koren, Robert Bell, and Chris Volinsky. „Matrix factorization techniques for recommender systems“. In: *Computer* 42.8 (2009) (cit. on p. 111).

[Kru+12] Philippe Kruchten, Robert L Nord, and Ipek Ozkaya. „Technical debt: From metaphor to theory and practice“. In: *Ieee software* 29.6 (2012), pp. 18–21 (cit. on p. 112).

[Lav12] Pierre Lavallée. „Le sondage déterminé selon les répondants: un survol méthodologique“. In: *Actes des Journées de Méthodologie Statistique* (2012). [Online; accessed 27-April-2018] (cit. on pp. 11, 123, 130).

[LC01] Pierre Lavallée and Pierre Caron. „Estimation par la méthode généralisée du partage des poids: Le cas du couplage d'enregistrements“. In: *Survey Methodology* 27.2 (2001), pp. 171–188 (cit. on p. 137).

[Les+05] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. „Graphs over time: densification laws, shrinking diameters and possible explanations“. In: *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM. 2005, pp. 177–187 (cit. on pp. 12, 134, 164, 180).

[Les09] Éric Lesage. „Calage non linéaire“. In: *Actes Xèmes Journées de Méthodologie Statistique* (2009) (cit. on pp. 66, 161).

[LF06] Jure Leskovec and Christos Faloutsos. „Sampling from large graphs“. In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2006, pp. 631–636 (cit. on pp. 125, 136, 164, 171, 180).

[LM18] Vincent Loonis and Xavier Mary. „Determinantal sampling designs“. In: *Journal of Statistical Planning and Inference* (2018) (cit. on pp. 71, 79).

[Lo91] Albert Y Lo. „Bayesian bootstrap clones and a biometry function“. In: *Sankhyā: The Indian Journal of Statistics, Series A* (1991), pp. 320–333 (cit. on p. 73).

[Lo93] Albert Y Lo. „A Bayesian bootstrap for censored data“. In: *The Annals of Statistics* 21.1 (1993), pp. 100–123 (cit. on p. 73).

[LS14] Jure Leskovec and Rok Sosič. *Snap.py: SNAP for Python, a general purpose network analysis and graph mining tool in Python*. `http://snap.stanford.edu/snappy`. June 2014 (cit. on pp. 136, 163, 164).

[Lum+04] Thomas Lumley et al. „Analysis of complex survey samples“. In: *Journal of Statistical Software* 9.1 (2004), pp. 1–19 (cit. on pp. 9, 58).

[Lyo03] Russell Lyons. „Determinantal probability measures“. In: *Publications Mathématiques de l'Institut des Hautes Études Scientifiques* 98 (2003), pp. 167–212 (cit. on p. 79).

[Mal+10] Grzegorz Malewicz, Matthew H Austern, Aart JC Bik, et al. „Pregel: a system for large-scale graph processing“. In: *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*. ACM. 2010, pp. 135–146 (cit. on p. 112).

[MAR] Thomas Merly-Alpa and Antoine Rebecq. *L'algorithme CURIOS pour l'optimisation du plan de sondage en fonction de la non-réponse*. `http://papersjds15.sfds.asso.fr/submission_29.pdf`. [Online; accessed 7-April-2017] (cit. on pp. 22, 26, 145).

[Mir+15] Sherzod M Mirakhmedov, Sreenivasa R Jammalamadaka, and Magnus Ekström. „Edgeworth expansions for two-stage sampling with applications to stratified and cluster sampling“. In: *Canadian Journal of Statistics* 43.4 (2015), pp. 578–599 (cit. on p. 81).

[Mue+06] Christopher Mueller, Douglas P Gregor, and Andrew Lumsdaine. „Distributed Force-Directed Graph Layout and Visualization.“ In: *EGPGV* 6 (2006), pp. 83–90 (cit. on p. 112).

[Mus+11] Eni Mustafaraj, Samantha Finn, Carolyn Whitlock, and Panagiotis Takis Metaxas. „Vocal minority versus silent majority: Discovering the opionions of the long tail“. In: *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom)*. 2011, pp. 103–110 (cit. on p. 137).

[Mye+14] Seth A Myers, Aneesh Sharma, Pankaj Gupta, and Jimmy Lin. „Information network or social network?: the structure of the twitter follow graph“. In: *Proceedings of the companion publication of the 23rd international conference on World wide web companion*. International World Wide Web Conferences Steering Committee. 2014, pp. 493–498 (cit. on pp. 138, 145, 146).

[New03] Mark EJ Newman. „The structure and function of complex networks“. In: *SIAM review* 45.2 (2003), pp. 167–256 (cit. on p. 121).

[New80] Charles M Newman. „Normal fluctuations and the FKG inequalities“. In: *Communications in Mathematical Physics* 74.2 (1980), pp. 119–128 (cit. on p. 88).

[Ney34] Jerzy Neyman. „On the two different aspects of the representative method: the method of stratified sampling and the method of purposive selection“. In: *Journal of the Royal Statistical Society* (1934), pp. 558–625 (cit. on pp. 2, 20, 144).

[NM65] John A Nelder and Roger Mead. „A simplex method for function minimization“. In: *The computer journal* 7.4 (1965), pp. 308–313 (cit. on p. 27).

[NS01] Krzysztof Nowicki and Tom A B Snijders. „Estimation and prediction for stochastic blockstructures“. In: *Journal of the American Statistical Association* 96.455 (2001), pp. 1077–1087 (cit. on p. 136).

[Oli12] Paulo Eduardo Oliveira. *Asymptotics for associated random variables*. Springer Science & Business Media, 2012 (cit. on pp. 72, 87, 89).

[Pag+99] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. *The PageRank citation ranking: Bringing order to the web.* Tech. rep. Stanford InfoLab, 1999 (cit. on p. 120).

[Pat+01] Ronald F Patterson, Wendy D Smith, Robert L Taylor, and Abolghassem Bozorgnia. „Limit theorems for negatively dependent random variables“. In: *Nonlinear Analysis: Theory, Methods & Applications* 47.2 (2001), pp. 1283–1295 (cit. on pp. 11, 84).

[PDT91] Odile Pons and Elisabeth De Turckheim. „Von Mises method, bootstrap and Hadamard differentiability for nonparametric general models“. In: *Statistics* 22.2 (1991), pp. 205–214 (cit. on pp. 99, 100).

[Pei14] Tiago P. Peixoto. „The graph-tool python library“. In: *figshare* (2014) (cit. on p. 121).

[Pem11] Robin Pemantle. „Hyperbolicity and stable polynomials in combinatorics and probability“. In: *Current Development in Mathematics, Proceedings of the 2011 conference* (2011), pp. 57–124 (cit. on p. 74).

[Pol+99] Dimitris Politis, Joseph P Romano, and Michael Wolf. „Weak convergence of dependent empirical measures with application to subsampling in function spaces“. In: *Journal of statistical planning and inference* 79.2 (1999), pp. 179–190 (cit. on p. 74).

[PU97] Magda Peligrad and Sergey Utev. „Central limit theorem for linear processes“. In: *The Annals of Probability* 25.1 (1997), pp. 443–456 (cit. on pp. 86–88).

[RB09] Douglas Rivers and Delia Bailey. „Inference from matched samples in the 2008 US national elections“. In: *Proceedings of the Joint Statistical Meetings*. 2009, pp. 627–639 (cit. on p. 137).

[Reb16] Antoine Rebecq. „Icarus: an R package for calibration in survey sampling“. R package version 0.2.0. 2016 (cit. on p. 39).

[RFM16] Antoine Rebecq and Cyril Favre-Martinoz. „Équilibrage des poids des ZAE pour le plan de sondage OCTOPUSSE“. In: *Document de travail INSEE* (2016) (cit. on p. 62).

[Rie94] Helmut Rieder. „Von Mises Functionals“. In: *Robust Asymptotic Statistics*. Springer, 1994, pp. 1–37 (cit. on p. 101).

[RMA15] Antoine Rebecq and Thomas Merly-Alpa. „Algorithme CURIOS et méthode de "priorisation" pour les enquêtes en face-à-face. Application à l'enquête Patrimoine 2014.“ In: *Actes des Journées de Méthodologie Statistique* (2015) (cit. on p. 27).

[RV01] Gildas Roy and Aurélie Vanheuverzwyn. „Redressement par la macro CALMAR : applications et pistes d'amélioration“. In: *Traitements des fichiers d'enquête* (2001), pp. 31–46 (cit. on pp. 39, 64).

[Sau03] Olivier Sautory. „Calmar 2: A new version of the Calmar calibration adjustment program“. In: *Proceedings of Statistics Canada Symposium*. 2003 (cit. on pp. 8, 9, 57).

[Sau15] Olivier Sautory. „Les enjeux méthodologiques liés à l'usage de bases de sondage imparfaites“. In: *Actes des Journées de Méthodologie Statistique* (2015) (cit. on p. 141).

[Sau93] Olivier Sautory. „La macro Calmar. Redressement d'un échantillon par calage sur marges“. In: *Document F9310, DSDS, INSEE* (1993) (cit. on pp. 42, 57).

[Sax+17] Akrati Saxena, Ralucca Gera, and SRS Iyengar. „Degree Ranking Using Local Information“. In: *arXiv preprint arXiv:1706.01205* (2017) (cit. on p. 172).

[Shv+10] Konstantin Shvachko, Hairong Kuang, Sanjay Radia, and Robert Chansler. „The hadoop distributed file system“. In: *Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on*. Ieee. 2010, pp. 1–10 (cit. on p. 111).

[Slo+13] Luke Sloan, Jeffrey Morgan, William Housley, et al. „Knowing the tweeters: Deriving sociologically relevant demographics from Twitter“. In: *Sociological Research Online* 18.3 (2013), p. 7 (cit. on p. 137).

[Sol+15] Gary Solon, Steven J Haider, and Jeffrey M Wooldridge. „What are we weighting for?“ In: *Journal of Human resources* 50.2 (2015), pp. 301–316 (cit. on p. 21).

[SW86] Galen R Shorack and Jon A Wellner. „Empirical Processes with Applications to Statistics Wiley Series in Probability and Mathematical Statistics: Probability and Mathematical Statistics“. In: *New York* (1986) (cit. on p. 99).

[Sär+03] Carl-Erik Särndal, Bengt Swensson, and Jan Wretman. *Model assisted survey sampling*. Springer Science & Business Media, 2003 (cit. on pp. 141, 157).

[TH14] Stefan Theussl and Kurt Hornik. „Rglpk: R/GNU linear programming kit interface“. In: *R package version 0.6* (2014) (cit. on p. 42).

[Tho06] Steven K Thompson. „Adaptive web sampling“. In: *Biometrics* 62.4 (2006), pp. 1224–1234 (cit. on pp. 152, 161).

[Tho90] Steven K Thompson. „Adaptive cluster sampling“. In: *Journal of the American Statistical Association* 85.412 (1990), pp. 1050–1059 (cit. on p. 146).

[Tho91] Steven K Thompson. „Stratified adaptive cluster sampling“. In: *Biometrika* (1991), pp. 389–397 (cit. on p. 148).

[Tho98] Steven K Thompson. „Adaptive sampling in graphs“. In: *Proceedings of the Section on Survey Methods Research, American Statistical Association*. 1998, pp. 13–22 (cit. on pp. 115, 148, 152).

[Til01] Yves Tillé. „Théorie des sondages“. In: *Dunod, Paris* (2001) (cit. on p. 155).

[TM16] Yves Tillé and Alina Matei. „Sampling: survey sampling“. In: *R package version* 2.8 (2016) (cit. on pp. 9, 58).

[Tum+10] Andranik Tumasjan, Timm Oliver Sprenger, Philipp G Sandner, and Isabell M Welpe. „Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment.“ In: *ICWSM* 10 (2010), pp. 178–185 (cit. on p. 136).

[Van01] Camille Vanderhoeft. „Generalized calibration at statistic belgium“. In: *Technical Report 3, Statistics Belgium WorkingPaper* (2001) (cit. on pp. 6, 39, 65).

[Vav+13] Vinod Kumar Vavilapalli, Arun C Murthy, Chris Douglas, et al. „Apache hadoop yarn: Yet another resource negotiator“. In: *Proceedings of the 4th annual Symposium on Cloud Computing*. ACM. 2013, p. 5 (cit. on p. 111).

[VH08] Erik Volz and Douglas D Heckathorn. „Probability based estimation theory for respondent driven sampling“. In: *Journal of official statistics* 24.1 (2008), p. 79 (cit. on p. 123).

[VM+02] Christian Von Mering, Roland Krause, Berend Snel, et al. „Comparative assessment of large-scale data sets of protein–protein interactions“. In: *Nature* 417.6887 (2002), p. 399 (cit. on p. 173).

[VT17] Kyle Vincent and Steve Thompson. „Estimating population size with link-tracing sampling“. In: *Journal of the American Statistical Association* 112.519 (2017), pp. 1286–1295 (cit. on p. 130).

[VW96] Aad W Van der Vaart and Jon A Wellner. *Weak Convergence and Empirical Processes*. Springer, New York, 1996 (cit. on pp. 91, 93, 96–99).

[Wan12] Jianqiang C. Wang. „Sample distribution function based goodness-of-fit test for complex surveys“. In: *Computational Statistics & Data Analysis* 56.3 (2012), pp. 664–679 (cit. on p. 70).

[WB89] John K Watters and Patrick Biernacki. „Targeted sampling: options for the study of hidden populations“. In: *Social problems* 36.4 (1989), pp. 416–430 (cit. on p. 11).

[WF94] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications*. Vol. 8. Cambridge university press, 1994 (cit. on p. 134).

[Wic11] Hadley Wickham. „testthat: Get started with testing“. In: *The R Journal* 3.1 (2011), pp. 5–10 (cit. on p. 66).

[WS98] Duncan J Watts and Steven H Strogatz. „Collective dynamics of ‘small-world’networks“. In: *nature* 393.6684 (1998), pp. 440–442 (cit. on pp. 120, 133, 136, 138).

[Xin+13] Reynold S Xin, Joseph E Gonzalez, Michael J Franklin, and Ion Stoica. „Graphx: A resilient distributed graph system on spark“. In: *First International Workshop on Graph Data Management Experiences and Systems*. ACM. 2013, p. 2 (cit. on p. 112).

List of Figures

1.1 Variance de l'estimateur d'Horvitz-Thompson en fonction du paramètre λ pour un jeu de données explicité au chapitre 2. Le choix de λ optimal (dans le coude de la courbe) est indiqué par un flèche. 4

1.2 Distribution des facteurs de repondération pour un problème de calage avec deux choix de méthodes différents 7

1.3 Exemple de travail avec Icarus sous l'*IDE* Rstudio 8

1.4 Illustration de l'échantillonnage boule de neige (snowball sampling) . 13

1.5 Illustration d'un sondage adaptatif. La population est représentée sur un carré de 20 $\times$ 20 et les sous-réseaux sont marqués par les points noirs. Dans la première phase de sondage (a), 10 carreaux sont sélectionnés (marqués en bleu ciel). Dans la phase d'extension (b), on ajoute à l'échantillon la totalité des sous-réseaux qui peuvent être observés à partir de la première phase. Dans cette illustration, la population est également partitionnée en deux strates, marquées par la ligne verticale. 13

2.1 Exemple d'un point de torsion de la fonction $V(\lambda)$ pour un plan de sondage explicité dans Merly-Alpa and Rebecq [MAR] 26

2.2 Variance de l'estimateur d'Horvitz-Thompson du total du chiffre d'affaire dans le cadre d'un compromis avec la répartition de Neyman. 29

3.1 RRMSE (en %) des estimateurs par division. *Note de lecture :chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.* 46

3.2 RRMSE (en %) des estimateurs par division (zoom du graphique en figure 3.1). *Note de lecture :chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.* 47

3.3 RRMSE (en %) des estimateurs par division. *Note de lecture :chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.* 50

3.4 RRMSE (en %) des estimateurs par division (zoom du graphique en figure 3.3). *Note de lecture :chaque point représente le RRMSE de l'estimateur du total d'une variable donnée sur un domaine donné pour un scénario de calage donné. Un point situé au-dessus de la bissectrice indique un RRMSE supérieur au RRMSE obtenu par le raking ratio pour ce domaine.* 51

5.1 Selection of a sample of size $n = 40$ in a population of units (black dots) divided in $M = 12$ primary units of equal size $N_j = 64,\ j = 1, ..., M$ ($N = 768$) with a two-stage design. First, a sample of $m = 4$ first-degree units are selected between the $M = 12$ with simple random sampling (in pink). Then in each of theses 4 selected primary units, 10 units of the population are selected (red dots), again via simple random sampling. 80

5.2 Matrix of the $\Delta_{ij}, i, j \in [\![1, 1000]\!]$ with M = 20, m = 5, $n_0 = 10$. The positive coefficients are shown in yellow, the negative ones in blue. . . 82

5.3 Asymptotic confidence bands at the 95% level with two different values for H, obtained with $\beta = 1$, $\sigma_u^2 = 0.01$, $N = 10000$, $c = 0.05$ and 10000 simulations. We observe that the bands are more uniform with a higher number of strata (i.e. quantiles in the balancing constraints). 103

5.4 Example of a graph with 7 vertices and 6 edges. 105

5.5 Total sample (with extension), for a snowball sample of initial sample size 1 on the graph of figure 5.4, depending on the unit selected at the initial stage 106

5.6 An Erdős-Rényi model of 100 vertices with p=0.01. The graph is unconnected. 106

6.1 Different visualization algorithms for the same graph (Erdős-Rényi with 100 nodes and p=0.01). 113

6.2 Drug use graph population example 116

6.3 Degree distributions. Left: the router-level Internet network graph from CAIDA data. Right: the network of measured interactions among proteins in yeast. In each plot, both x- and y-axes are in base-2 logarithmic scale. Graph excerpted from [Kol09] 117

6.4 Graph example with betweenness centrality. The darker the color, the higher the betweenness centrality. 119

6.5 Graph example with closeness centrality. The darker the color, the higher the centrality. 119

6.6 Graph example with pagerank centrality. The darker the color, the higher the centrality. 120

6.7 Examples of possible values of local clustering coefficient for the vertex in orange, depending on the configuration of the network for its friends. 121

6.8 Illustration of a sampled graph by the induced subgraph sampling method. The population graph is connected but the sampled graph is not. Thus plugin estimation of connectivity fails. 124

6.9 Example of “induced subgraph sampling". 9 nodes are selected (in yellow) among the 24 in the graph population. The edges observed (in red) are those connecting two nodes of the sample. 126

6.10 Example of “incident subgraph sampling". 5 edges are selected (in yellow) among the 29 edges in the population graph. The observed nodes (in red) are those that are linked by at least one edge in the sample.127

6.11 Example of “link tracing sampling". Sampled source nodes (in yellow): B,H,P. Sampled target nodes (in yellow): O,L,Q. Observed nodes (in red) along the paths from sources to targets: C,E,G,N,K,R. Observed edges colored in red. 129

6.12 Illustration of stratified adaptive sampling. The population is described as a 20×20 grid (and two strata partitioning the population in two, marked with the vertical red line). The networks of units are represented by aggregated little dots. In the first sampling phase (a), five squares are selected in each stratum (marked in light blue). In the second phase (b), the sample is extended with the networks (and their sides) that could be reached using the first phase sample. 131

6.13 Illustration of snowball sampling 131

6.14 Example of Erdős-Rényi graphs with different parameters 132

6.15 Example of an undirected Barabási-Albert graph with 30 vertices. Vertices that appeared earlier (indicated by a low vertex index) have a higher attachment probability (and thus generally a higher degree). Mean degree for the graph is 1.93, kurtosis of the degree distribution is 10.4. 133

6.16 Example of a small-world graph generated with the Watts-Strogatz model, with 50 vertices. Mean degree is 10. Global clustering coefficient is 0.56. 134

7.1 Illustration of snowball sampling 146

7.2 Illustration of stratified adaptive sampling. The population is described as a 20×20 grid (and two strata partitioning the population in two, marked with the vertical red line). The networks of units are represented by aggregated little dots. In the first sampling phase (a), five squares are selected in each stratum (marked in light blue). In the second phase (b), the sample is extended with the networks (and their sides) that could be reached using the first phase sample. 147

8.1 Mean squared errors for plugin and weighted estimates (logarithmic scale) 166

8.2 Absolute value of bias for plugin and weighted estimates (logarithmic scale) 166

8.3 Size of snowball sample for forest-fire model in function of forward burning probabilities and number of ambassadors. Initial sampling design: Bernoulli of expected size 50 172
8.4 Value of statistics of interest for the forest-fire model with forward burning probabilities between 0.01 and 0.30. 172
8.5 Design effect for some variables of interest for forest-fire model 176
8.6 Design effect for some variables of interest for Barabási-Albert model and three real graph models (logarithmic scale) 177
8.7 Relative design effect between snowball pps sampling and pps sampling with initial sample for 6 different (graph) auxiliary variables. A point below the dashed line shows that snowball pps sampling is more effective cost-wise than one-stage pps sampling 179

List of Tables

2.1 Distribution des tailles d'échantillon par strate pour les deux répartitions, et tailles d'échantillon pour les strates correspondant aux tailles d'échantillon maximales. 28

2.2 Distribution des tailles d'échantillon par strate pour la répartition obtenue, ainsi que pour les deux strates correspondant aux tailles d'échantillon maximales pour la répartition de Neyman et la répartition proportionnelle. 30

2.3 Dispersion des poids et variance des estimateurs du chiffre d'affaires et de z pour plusieurs répartitions. 31

2.4 Précisions locales de l'estimateur du total de chiffre d'affaires pour plusieurs répartitions. 32

3.1 Fonctions de distance associées aux méthodes de calage classiques . . . 40

5.1 Distribution of width of confidence bands for different values of N and H. Results obtained with 10000 simulations 104

7.1 Estimates and characteristics for three different sampling designs . . . 159

8.1 Descriptive statistics for the 3 networks considered 173

8.2 Size of full snowball sample for initial design Bernoulli of size 50 . . . 174

8.3 Graph variables of interest classified relatively to the efficiency (design effect < 1: efficient, design effect > 2: inefficient, else: medium/undecided) In black: using $\hat{Deff}_1$ and $\hat{Deff}_2$ estimates. In orange: using Monte-Carlo estimates 177

8.4 Table of estimated design effects 184