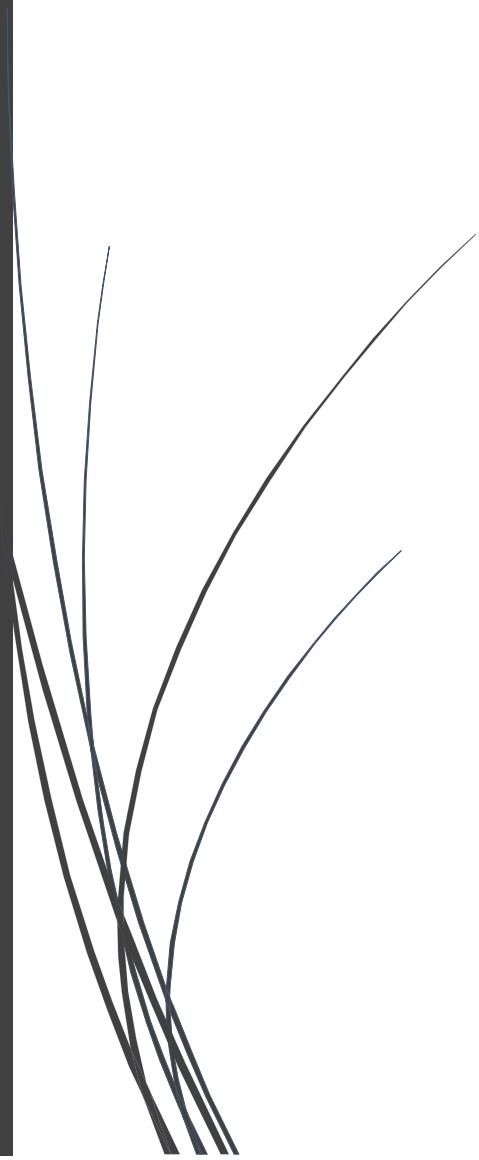
25/04/2018

PROJET STA108

Enquêtes & Sondages

A l'attention de Fabienne LE SAGER, Thierry HORSIN, Aurélie VANHEUVERZWYN et Sylvie ROUSSEAU.



Guillaume CHEVRON
CONSERVATOIRE NATIONAL DES ARTS ET METIERS

le cnam

Table des matières

I. Objectif de l'étude et du projet

A. Origine et nature des données

Fondée sur un jeu de données en libre accès, fourni par le ministère de l'intérieur, cette étude a pour objectif d'analyser la qualité des estimations effectuées au second tour de l'élection présidentielle française de 2007, selon la méthode d'échantillonnage.

Pour améliorer la clarté des estimations, les fichiers électoraux ont été fusionnés avec des données INSEE qui donnent la région, la tranche d'unité urbaine et l'heure de fermeture des bureaux. Pour chacun des bureaux, il existe des informations auxiliaires couplées aux résultats du premier et du deuxième tour. Ici, on s'intéressera à l'estimation des totaux pour chaque candidat du deuxième tour, c'est-à-dire au nombre de voix obtenues et non au pourcentage enregistré par bureau.

B. Installation des différents package

Afin d'effectuer les différentes méthodes d'échantillonnage demandées, on pourra installer plusieurs packages utiles avant de commencer l'analyses statistiques du jeu de données.

```
#### INSTALLATION DES PACKAGES ET LIBRARIES ####

if(!require(sampling)){
        install.packages("sampling")
        library(sampling)
}
if(!require(survey)){
        install.packages("survey")
        library(survey)
}
if(!require(PracTools)){
        install.packages("PracTools")
        library(PracTools)
}

library(survey)
library(sampling)
library(PracTools)
```

Toutefois, on détaillera chacun des calculs et des fonctions utilisées pour cette étude, sans passer par les packages « Survey » et « Sampling » qui simplifient et épurent la lecture du code R, mais ne donnent pas un niveau de détail suffisant pour comprendre chacune des actions et opérations entreprises.

C. Lecture des données

Une fois les packages chargés, il faut ensuite procéder à la lecture du jeu de données en définissant le chemin d'accès du fichier initial.

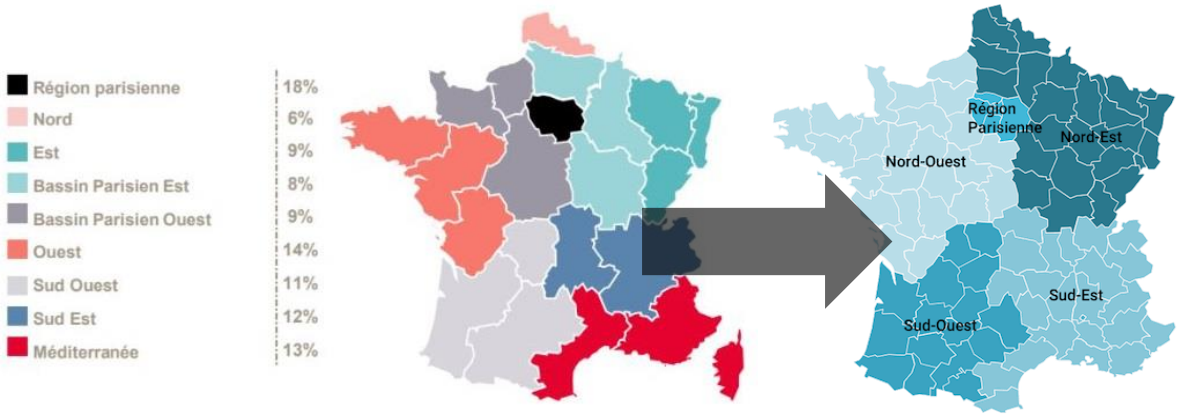
```
#### LECTURE DES DONNEES ####

setwd("D:\\Cours\\STA108\\pres07\\"); #Définis le chemin d'accès des données
P07 <- read.table("pres07.txt", header = TRUE, sep = "\t", quote="\"", dec = ".", fill = TRUE,
comment.char = "", as.is=TRUE); #Charge le fichier des présidentielles 2007
N <- length(P07$Ins.T107);        #Nombre total de bureaux de vote au premier tour (N=64007) ; on aurait
pu également utiliser la fonction "nrow(P07)" qui donne le même résultat
P07[is.na(P07)] <- 0 ; #Remplace les éventuelles valeurs NA par des zéros (Bureaux avec exprimes au
premier tour > 0)

P07$INDEX <- seq(1:N);    #Donne un numéro de 1 à N à chaque bureau de vote
P07$ident <- paste0(P07$D,P07$C,P07$NumBVot); #Créé un identifiant (nombre sans espace du fait de la
fonction "paste0") à partir du code du department, du code de la commune et du numéro du bureau de vote
dans la commune
```

On pourra également effectuer quelques opérations préliminaires afin d'indexer l'ensemble des bureaux à analyser d'une part ; et de les compter d'autre part (N = 64007).

D. Regroupement des régions

Le jeu de données sur lequel l'étude est fondée permet de segmenter les différents bureaux de vote en fonction de leur localisation sur le territoire français. Ainsi, chaque bureau de vote possède une région UDA 9 qui lui est propre. Le terme de régions UDA 9 correspond à un regroupement du territoire français métropolitain en 9 régions distinctes.

Les régions UDA 9 sont notamment utilisées pour bâtir des échantillons selon la méthode des quotas. Toutefois, l'un des objectifs de l'étude est d'opérer une stratification selon le croisement des régions UDA 5, et non UDA 9. De la même façon, le terme de régions UDA 5 correspond à un regroupement du territoire français métropolitain en 5 régions, utilisées également pour construire des échantillons selon la méthode des quotas mais leur usage est cependant moins fréquent que celui des régions UDA9.



Aussi, un regroupement des régions UDA 9 vers une segmentation en cinq régions UDA 5 doit être opéré afin de pouvoir pleinement effectuer la stratification spécifiée par l'étude :

	NORD EST	NORD OUEST	REGION PARISIENNE	SUD-EST	SUD-OUEST
BASSIN PARISIEN EST	8018	0	0	0	0
BASSIN PARISIEN OUEST	0	7291	0	0	0
EST	7229	0	0	0	0
MEDITERRANEE	0	0	0	7041	0
NORD	3456	0	0	0	0
OUEST	0	8094	0	0	0
REGION PARISIENNE	0	0	6660	0	0
SUD-EST	0	0	0	7155	0
SUD-OUEST	0	0	0	0	9063

```
#### REGROUPEMENT DES UDA 5 ####

#Regroupe les régions UDA 9 vers une segmentation en région UDA 5
P07$LIBUDA5 <- P07$LIBUDA9;
P07$LIBUDA5 <- ifelse (P07$LIBUDA9 %in% c("REGION PARISIENNE"), "REGION PARISIENNE",P07$LIBUDA5);
#Region Parisienne
P07$LIBUDA5 <- ifelse (P07$LIBUDA9 %in% c("BASSIN PARISIEN EST", "EST", "NORD"), "NORD
EST",P07$LIBUDA5); #Nord Est
P07$LIBUDA5 <- ifelse (P07$LIBUDA9 %in% c("BASSIN PARISIEN OUEST", "OUEST"), "NORD OUEST",P07$LIBUDA5);
#Nord Ouest
P07$LIBUDA5 <- ifelse (P07$LIBUDA9 %in% c("SUD-OUEST"), "SUD-OUEST",P07$LIBUDA5); #Sud Ouest
P07$LIBUDA5 <- ifelse (P07$LIBUDA9 %in% c("MEDITERRANEE", "SUD-EST"), "SUD-EST",P07$LIBUDA5); #Sud Est
table(P07$LIBUDA5,P07$LIBUDA9); #Affiche la matrice 5x9 du nombre de modifications effectuées

#Trie les données
P07 <- P07[order(P07$LIBUDA5, P07$HORAIRE, P07$D,P07$C,P07$NumBVot) ,]; #Trie les données par LIBUDA5,
puis, pour chaque valeur de LIBUDA5, trie par horaire
```

On triera ensuite les données pour améliorer la lisibilité des données.

II. Formules et calculs à opérer

A. Notations et formules

Une fois les bureaux segmentés en régions UDA 5, on définira les notations et variables propres à l'étude (taille de l'échantillon, nombre de simulations, nombre d'inscrits au premier tour, etc.).

```
#### PRINCIPALES NOTATION & FORMULES ####

n <- 300;        #Taille de l'échantillon
S <- 100;        #Nombre de simulation
P07$PI <- (n * P07$Ins.T107) / sum(P07$Ins.T107); #Calcule les probabilités d'inclusions
P07$Wi <- 1/P07$PI; #Calcule les poids
sum(P07$PI); #Vérifie que la somme des probabilités d'inclusion est égale à n=300

Total.Inscrits.T107 <- sum(P07$Ins.T107); #Calcul du nombre d'inscrits au premier tour
Total.Exprimes.T107 <- sum(P07$Exp.T107); #Calcul du nombre de vo tes exprimés au premier tour
(excluant les votes nuls et blancs)
Total.Exprimes.T207 <- sum(P07$Exp.T207); #Calcul du nombre de votes exprimés au second tour (excluant
les votes nuls et blancs)
```

On vérifie que la somme des probabilités d'inclusion est bien égale à n=300, c'est-à-dire égale à la taille de l'échantillon.

B. Calcul des probabilités d'inclusion

On notera « Proba.inclusion » la fonction qui consiste à calculer chacune des probabilités d'inclusion d'ordre un, c'est-à-dire chacune des probabilités pour que le bureau de vote k de N appartienne à l'échantillon.

```
#### CALCUL DES PROBABILITES D'INCLUSION PROPORTIONNELLES A LA TAILLE DES BUREAUX ####

Proba.inclusion <- function(x, n, y, N, tocorrect=TRUE, alarm=FALSE){
  if(length(x[x <= 0] > 0)) {
    print("Erreur : Problème de couverture car au moins l'une des probabilités d'inclusion est
inférieure à zéro");}     #Vérifie si toutes les probabilités d'inclusion sont supérieures à zéro

  if(length(x[x >= 1] > 0)) {
    print("Erreur : au moins l'un des probabilités d'inclusion est supérieure à 1"); #Vérifie si toutes
les probabilités d'inclusion sont comprises en 0 (exclus) et 1 (inclus)

    if(tocorrect){
        if(length(x[x > 1] > 0)){
          inscrits <- sum(y);
          isover <- FALSE;
          overone <- vector(,N); #Créé un vecteur de taille N composé uniquement de valeurs FALSE
        while(!isover){
          i <- 1;
    while ((i <= N) && (x[i] <= 1)){
            i <- i+1;
          }
    if(i == N+1) {isover <- TRUE;}
    else{
      overone[i] <- TRUE;
      inscrits <- inscrits - y[i];
      for(j in seq(1:N)){
          if(!overone[j]) {x[j] <- n/inscrits*y[j];}
          else {x[j] <- 1;}
            }
          }
        }
      }
    }
  }
  if(!(sum(x)==n)) {print("Erreur : le plan à probabilités inégales n'est pas de taille fixe car la
somme des probabilités d'inclusion est différente de n");} #Calcule la taille du plan pour vérifier
s'il est de taille fixe (sum(P_k) = n)
  if(alarm) {print("Le calcul des probabilités d'inclusion ne comporte pas d'erreurs");}
}
```

On vérifiera ensuite que chacune des probabilités d'inclusion d'ordre un est bien supérieure à 1 pour tout k de N. Dans le cas contraire, on parle de problèmes de couverture : il existerait des bureaux de vote dans la population qui ne pourraient pas figurer dans l'échantillon.

C. Calcul des probabilités inégales

On notera « PI » l'ensemble des probabilités d'inclusion d'ordre un, calculées à partir de la formule précédente, dans le cadre du premier tour de l'élection présidentielle.

```
#### SONDAGE ECHANTILLONNE A PROBIBILITES INEGALES ####

P07$PI <- n*P07$Ins.T107/Total.Inscrits.T107; #Calcul des probabilité inégales pour le premier tour

Proba.inclusion(P07$PI, n, P07$Ins.T107, N, alarm=TRUE);
```

D. Calcul des probabilités pour un plan stratifié

On définira les différentes variables et notations utilisées dans le cadre d'un plan stratifié, notamment le nombre de strates ainsi que le nombre de bureaux de vote par strate.

```
#### SONDAGE PAR PLAN STRATIFIE ####

#La stratification est effectuée selon le croisement "Région UDA5" x "Horaire de fermeture du bureau"

#Applique la fonction MATCH pour chaque sous-vecteur de chaque région (le -1 est nécessaire car les
vecteurs dans R sont indexé à partir de 1 et non 0)
Classement <- c(
  match(c("18h","19h","20h"), P07$HORAIRE[P07$LIBUDA5=="NORD EST"]),
  match(c("18h","19h","20h"), P07$HORAIRE[P07$LIBUDA5=="NORD OUEST"])+match("NORD OUEST", P07$LIBUDA5)-
1,
  match(c("18h","19h","20h"), P07$HORAIRE[P07$LIBUDA5=="REGION PARISIENNE"])+match("REGION PARISIENNE",
P07$LIBUDA5)-1,
  match(c("18h","19h","20h"), P07$HORAIRE[P07$LIBUDA5=="SUD-EST"])+match("SUD-EST", P07$LIBUDA5)-1,
  match(c("18h","19h","20h"), P07$HORAIRE[P07$LIBUDA5=="SUD-OUEST"])+match("SUD-OUEST", P07$LIBUDA5)-1,
  N+1)

Nombre.Strates <- 15;     #Nombre de strates
Nh <- vector("numeric", Nombre.Strates);  #Nombre de bureaux de votes dans chaque strate
nh <- vector("numeric", Nombre.Strates);  #Taille de l'échantillon pour chaque strate
P07$PIh <- P07$PI;        #Probabilités inégales pour chaque strate au premier tour
for (h in seq(1:Nombre.Strates)){
  Nh[h] <- Classement[h+1] - Classement[h];
  nh[h] <- Nh[h]/N*n;     #Etant donné que la taille de l'échantillon n'est pas forcément une valeur
entière (les cas où nh[h]!=INTEGER), on vérifie cette condition par le biais d'une boucle permettant de
vérifier si le plan est de taille fixe
  if((nh[h] < 1) && (nh[h] > 0)){nh[h] <- 1;}      #Lorsque la taille de l'échantillon est comprise
entre 0 et 1, on donne la valeur à 1 à l'échantillon afin de ne pas perdre d'information sur la strate
}

Nb.Dec <- n-sum(trunc(nh));
Rest.Dec <- nh - trunc(nh);
Class.dec <- order(Rest.Dec, decreasing = TRUE);
nh <- trunc(nh); #Tronque la taille de l'échantillon de chaque strate

#Correction par troncature des potentielles erreurs décimales
for (i in seq(1:Nb.Dec)){
  nh[Class.dec[i]] <- nh[Class.dec[i]]+1;
}

for (h in seq(1:Nombre.Strates)){
  N.T107_h <- sum(P07$Ins.T107[seq(Classement[h], Classement[h+1]-1)]);     #Nombre d'inscrits au
premier tour pour chaque strate

  for(i in seq(Classement[h], Classement[h+1]-1)) {
    P07$PIh[i] <- nh[h]*P07$Ins.T107[i]/N.T107_h;  #Calcul des probabiliés d'inclusion pour chaque
strate
  }

  Proba.inclusion(P07$PIh[seq(Classement[h], Classement[h+1]-1)], nh[h],
P07$Ins.T107[seq(Classement[h], Classement[h+1]-1)], Nh[h]);
}

Proba.inclusion(P07$PIh, n, P07$Ins.T107, N, tocorrect=FALSE, alarm=TRUE);
```

Après avoir vérifié que le plan est bien de taille fixe, on calculera les probabilités d'inclusions d'ordre un dans le cadre d'un plan stratifié.

III. Etude de distributions d'échantillonnage selon la méthode

A. Calculs des véritables résultats

On cherchera ensuite à afficher la moyenne et l'intervalle de confiance pour chacun des deux candidats dans le cadre d'un sondage aléatoire simple et sans remise.

```
#### AFFICHAGE DES VERITABLES RESULTATS PROVENANT DU JEU DE DONNEES ORIGINAL ####

print(c(sum(P07$NbVoix.SARK_T207)*100/Total.Exprimes.T207, 0), digits=4); #Affiche la moyenne et
l'intervalle de confiance (Ic = 1.96*sqrt(var(Sarkozy)/s)) pour Nicolas Sarkozy

print(c(sum(P07$NbVoix.ROYA_T207)*100/Total.Exprimes.T207, 0), digits=4); #Affiche la moyenne et
l'intervalle de confiance (Ic = 1.96*sqrt(var(Royale)/s)) pour Ségolène Royale
```

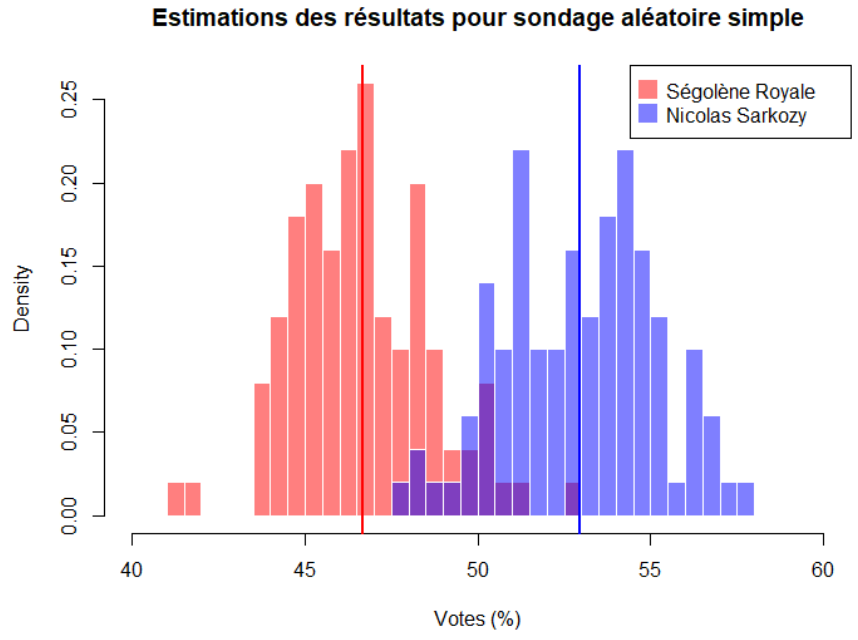
A partir des données fournies par le ministère de l'intérieur, on obtient les résultats définitifs officiels suivants :

Candidats	Ségolène Royale	Nicolas Sarkozy
Nombre de voix	16 042 083	18 326 396
Pourcentage de votes	46,676%	53,323%

B. Simulation des résultats pour un plan aléatoire simple

On pourra ensuite procéder aux 100 simulations afin d'obtenir les estimations des résultats pour un plan simple.

```
#### SIMULATION POUR UN SONDAGE ALEATOIRE SIMPLE ####

s <- 1;
for (s in seq(1:S)) {
  x.SAS <- sample(P07$INDEX, n);
  SAS.SARK_T207[s] <- N/n*sum(P07$NbVoix.SARK_T207[x.SAS])*100/Total.Exprimes.T207;
  SAS.ROYA_T207[s] <- N/n*sum(P07$NbVoix.ROYA_T207[x.SAS])*100/Total.Exprimes.T207;
}

#Affiche les estimations des résultats par le biais d'un sondage aléatoire simple
print(c(mean(SAS.SARK_T207), 1.96*sqrt(var(SAS.SARK_T207)/s)), digits=2);
print(c(mean(SAS.ROYA_T207), 1.96*sqrt(var(SAS.ROYA_T207)/s)), digits=2);
```



Les estimations par le biais d'un plan stratifié puis d'un sondage aléatoire simple donnent les résultats suivants : une moyenne de 52,955% avec un intervalle de confiance de plus ou moins 0,434% pour Nicolas Sarkozy d'une part ; et une moyenne de 46,658% avec un intervalle de confiance de plus ou moins 0,394% pour Ségolène Royale d'autre part.

Les intervalles de confiance obtenus par le calcul sont assez élevés (≈0,4%), ce qui peut s'expliquer intuitivement par la méthode d'échantillonnage utilisée. On procède à un sondage échantillonné à probabilités inégales afin d'améliorer la précision.

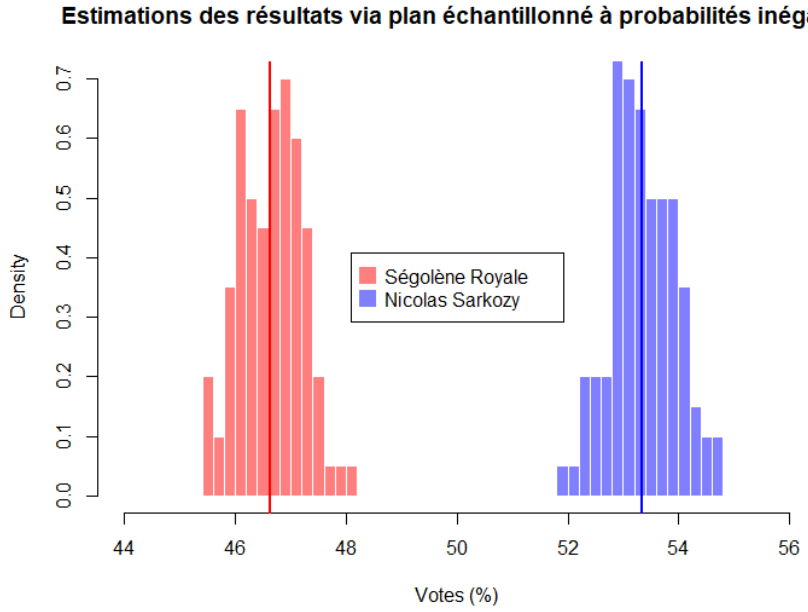
Candidats	Ségolène Royale		Nicolas Sarkozy	
	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Estimations SAS	46,658%	± 0,394%	52,955%	± 0,434%
Delta SAS	-0,018%	± 0,394%	-0,368%	± 0,434%

C. Simulation des résultats pour un plan échantillonné à probabilités inégales

On procédera ensuite aux 100 simulations dans le cadre d'un plan échantillonné à probabilités inégales afin d'obtenir les estimations des résultats.

```
#### SIMULATION POUR UN PLAN ECHANTILLONNE A PROBABILITES INEGALES ####

s <- 1;
for (s in seq(1:S)) {

  x.PI <- sample(P07$INDEX,n,replace=FALSE,P07$PI[P07$INDEX]);
  PI.SARK_T207[s] <- sum(P07$NbVoix.SARK_T207[x.PI]/P07$PI[x.PI])*100/Total.Exprimes.T207;
  PI.ROYA_T207[s] <- sum(P07$NbVoix.ROYA_T207[x.PI]/P07$PI[x.PI])*100/Total.Exprimes.T207;
}

#Affiche les estimations des résultats par le biais d'un plan échantillonné à probabilités inégales
print(c(mean(PI.SARK_T207), 1.96*sqrt(var(PI.SARK_T207)/s)), digits=2);
print(c(mean(PI.ROYA_T207), 1.96*sqrt(var(PI.ROYA_T207)/s)), digits=2);

hist(PI.ROYA_T207, freq=FALSE, border='white', col=rgb(1,0,0,0.5), xlab='Votes (%)', xlim=c(44,56),
main='Estimations des résultats via plan échantillonné à probabilités inégales',breaks=10)
hist(PI.SARK_T207, freq=FALSE, border='white', col=rgb(0,0,1,0.5), xlab='Votes (%)', xlim=c(45,55),
main='Estimations des résultats via plan échantillonné à probabilités inégales',breaks=20, add=T) ;

#Ajoute les moyennes des distributions
abline(v = mean(PI.ROYA_T207), col = "red", lwd = 2)
abline(v = mean(PI.SARK_T207), col = "blue", lwd = 2);

#Ajoute la légende
legend("center", legend=c("Ségolène Royale","Nicolas Sarkozy"), col=c(rgb(1,0,0,0.5), rgb(0,0,1,0.5)),
pt.cex=2, pch=15);
```

Après avoir affiché la distribution statistique des résultats estimés à probabilités inégales, on obtient d'une part une moyenne de 53,347% avec un intervalle de confiance de plus ou moins 0,114% pour Nicolas Sarkozy ; et d'autre part une moyenne de 46,634% avec un intervalle de confiance de plus ou moins 0,109% pour Ségolène Royale.



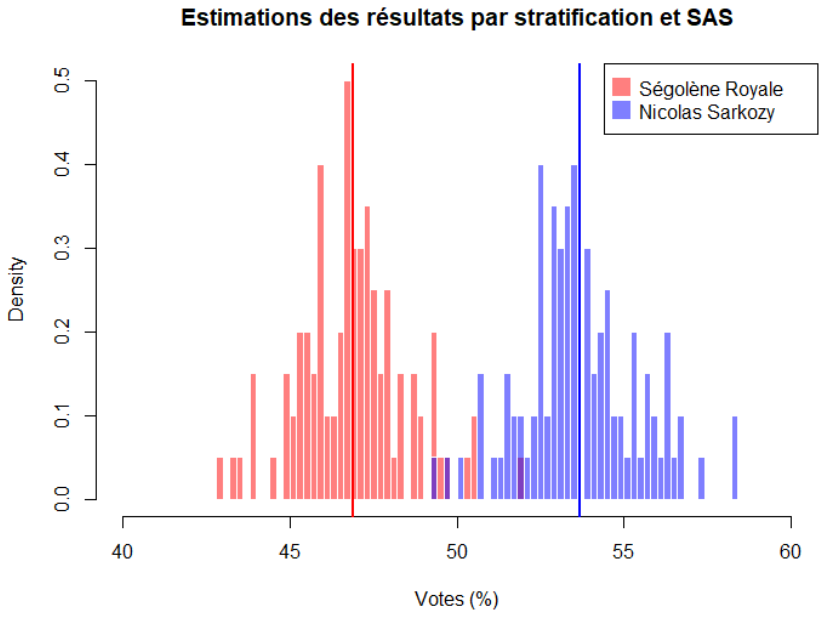
Candidats	Ségolène Royale		Nicolas Sarkozy	
	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Estimations PI	46,634%	± 0,109%	53,347%	± 0,114%
Delta PI	-0,042%	± 0,109%	+0,024%	± 0,114%

D. Simulation des résultats pour un plan stratifié

Le plan stratifié pourra être construit à partir d'un choix d'allocation de manière proportionnelle en partant du principe que l'échantillon obtenu est dit « représentatif ».

```
#### SIMULATION POUR UN PLAN STRATIFIE SUIVIS DE DEUX METHODES D'ECHANTILLONNAGE ####

s <- 1;
for (s in seq(1:S)) {
  x.Strat.avec.SAS <- c();
  x.Strat.avec.PI <- c();

  for (h in seq(1:Nombre.Strates)){
    #Avec Sondage Aléatoire Simple
    x.Strat.avec.SAS <- c(x.Strat.avec.SAS, sample(seq(Classement[h], Classement[h+1]-1),nh[h]));

    #Avec Probabilités inégales
    x.Strat.avec.PI <- c(x.Strat.avec.PI, sample(seq(Classement[h], Classement[h+1]-
1),nh[h],replace=FALSE,P07$PIh[seq(Classement[h], Classement[h+1]-1)]));
  }

#Calcul à partir de N/n du fait des allocations proportionnelles (Nh/nh=N/n)
Strat.avec.SAS.ROYA_T207[s] <- N/n*sum(P07$NbVoix.ROYA_T207[x.Strat.avec.SAS])*100/Total.Exprimes.T207;


Strat.avec.SAS.SARK_T207[s] <- N/n*sum(P07$NbVoix.SARK_T207[x.Strat.avec.SAS])*100/Total.Exprimes.T207;


par(mfcol=c(1,1));
hist(Strat.avec.SAS.ROYA_T207, freq=FALSE, border='white', col=rgb(1,0,0,0.5), xlab='Votes (%)',
xlim=c(40,60), main='Estimations des résultats par stratification et SAS',breaks=40)
hist(Strat.avec.SAS.SARK_T207, freq=FALSE, border='white', col=rgb(0,0,1,0.5), xlab='Votes (%)',
xlim=c(42,58), main='Estimations des résultats par stratification et SAS',breaks=40, add=T) ;

#Ajoute les moyennes des distributions
abline(v = mean(Strat.avec.SAS.ROYA_T207), col = "red", lwd = 2)
abline(v = mean(Strat.avec.SAS.SARK_T207), col = "blue", lwd = 2);

#Ajoute la légende
legend("topright", legend=c("Ségolène Royale","Nicolas Sarkozy"), col=c(rgb(1,0,0,0.5), rgb(0,0,1,0.5)),
pt.cex=2, pch=15);
```

Les estimations par le biais d'un plan stratifié puis d'un sondage aléatoire simple donnent les résultats suivants : une moyenne de 53,676% avec un intervalle de confiance de plus ou moins 0,344% pour Nicolas Sarkozy d'une part ; et une moyenne de 46,879% avec un intervalle de confiance de plus ou moins 0,317% pour Ségolène Royale d'autre part.



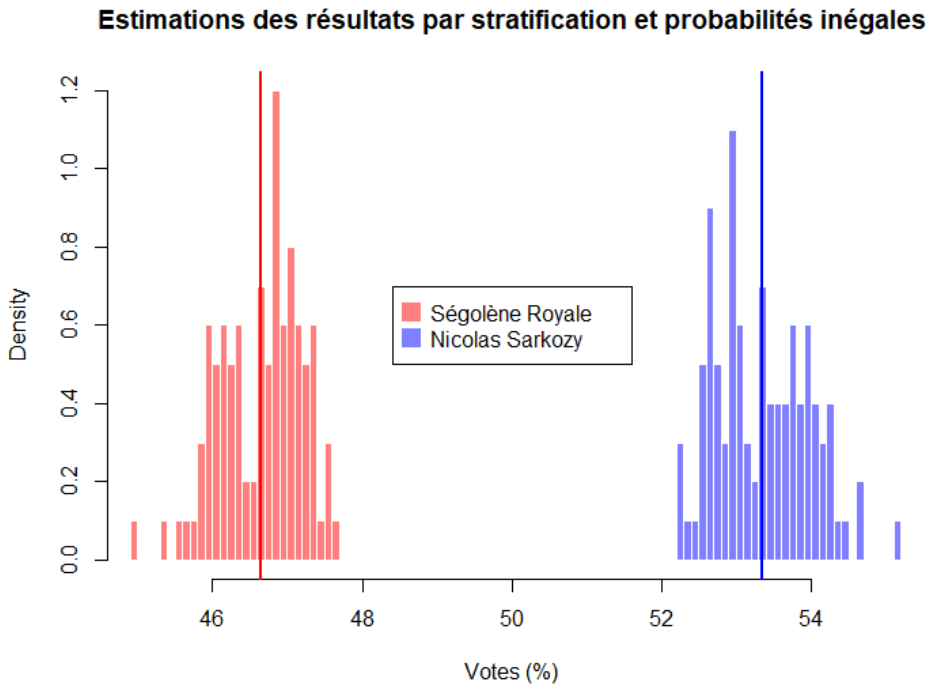
A partir d'un sondage aléatoire simple au sein des différentes strates, on obtient les résultats suivants :

Candidats	Ségolène Royale		Nicolas Sarkozy	
	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Estimations Stratification/SAS	46,879%	± 0,317%	53,676%	± 0,344%
Delta Stratification/SAS	+0,203%	± 0,317%	+0,353%	± 0,344%

Afin de gagner en précision et de réduire l'intervalle de confiance (≈0,3%), on procède à un échantillonnage avec probabilités inégales et sans remise.

```
#Calcul à partir des probabilités d'inclusion
Strat.avec.PI.ROYA_T207[s] <-
sum(P07$NbVoix.ROYA_T207[x.Strat.avec.PI]/P07$PIh[x.Strat.avec.PI])*100/Total.Exprimes.T207;

Strat.avec.PI.SARK_T207[s] <-
sum(P07$NbVoix.SARK_T207[x.Strat.avec.PI]/P07$PIh[x.Strat.avec.PI])*100/Total.Exprimes.T207;
}

#Affiche les estimations des résultats par le biais d'un plan stratifié puis d'un sondage à probabilités
inégales
print(c(mean(Strat.avec.PI.SARK_T207), 1.96*sqrt(var(Strat.avec.PI.SARK_T207)/s)), digits=3);
print(c(mean(Strat.avec.PI.ROYA_T207), 1.96*sqrt(var(Strat.avec.PI.ROYA_T207)/s)), digits=3);

par(mfcol=c(1,1));
hist(Strat.avec.PI.ROYA_T207, freq=FALSE, border='white', col=rgb(1,0,0,0.5), xlab='Votes (%)',
xlim=c(45,55), main='Estimations des résultats par stratification et probabilités inégales',breaks=30)
hist(Strat.avec.PI.SARK_T207, freq=FALSE, border='white', col=rgb(0,0,1,0.5), xlab='Votes (%)',
xlim=c(45,55), main='Estimations des résultats par stratification et probabilités inégales',breaks=30,
add=T) ;

#Ajoute les moyennes des distributions
abline(v = mean(Strat.avec.PI.ROYA_T207), col = "red", lwd = 2)
abline(v = mean(Strat.avec.PI.SARK_T207), col = "blue", lwd = 2);

#Ajoute la légende
legend("center", legend=c("Ségolène Royale","Nicolas Sarkozy"), col=c(rgb(1,0,0,0.5), rgb(0,0,1,0.5)),
pt.cex=2, pch=15);
```

Les estimations par le biais d'un plan stratifié puis d'un sondage aléatoire simple donnent les résultats suivants : une moyenne de 53,337% avec un intervalle de confiance de plus ou moins 0,123% pour Nicolas Sarkozy d'une part ; et une moyenne de 46,645% avec un intervalle de confiance de plus ou moins 0,107% pour Ségolène Royale d'autre part.



A partir d'un sondage à probabilités inégales au sein des différentes strates, on obtient les résultats suivants :

Candidats	Ségolène Royale		Nicolas Sarkozy	
	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Estimations Stratification / PI	46,645%	± 0,107%	53,337%	± 0,123%
Delta Stratification / PI	-0,031%	± 0,107%	-0,014%	± 0,123%

IV. Correction du biais des grandes villes

A. Identification des bureaux de vote

Afin d'effectuer la correction du biais, on identifiera les bureaux en fonction de leurs horaires de fermeture, puis on les regroupera par tranche d'horaire (18h, 19h et 20h).

```
#### IDENTIFICATION DES BUREAUX DE 18H ET DE 18H+19H ####

#Regroupe les bureaux par horaire (18h, 19h et 20h)
Bureaux.18h <- c();
Bureaux.19h <- c();

for (i in seq(1:sum(nh))){
  if(P07$HORAIRE[x.Strat.avec.PI[i]]=="18h"){
    Bureaux.18h <- c(Bureaux.18h, x.Strat.avec.PI[i]); #Regroupe les bureaux fermant à 18h
  }
  else if(P07$HORAIRE[x.Strat.avec.PI[i]]=="19h"){
    Bureaux.19h <- c(Bureaux.19h, x.Strat.avec.PI[i]); #Regroupe les bureaux fermant à 19h
  }
}

Bureaux.19h <- c(Bureaux.18h, Bureaux.19h);
n.bureaux.18h  <- length(Bureaux.18h);
n.bureaux.19h  <- length(Bureaux.19h);
```

B. Calculs intermédiaires

```
#### CALCULS DE LA PENTE ESTIMEE ET DU COEFFICIENT DE CORRELATION ####

#Calcule le nombre de votes exprimés, au premier tour et second tour, par horaire (18h et 19h)
Exprimes.18h.T107 <- sum(P07$Exp.T107[P07$HORAIRE =="18h"]);
Exprimes.18h.T207 <- sum(P07$Exp.T207[P07$HORAIRE =="18h"]);
Exprimes.19h.T107 <- sum(P07$Exp.T107[P07$HORAIRE !="20h"]);
Exprimes.19h.T207 <- sum(P07$Exp.T207[P07$HORAIRE !="20h"]);

#Définis les totaux des bureaux fermant à 18h et 19h
Total <- vector("numeric", 12); #Créé un vecteur de taille 12 correspondant au nombre de candidats au
premier tour
Total.PI.18h <- vector("numeric", 12);
Total.PI.19h <- vector("numeric", 12);

#Définis les pentes estimées de la droite de régression pour les bureaux fermant à 18h et 19h
b.ROYA.18h <- vector("numeric", 12);
b.SARK.18h <- vector("numeric", 12);
b.ROYA.19h <- vector("numeric", 12);
b.SARK.19h <- vector("numeric", 12);

#Définis les coéfficients de corrélation pour les bureaux fermant à 18h et 19h
Rho.ROYA.18h <- vector("numeric", 12);
Rho.SARK.18h <- vector("numeric", 12);
Rho.ROYA.19h <- vector("numeric", 12);
Rho.SARK.19h <- vector("numeric", 12);

#Procède aux calculs des pentes de la droite de régression et des coéfficients de corrélation pour les
différentes horaires mesurées (18h et 19h)
for(i in seq(1:12)){

  #Calcule les totaux
  Total[i] <- sum(P07[17+i]);
  Total.PI.18h[i] <- sum(P07[Bureaux.18h, 17+i]/P07$PIh[Bureaux.18h]);
  Total.PI.19h[i] <- sum(P07[Bureaux.19h, 17+i]/P07$PIh[Bureaux.19h]);

  #Calcule les pentes de régression ainsi que les coéffiencients de corrélation pour la candidate
Ségolène Royale
  b.ROYA.18h[i] <- sqrt(cov(P07$NbVoix.ROYA_T207[Bureaux.18h], P07[Bureaux.18h,
17+i]))/var(P07[Bureaux.18h, 17+i]);
  Rho.ROYA.18h[i] <- cor(P07$NbVoix.ROYA_T207[Bureaux.18h], P07[Bureaux.18h, 17+i]);
  b.ROYA.19h[i] <- sqrt(cov(P07$NbVoix.ROYA_T207[Bureaux.19h], P07[Bureaux.19h,
17+i]))/var(P07[Bureaux.19h, 17+i]);
  Rho.ROYA.19h[i] <- cor(P07$NbVoix.ROYA_T207[Bureaux.19h], P07[Bureaux.19h, 17+i]);

  #Calcule les pentes de régression ainsi que les coéffiencients de corrélation pour le candidat Nicolas
Sarkozy
  b.SARK.18h[i] <- sqrt(cov(P07$NbVoix.SARK_T207[Bureaux.18h], P07[Bureaux.18h,
17+i]))/var(P07[Bureaux.18h, 17+i]);
  Rho.SARK.18h[i] <- cor(P07$NbVoix.SARK_T207[Bureaux.18h], P07[Bureaux.18h, 17+i]);
  b.SARK.19h[i] <- sqrt(cov(P07$NbVoix.SARK_T207[Bureaux.19h], P07[Bureaux.19h,
17+i]))/var(P07[Bureaux.19h, 17+i]);
  Rho.SARK.19h[i] <- cor(P07$NbVoix.SARK_T207[Bureaux.19h], P07[Bureaux.19h, 17+i]);
}

#Calcule le nombre de votes exprimés par sondage échantillonné à probabilités inégales
Exprimes.PI.18h.T207 <- sum(P07$Exp.T207[Bureaux.18h]/P07$PIh[Bureaux.18h]);
Exprimes.PI.19h.T207 <- sum(P07$Exp.T207[Bureaux.19h]/P07$PIh[Bureaux.19h]);
```

C. Estimations brutes et redressées avant 19h

```
#### ESTIMATIONS BRUTES ET REDRESSEES AVANT 19H ####
#Calcule les estimations brutes sur les résultats du premier tour avant 19h
Total.brut.voix.PI.ROYA.18h <- sum(P07$NbVoix.ROYA_T207[Bureaux.18h]/P07$PIh[Bureaux.18h]);
Total.brut.voix.PI.SARK.18h <- sum(P07$NbVoix.SARK_T207[Bureaux.18h]/P07$PIh[Bureaux.18h]);
print(c(Total.brut.voix.PI.SARK.18h*100/Exprimes.PI.18h.T207,0 ), digits=5);
print(c(Total.brut.voix.PI.ROYA.18h*100/Exprimes.PI.18h.T207,0 ), digits=5);

#Calcule les estimations redressées (total et variance) sur les résultats du premier tour avant 19h
Total.redresse.voix.PI.ROYA.18h <- (Total.brut.voix.PI.ROYA.18h + b.ROYA.18h[8]*(Total[8]-
Total.PI.18h[8]))*100/Exprimes.PI.18h.T207 ;
Variance.total.redresse.voix.PI.ROYA.18h <- N^2*(1-
n.bureaux.18h/N)*var(P07$NbVoix.ROYA_T207[Bureaux.18h])/n.bureaux.18h*(1-
Rho.ROYA.18h[8]^2)*(100/Exprimes.PI.18h.T207)^2;

Total.redresse.voix.PI.SARK.18h <- (Total.brut.voix.PI.SARK.18h + b.SARK.18h[12]*(Total[12]-
Total.PI.18h[12]))*100/Exprimes.PI.18h.T207 ;
Variance.total.redresse.voix.PI.SARK.18h <- N^2*(1-
n.bureaux.18h/N)*var(P07$NbVoix.SARK_T207[Bureaux.18h])/n.bureaux.18h*(1-
Rho.SARK.18h[8]^2)*(100/Exprimes.PI.18h.T207)^2;

#Affiche les estimations redressées (total et variance) sur les résultats du premier tour avant 19h
print(c(Total.redresse.voix.PI.SARK.18h, 1.96*sqrt(Variance.total.redresse.voix.PI.SARK.18h/sum(nh))),
digits=3);
print(c(Total.redresse.voix.PI.ROYA.18h, 1.96*sqrt(Variance.total.redresse.voix.PI.ROYA.18h/sum(nh))),
digits=3);
```

Avant 19h, on obtient donc les résultats suivants :

Candidats	Ségolène Royale		Nicolas Sarkozy	
	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Estimations brutes	45,639%	± 0,000%	54,361%	± 0,000%
Estimations redressées	45,944%	± 0,116%	54,628%	± 0,377%
Delta Estimations redressées	-0,732%	± 0,116%	+1,305%	± 0,377%

D. Estimations brutes et redressées à 20h

```
#### ESTIMATIONS BRUTES ET REDRESSEES A 20H ####
#Calcule les estimations brutes sur les résultats du premier tour à 20h
Total.brut.voix.PI.ROYA.19h <- sum(P07$NbVoix.ROYA_T207[Bureaux.19h]/P07$PIh[Bureaux.19h]);
Total.brut.voix.PI.SARK.19h <- sum(P07$NbVoix.SARK_T207[Bureaux.19h]/P07$PIh[Bureaux.19h]);

#Affiche les estimations brutes sur les résultats du premier tour à 20h
print(c(Total.brut.voix.PI.SARK.19h*100/Exprimes.PI.19h.T207,0 ), digits=5);
print(c(Total.brut.voix.PI.ROYA.19h*100/Exprimes.PI.19h.T207,0 ), digits=5);

#Calcule les estimations redressées sur les résultats du premier tour à 20h
Total.redresse.voix.PI.ROYA.19h <- (Total.brut.voix.PI.ROYA.19h + b.ROYA.19h[8]*(Total[8]-
Total.PI.19h[8]))*100/Exprimes.PI.19h.T207 ;
Variance.total.redresse.voix.PI.ROYA.19h <- N^2*(1-
n.bureaux.19h/N)*var(P07$NbVoix.ROYA_T207[Bureaux.19h])/n.bureaux.19h*(1-
Rho.ROYA.19h[8]^2)*(100/Exprimes.PI.19h.T207)^2;

Total.redresse.voix.PI.SARK.19h <- (Total.brut.voix.PI.SARK.19h + b.SARK.19h[12]*(Total[12]-
Total.PI.19h[12]))*100/Exprimes.PI.19h.T207 ;
Variance.total.redresse.voix.PI.SARK.19h <- N^2*(1-
n.bureaux.19h/N)*var(P07$NbVoix.SARK_T207[Bureaux.19h])/n.bureaux.19h*(1-
Rho.SARK.19h[8]^2)*(100/Exprimes.PI.19h.T207)^2;

#Affiche les estimations redressées sur les résultats du premier tour à 20h
print(c(Total.redresse.voix.PI.SARK.19h, 1.96*sqrt(Variance.total.redresse.voix.PI.SARK.19h/sum(nh))),
digits=3);
print(c(Total.redresse.voix.PI.ROYA.19h, 1.96*sqrt(Variance.total.redresse.voix.PI.ROYA.19h/sum(nh))),
digits=2);
```

A 20h, on obtient donc les résultats suivants :

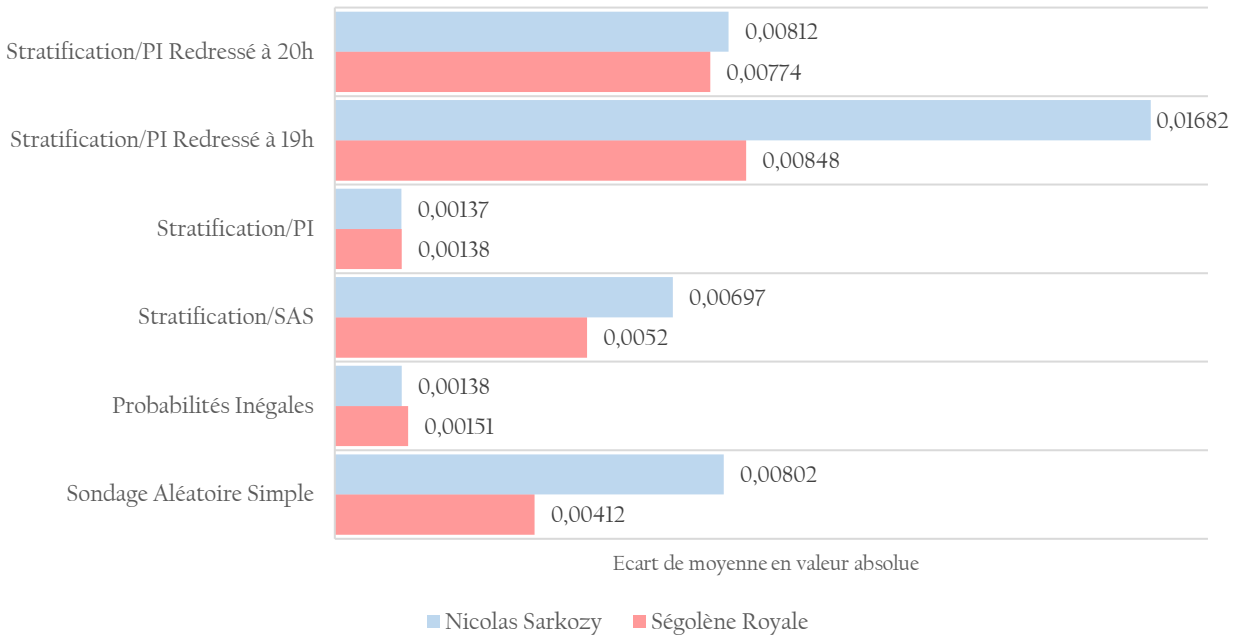
Candidats	Ségolène Royale		Nicolas Sarkozy	
	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Estimations brutes	47,332%	± 0,00%	52,668%	± 0,00%
Estimations redressées	47,385%	± 0,065%	52,732%	± 0,221%
Delta Estimations redressées	+0,709%	± 0,065%	-0,591%	± 0,221%

Conclusion de l'étude

A partir des estimations de résultats obtenus à partir des différentes méthodes d'échantillonnage utilisées, on peut comparer les moyennes combinées aux intervalles de confiance respectifs afin d'identifier les méthodes les plus précises et efficaces.

Candidats	Ségolène Royale		Nicolas Sarkozy	
Estimations obtenues :	Moyenne	Intervalle de confiance	Moyenne	Intervalle de confiance
Résultats Fichier	46,676%	± 0,000%	53,323%	± 0,000%
Sondage Aléatoire Simple	46,658%	± 0,394%	52,955%	± 0,434%
Plan à Probabilités Inégales	46,634%	± 0,109%	53,347%	± 0,114%
Stratification & SAS	46,879%	± 0,317%	53,676%	± 0,344%
Stratification & PI	46,645%	± 0,107%	53,337%	± 0,123%
Stratification & PI redressé à 19h	45,944%	± 0,116%	54,628%	± 0,377%
Stratification & PI redressé à 20h	47,385%	± 0,065%	52,732%	± 0,221%

En ajoutant les intervalles de confiance aux moyennes en valeurs absolues, on observe que la méthode d'échantillonnage par probabilités inégales semble être la plus intéressante compte tenu des très bonnes estimations obtenues, que ce soit par la méthode classique ou par le biais d'un plan stratifié.



Toutefois, la stratification accompagnée d'un plan à probabilités inégales devrait intuitivement être la méthode la plus précise lorsque les résultats ont été redressés à 19h puis à 20h. Il semblerait donc que le calcul effectué dans la dernière partie de l'étude (dédiée au redressement des résultats) soit partiellement erroné bien que la méthode utilisée soit louable.

Annexes

Listes des variables issues des données propres à l'élection présidentielle 2007

Numéro	Variable	Contenu
1	D	Code département
2	C	Code commune
3	CODGEO	D+C
4	LIBGEO	Libellé commune
5	REG	Région INSEE
6	DEP	Code département
7	TUU2010	Tranche d'unité urbaine
8	LIBREGION	Libellé de région INSEE
9	LIBTUU2010	Libellé de TUU2010
10	POP_MUN_2008	Population municipale 2008
11	HORAIRE	Horaire de fermeture du bureau
12	NumBVot	Numéro de bureau de vote dans la commune
13	LIBUDA9	Libellé de région UDA9
14	CODEUDA	Code région UDA9
15	Ins.T107	Inscrits 1er tour 2007
16	Vot.T107	Votants 1er tour 2007
17	Exp.T107	Exprimés 1er tour 2007
18	NbVoix.BESA_T107	Nombre de voix exprimées au premier tour
19	NbVoix.BUFF_T107	
20	NbVoix.SCHI_T107	
21	NbVoix.BAYR_T107	
22	NbVoix.BOVE_T107	
23	NbVoix.VOYN_T107	
24	NbVoix.VILL_T107	
25	NbVoix.ROYA_T107	
26	NbVoix.NIHO_T107	
27	NbVoix.LEPE_T107	
28	NbVoix.LAGU_T107	
29	NbVoix.SARK_T107	
30	Ins.T207	Nombre d'inscrits au second tour
31	Vot.T207	Nombre de votants au second tour
32	Exp.T207	Nombre de votes exprimés au second tour
33	NbVoix.SARK_T207	Nombre de votes exprimés au second tour
34	NbVoix.ROYA_T207	
35	PctVoix.BESA_T107	Pourcentage de votes exprimés au premier tour
36	PctVoix.BUFF_T107	
37	PctVoix.SCHI_T107	
38	PctVoix.BAYR_T107	
39	PctVoix.BOVE_T107	
40	PctVoix.VOYN_T107	
41	PctVoix.VILL_T107	
42	PctVoix.ROYA_T107	
43	PctVoix.NIHO_T107	
44	PctVoix.LEPE_T107	
45	PctVoix.LAGU_T107	
46	PctVoix.SARK_T107	
47	PctVoix.SARK_T207	Pourcentage sur les votes exprimés au second tour
48	PctVoix.ROYA_T207	
49	ident	