Imputation multiple à l'aide des méthodes d'analyse factorielle

Vincent Audigier & Julie Josse & François Husson

Agrocampus Rennes

45e journées de Statistique, Toulouse, 28 Mai 2013

Données manquantes

			X				Y
NA					*NA*	*NA*	
			NA				
	NA						*NA*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
		NA	*NA*	*NA*			

Données manquantes

			X				Y
NA					*NA*	*NA*	
			NA				
	NA						*NA*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
		NA	*NA*	*NA*			

- Suppression des individus : cas complet

Données manquantes

			X				Y
NA					*NA*	*NA*	
			NA				
	NA						*NA*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
		NA	*NA*	*NA*			

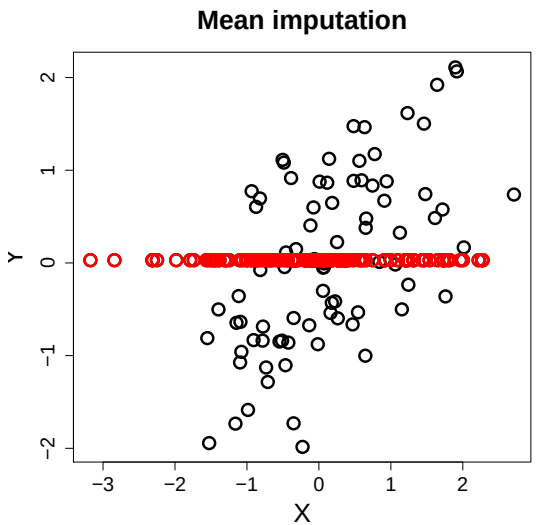
- Suppression des individus : cas complet
- Adaptation de la méthode au cas incomplet

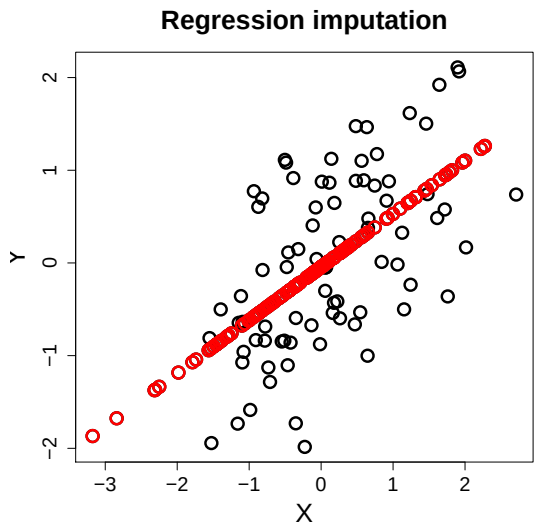
Données manquantes

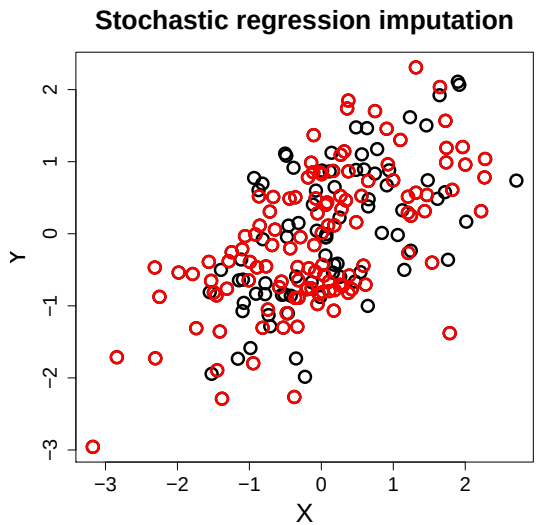
			X				Y
NA					*NA*	*NA*	
			NA				
	NA						*NA*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
		NA	*NA*	*NA*			

- Suppression des individus : cas complet
- Adaptation de la méthode au cas incomplet
- Imputation

Méthodes d'imputation simple





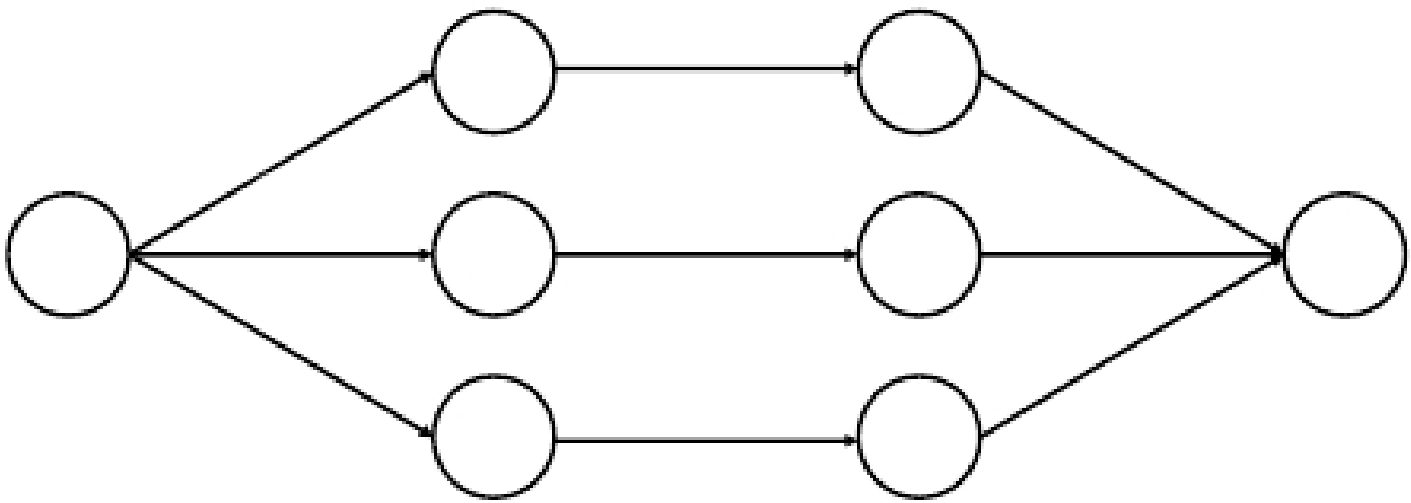


$\mu_y = 0$	0.01	0.01	0.01
$\sigma_y^2 = 1$	0.5	0.72	0.99
$\rho = 0.6$	0.30	0.78	0.59
$IC_{\mu_y 95\%}$	39.4	61.6	70.8

Principe

IM comme solution au problème de sous-estimation de la variance

Tableau incomplet Imputation Analyse Agrégation



Principe

Estimation du paramètre d'intérêt

$$\hat{\theta} = \frac{1}{M} \sum_{m=1}^{M} \hat{\theta}_m$$

Décomposition de la variance

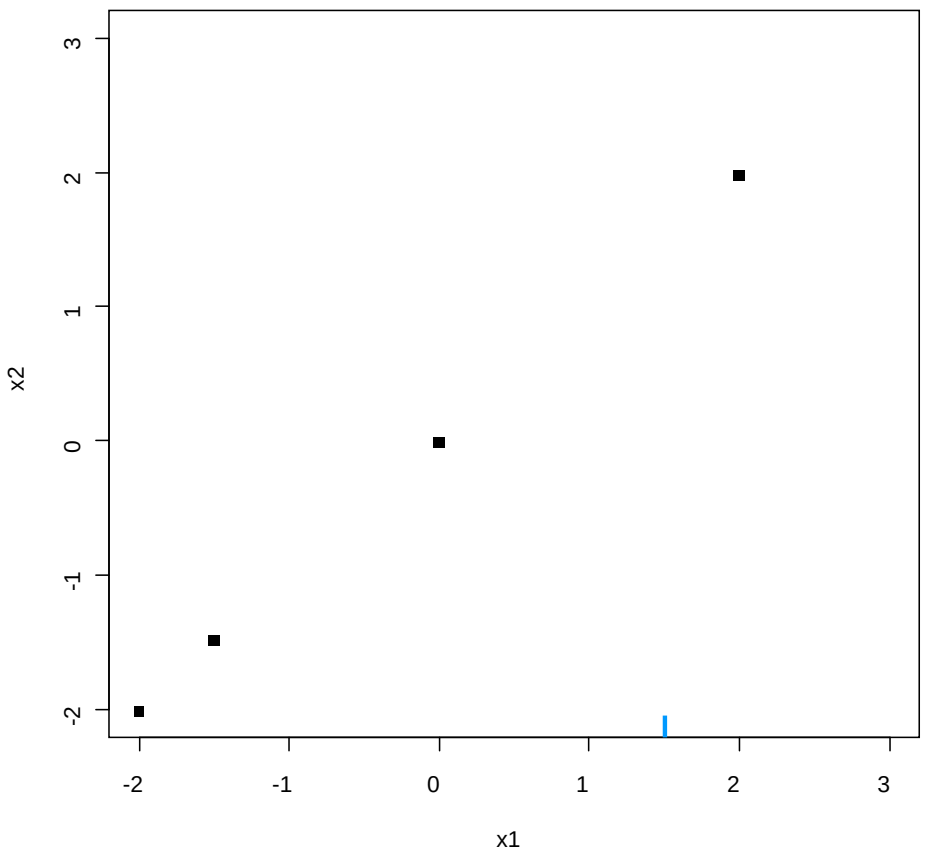
$$T = \frac{1}{M} \sum_{m=1}^{M} \widehat{Var}\left(\hat{\theta}_m\right) + \left(1 + \frac{1}{M}\right) \frac{1}{M-1} \sum_{m=1}^{M} \left(\hat{\theta}_m - \hat{\theta}\right)^2$$

Une méthode d'imputation multiple repose sur une méthode d'imputation simple

Algorithme : ACP itérative

Le jeu de données

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

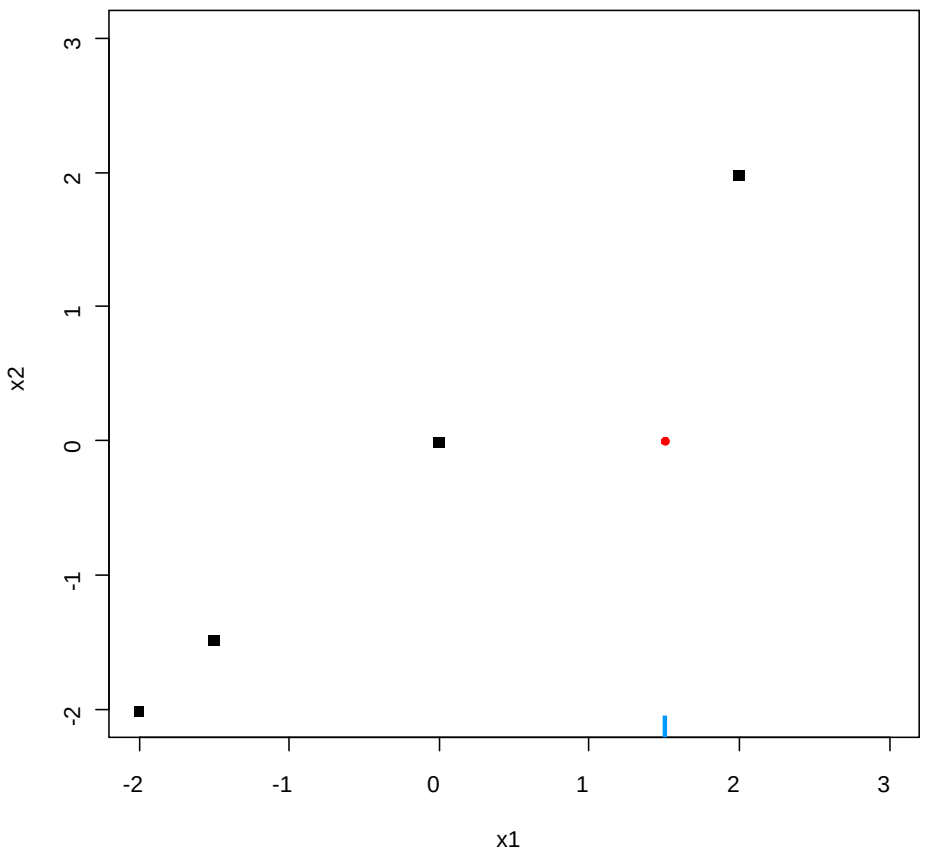


Algorithme : ACP itérative

Initialisation : imputation par la moyenne

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	0.00
2.0	1.98



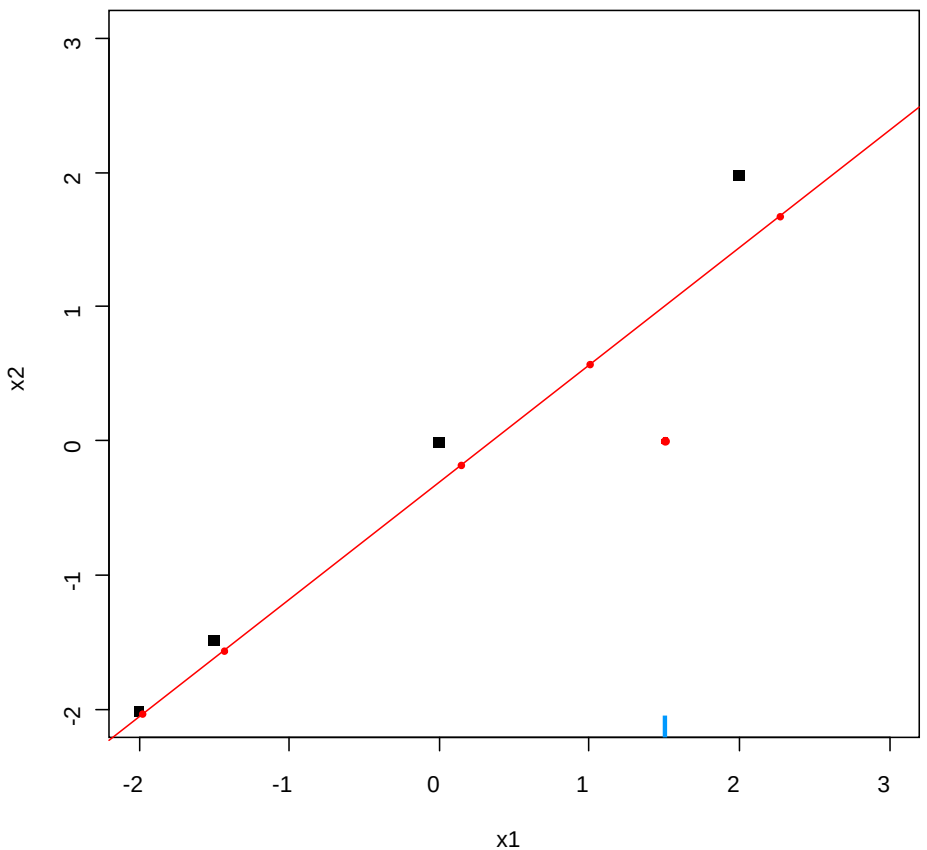
Algorithme : ACP itérative

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

$\widehat{x1}$	$\widehat{x2}$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	0.57
2.27	1.67

ACP sur le jeu complété à partir d'une dimension



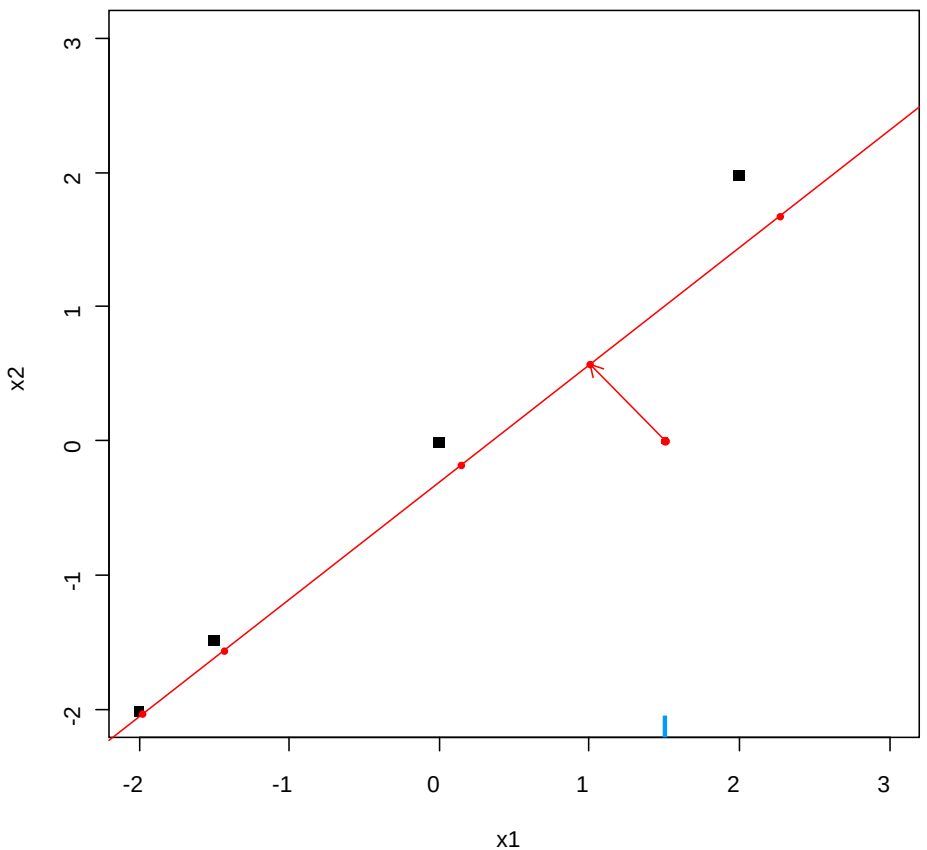
Algorithme : ACP itérative

Détermination de la valeur prédite par le modèle

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

$\hat{x1}$	$\hat{x2}$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	**0.57**
2.27	1.67



Algorithme : ACP itérative

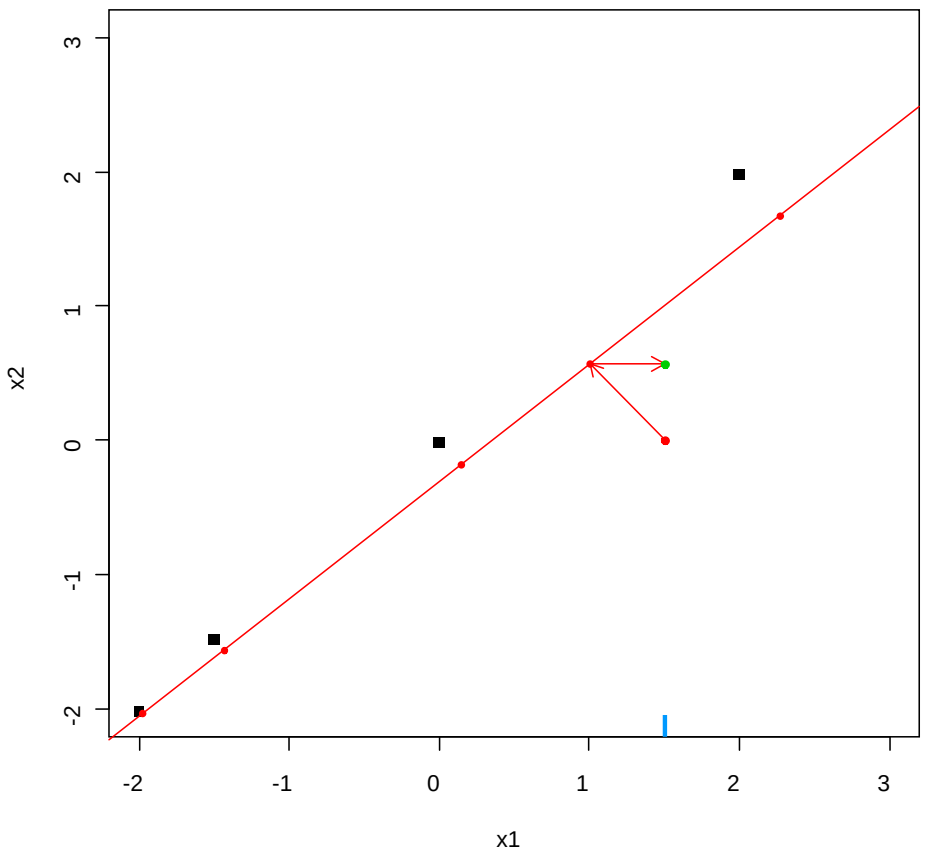
Etape d'imputation : $X^{\ell} = W * X + (1 - W) * \hat{X}^{\ell}$

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.00**
2.0	1.98

$\hat{x1}$	$\hat{x2}$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	**0.57**
2.27	1.67

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98



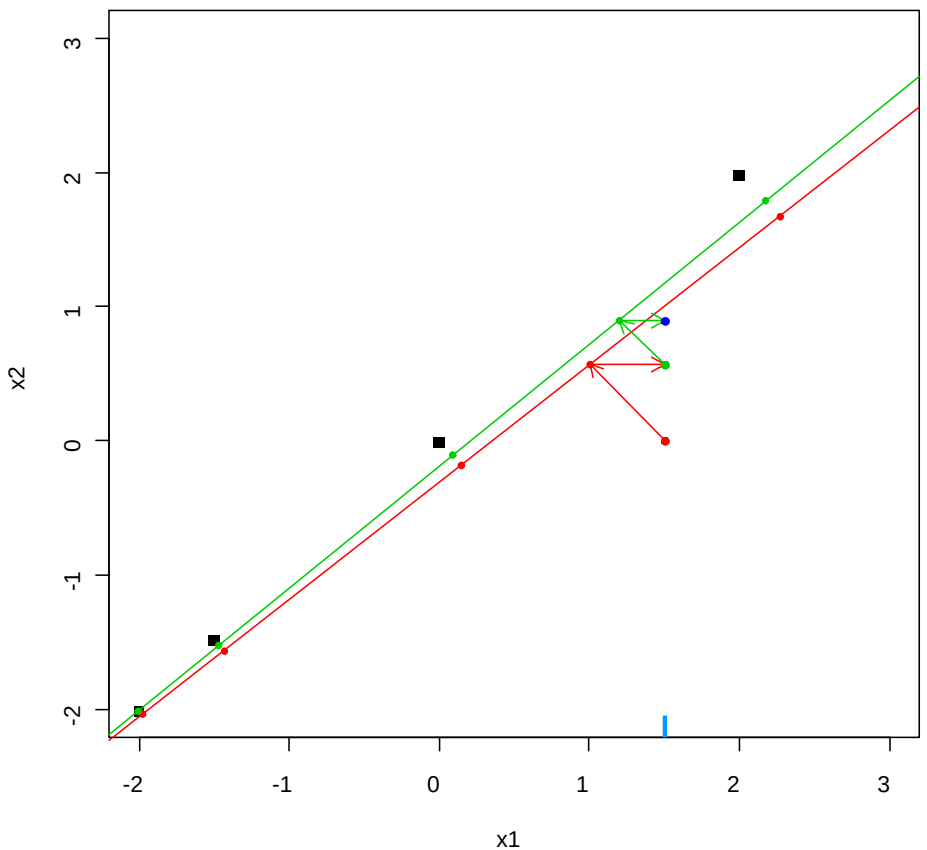
Algorithme : ACP itérative

ACP sur le jeu complété à partir d'une dimension

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98



Algorithme : ACP itérative

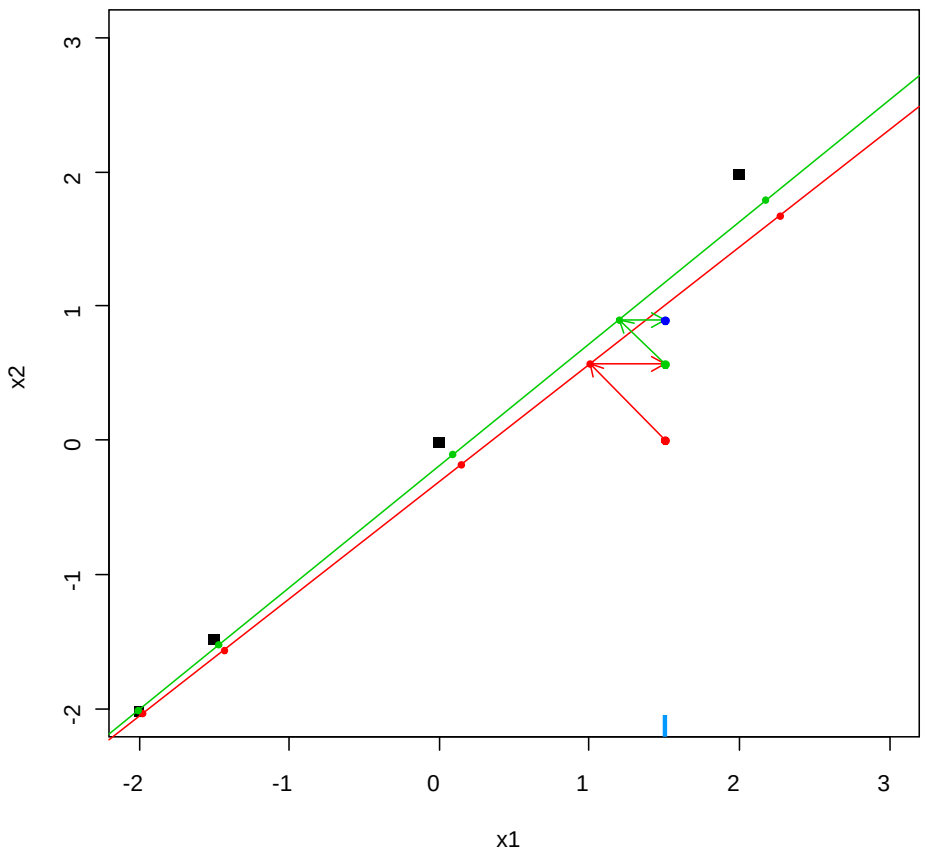
Etape d'imputation : $X^{\ell} = W * X + (1 - W) * \hat{X}^{\ell}$

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98

$\hat{x1}$	$\hat{x2}$
-2.00	-2.01
-1.47	-1.52
0.09	-0.11
1.20	**0.90**
2.18	1.78

x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.90**
2.0	1.98



Algorithme : ACP itérative

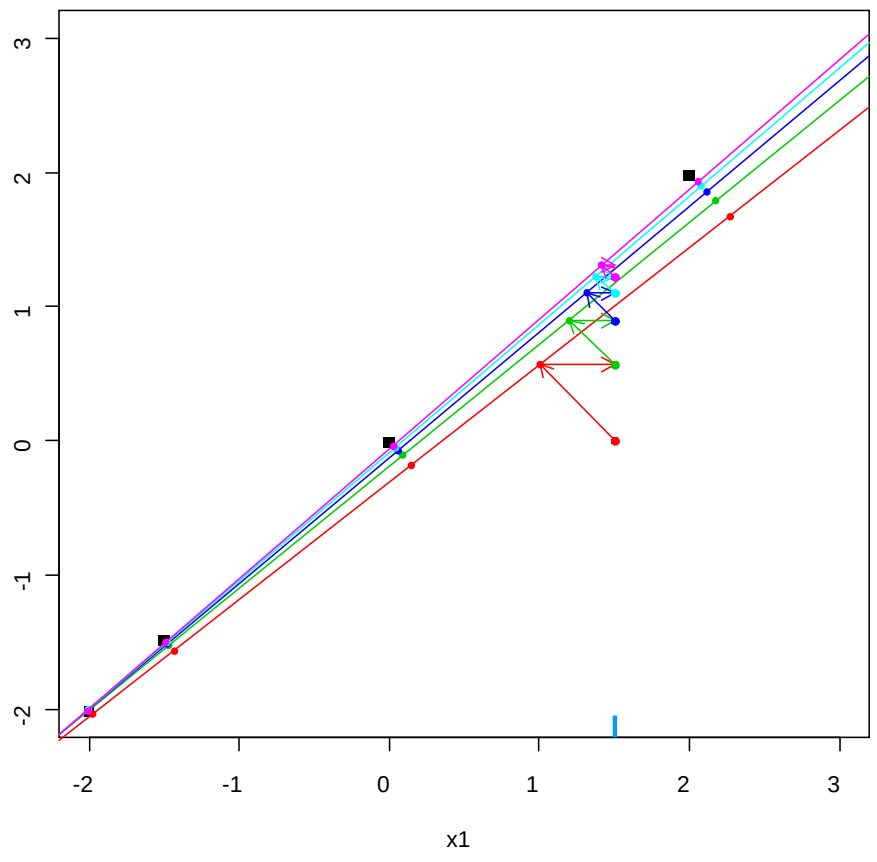
Processus itéré jusqu'à convergence

```
  x1    x2
-2.0 -2.01
-1.5 -1.48
 0.0 -0.01
 1.5    NA
 2.0  1.98

  x1    x2
-2.0 -2.01
-1.5 -1.48
 0.0 -0.01
 1.5  0.00
 2.0  1.98

   ^     ^
  x1    x2
-1.98 -2.04
-1.44 -1.56
 0.15 -0.18
 1.00  0.57
 2.27  1.67

  x1    x2
-2.0 -2.01
-1.5 -1.48
 0.0 -0.01
 1.5  0.57
 2.0  1.98
```



Algorithme : ACP itérative

1. initialisation $\ell = 0$: X^0 (imputation par la moyenne)
2. itération ℓ :
   (a) recherche de $\hat{F}^{\ell}_{I\times S}$ et $\hat{U}^{\ell}_{K\times S}$
   (b) imputation par $\hat{x}^{\ell}_{ik} = \sum_{s=1}^{S} \hat{f}^{\ell}_{is}\hat{u}^{\ell}_{ks} = \sum_{s=1}^{S} \sqrt{\hat{\lambda}_s}\hat{v}^{\ell}_{is}\hat{u}^{\ell}_{ks}$
3. (a) et (b) sont alternées jusqu'à convergence

Algorithme : ACP itérative régularisée

1. initialisation $\ell = 0$: X^0 (imputation par la moyenne)
2. itération ℓ :
   (a) recherche de $\hat{F}^{\ell}_{I\times S}$ et $\hat{U}^{\ell}_{K\times S}$
   (b) imputation par $\hat{x}^{\ell}_{ik} = \sum_{s=1}^{S} \left(\sqrt{\hat{\lambda}_s} - \frac{\hat{\sigma}^2}{\hat{\lambda}_s}\right) \hat{v}^{\ell}_{is} \hat{u}^{\ell}_{ks}$
3. (a) et (b) sont alternées jusqu'à convergence

Algorithme MIPCA

1. Initialisation
   - Estimation des paramètres du modèle d'ACP $X_{I\times K} = F_{I\times S} U^{\top}_{K\times S} + E_{I\times K}$ avec $E = (\epsilon_{ik})_{ik}$ et $\epsilon_{ik} \sim \mathcal{N}\left(0, \sigma^2\right) \Rightarrow \hat{F}, \hat{U}, \hat{\sigma}^2$

Algorithme MIPCA

1. Initialisation
   - Estimation des paramètres du modèle d'ACP
     $X_{I\times K} = F_{I\times S} U_{K\times S}^{\top} + E_{I\times K}$ avec $E = (\epsilon_{ik})_{ik}$ et
     $\epsilon_{ik} \sim \mathcal{N}\left(0, \sigma^2\right) \Rightarrow \hat{F}, \hat{U}, \hat{\sigma}^2$
2. Répéter M fois :
   - Propagation de l'incertitude sur $\hat{F}$ et $\hat{U}$
     - Création du tableau $X_m^{\star} = \hat{F}\hat{U}^{\top} + E_m^{\star}$ avec $E_m^{\star} \sim \mathcal{N}\left(0, \hat{\sigma}^2\right)$
     - ACP sur $X_m^{\star} \Rightarrow \hat{F}_m^{\star}, \hat{U}_m^{\star}, \hat{\sigma_m^{\star}}^2$

Algorithme MIPCA

1. Initialisation
   - Estimation des paramètres du modèle d'ACP $X_{I\times K} = F_{I\times S} U_{K\times S}^{\top} + E_{I\times K}$ avec $E = (\epsilon_{ik})_{ik}$ et $\epsilon_{ik} \sim \mathcal{N}(0, \sigma^2) \Rightarrow \hat{F}, \hat{U}, \hat{\sigma}^2$
2. Répéter M fois :
   - Propagation de l'incertitude sur $\hat{F}$ et $\hat{U}$
     - Création du tableau $X_m^\star = \hat{F}\hat{U}^{\top} + E_m^\star$ avec $E_m^\star \sim \mathcal{N}(0, \hat{\sigma}^2)$
     - ACP sur $X_m^\star \Rightarrow \hat{F}_m^\star, \hat{U}_m^\star, \hat{\sigma_m^\star}^2$
   - Respecter la distribution du jeu de données
     - Imputation simple : $X_m \leftarrow W * X + (1 - W) * \hat{X}_m^\star$
     - Ajout du résidu : $X_m \leftarrow X_m + \tilde{E}$ tiré selon $\mathcal{N}\left(0, \hat{\sigma^\star}^2\right)$

Algorithme Expectation Maximization Bootstrap

- Hypothèse : $X_{I\times K} \sim \mathcal{N}(\mu, \Sigma)$
- Algorithme :
  1. Bootstrap des individus
  2. Estimation de μ et Σ
  3. Imputation par régression aléatoire
- Implémenté dans le package `Amelia` du logiciel `R`

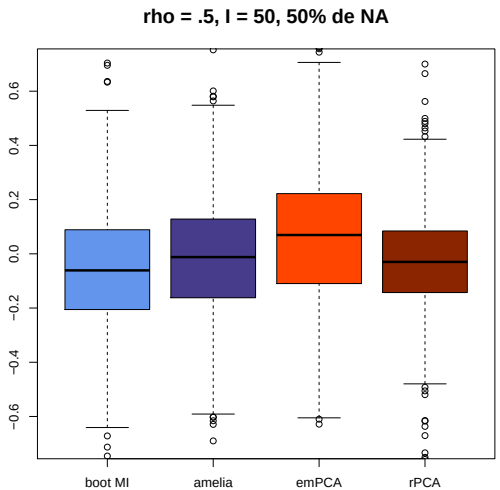
Imputation par équations enchaînées (MICE)

- Hypothèse sur les distributions conditionnelles
- Algorithme :
  1. Initialisation :
     - Spécifier un modèle pour chaque variable
     - Imputation des données manquantes par tirage dans les données observées
  2. Pour chaque variable
     - Estimation des paramètres du modèle
     - Imputation par régression aléatoire
  3. Répéter (2) un nombre de fois fixé à l'avance
- Implémenté dans le package `MICE` du logiciel `R`

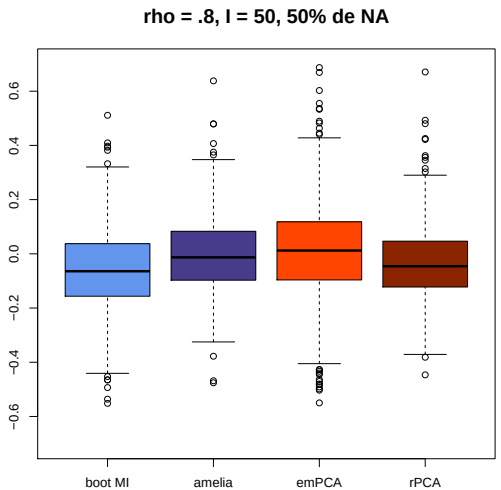
Protocole de simulation

- Paramètres d'intérêt : $\theta_1 = \beta_1$, $\theta_2 = \mathbb{E}[Y]$
- 1000 simulations
  - Simulation de X selon une loi normale pour 3 matrices de covariance différentes
  - Génération 50% de NA complètement au hasard
  - Imputation de $M = 20$ tableaux par EMB, MICE, MIPCA
- Evaluation par
  - le biais sur θ_i
  - le coverage

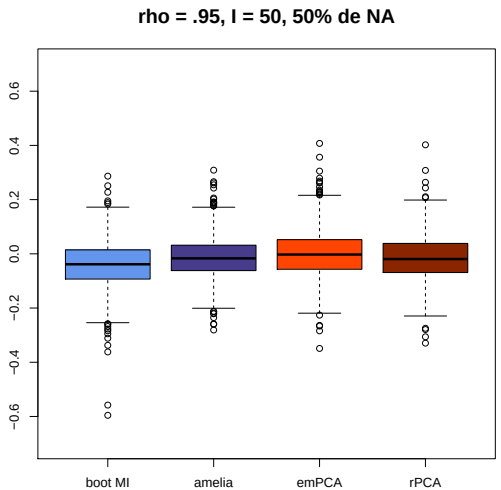
Résultats $\theta_1 = \beta_1$



collinearite faible

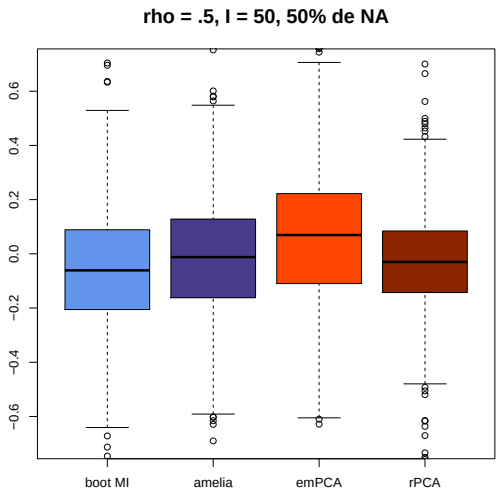


collinearite moyenne

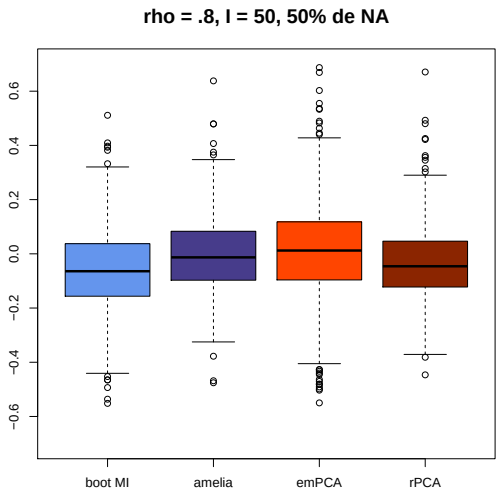


collinearite forte

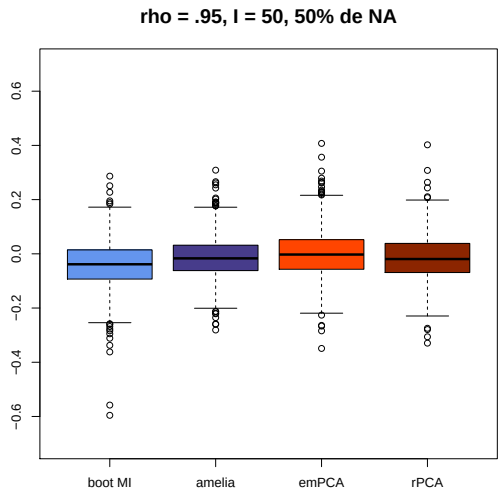
Résultats $\theta_1 = \beta_1$



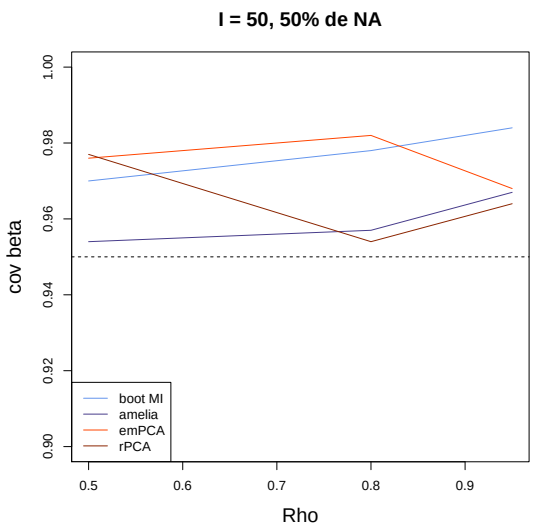
collinearite faible



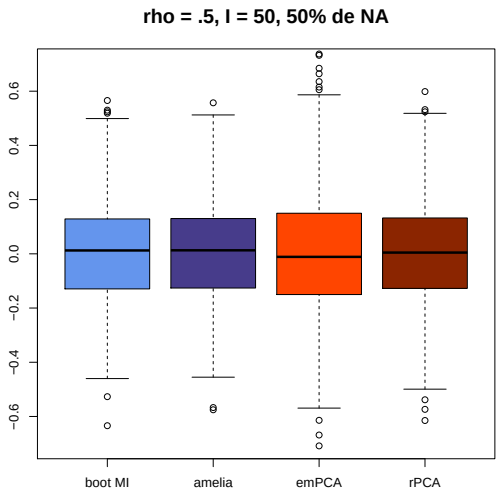
collinearite moyenne

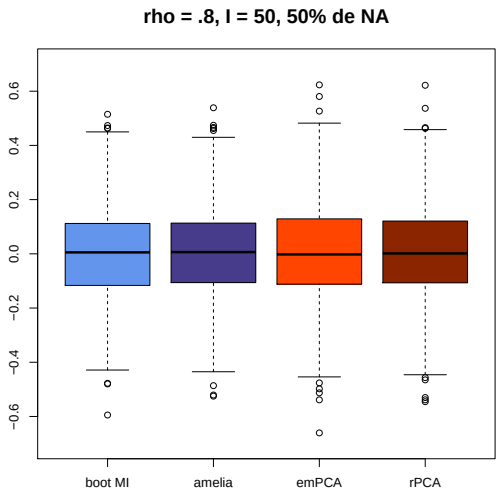


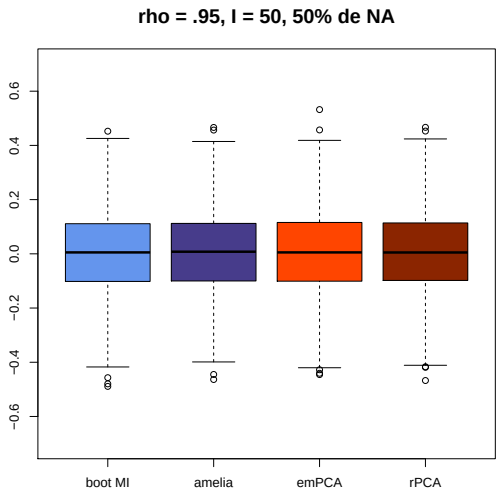
collinearite forte

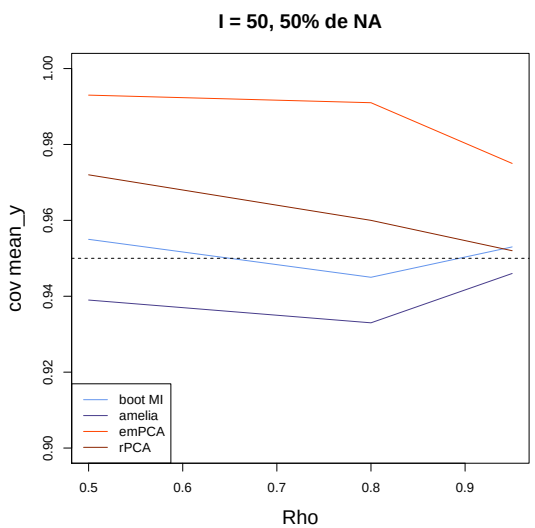


Résultats $\theta_2 = \mathbb{E}[Y]$









Conclusion

Une méthode nouvelle pour effectuer de l'imputation multiple qui

- permet d'imputer des jeux de données quantitatifs, y compris quand $I << K$.
- nécessite le réglage d'un paramètre (nombre de dimensions)
- est disponible dans le package `missMDA` du logiciel `R`
- peut s'étendre aux données mixtes