

# Gestion des données manquantes en/par ACP

---

François Husson

UP de mathématiques appliquées - l'institut Agro



Journées d'études en statistique – SFdS 2021

- ➊ Introduction
- ➋ ACP et reconstitution de données
- ➌ Algorithme d'ACP itérative
- ➍ Régularisation de l'ACP itérative
- ➎ Mise en œuvre pratique
- ➏ Conclusion



Gertrude Mary Cox

*“The best thing to do about missing values is not to have any”*

# Les données manquantes



Gertrude Mary Cox

*“The best thing to do about missing values is not to have any”*

## Est-ce un problème en big data ?



“One of the ironies of Big Data is that missing data play an ever more significant role” (R. Sameworth, 2019)

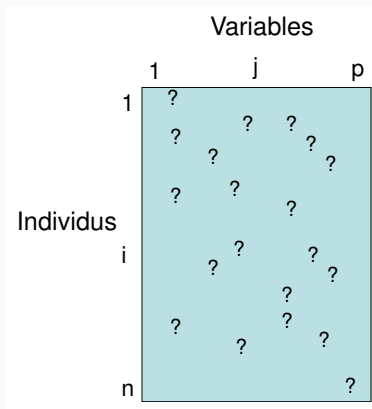
Une matrice  $n \times p$ , avec chaque cellule ayant une proba 0.01 d’être manquante

$p = 5 \Rightarrow \approx 95\%$  de lignes conservées

$p = 300 \Rightarrow \approx 5\%$  de lignes conservées

# Objectifs

- Faire une ACP sur un tableau incomplet
- Utiliser l'ACP comme alternative aux méthodes d'imputation simple ?
  - Modèle joint (**norm**) ou modèle conditionnel (**mice**)
  - k-plus proches voisins (**class**, **FNN**)
  - forêts aléatoires (**missForest**)
  - ...



## Exemple sur des données ozone

Code disponible : <http://factominer.free.fr/missMDA/ozone.R>

```
> don <- read.table("http://factominer.free.fr/missMDA/ozoneNA.csv",  
  header=TRUE,sep="," ,row.names=1)
```

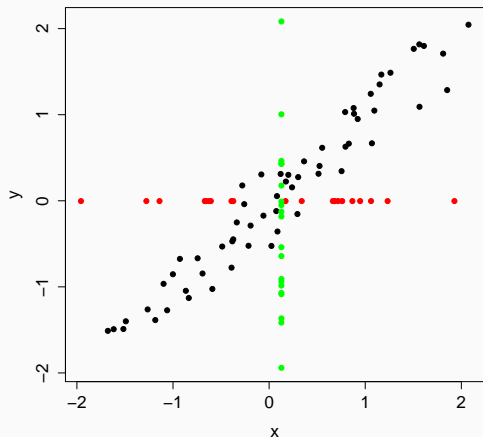
	O3	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	O3v
0601	82	15.6	18.5	NA	4	4	8	NA	-1.7101	-0.6946	84
0602	82	NA	NA	NA	5	5	7	NA	NA	NA	87
0603	92	NA	17.6	19.5	2	5	4	2.9544	1.8794	0.5209	82
0604	114	16.2	NA	NA	1	1	0	NA	NA	NA	92
0605	94	17.4	20.5	NA	8	8	7	-0.5	NA	-4.3301	114
0606	80	17.7	NA	18.3	NA	NA	NA	-5.6382	-5	-6	94
0607	NA	16.8	15.6	14.9	7	8	8	-4.3301	-1.8794	-3.7588	80
0610	79	14.9	17.5	18.9	5	5	4	0	-1.0419	-1.3892	NA
0611	101	NA	19.6	21.4	2	4	4	-0.766	NA	-2.2981	79
0612	NA	18.3	21.9	22.9	5	6	8	1.2856	-2.2981	-3.9392	101
0613	101	17.3	19.3	20.2	NA	NA	NA	-1.5	-1.5	-0.8682	NA
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.
0927	NA	16.2	20.8	22.1	6	5	5	-0.6946	-2	-1.3681	71
0928	99	16.9	23	22.6	NA	4	7	1.5	0.8682	0.8682	NA
0929	NA	16.9	19.8	22.1	6	5	3	-4	-3.7588	-4	99
0930	70	15.7	18.6	20.7	NA	NA	NA	0	-1.0419	-4	NA

## De (mauvaises) solutions faciles à mettre en œuvre

- Suppression des données manquantes : rarement intéressant ... mais souvent utilisée (fonction `lm` de R)

## De (mauvaises) solutions faciles à mettre en œuvre

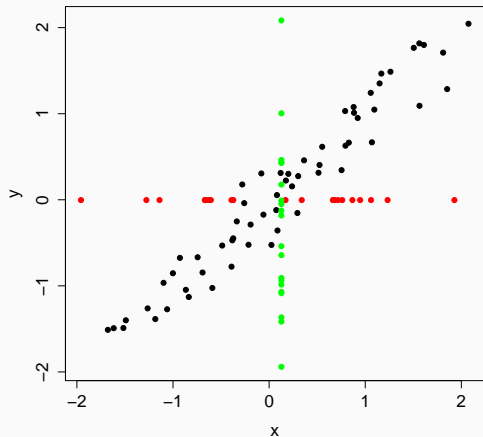
- Suppression des données manquantes : rarement intéressant ... mais souvent utilisée (fonction `lm` de R)
- Imputation par la moyenne (option par défaut dans de nombreux logiciels)





## De (mauvaises) solutions faciles à mettre en œuvre

- Suppression des données manquantes : rarement intéressant ... mais souvent utilisée (fonction `lm` de R)
- Imputation par la moyenne (option par défaut dans de nombreux logiciels)



Distorsion très importante des liaisons  
entre variables

# Etude du dispositif de données manquantes

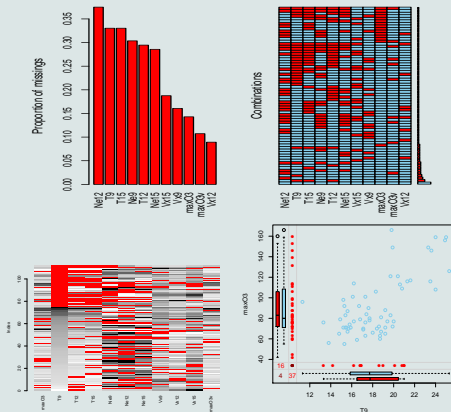
## Hypothèse

On suppose que le mécanisme conduisant à l'apparition de données manquantes est MCAR ou MAR

## Visualisation des données manquantes

```
> library(VIM)
> aggr(don, only.miss=TRUE, sortVar=TRUE)
```

```
> matrixplot(don, sortby=2)
> marginplot(don[,c("T9", "maxO3")])
```



- ① Introduction
- ② ACP et reconstitution de données
- ③ Algorithme d'ACP itérative
- ④ Régularisation de l'ACP itérative
- ⑤ Mise en œuvre pratique
- ⑥ Conclusion

## Ajustement du nuage en ACP

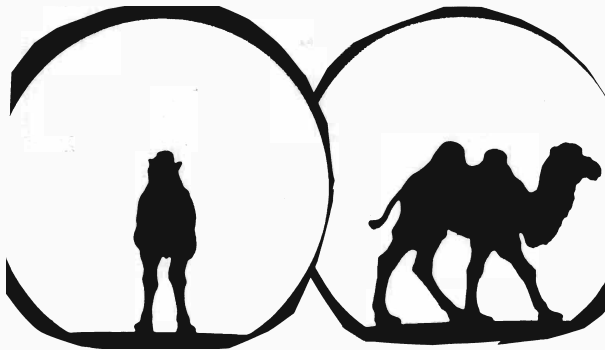
L'ACP vise à trouver le sous-espace qui fournit la meilleure représentation des données

L'ACP vise à trouver le sous-espace qui fournit la meilleure représentation des données



**Figure 1** – Chameau ou dromadaire ? source J.P. Fenelon

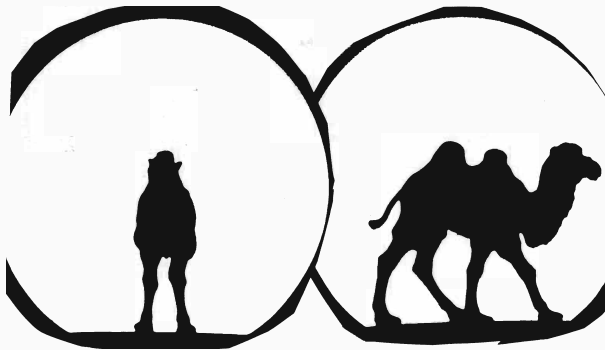
L'ACP vise à trouver le sous-espace qui fournit la meilleure représentation des données



**Figure 1** – Chameau ou dromadaire ? source J.P. Fenelon

# Ajustement du nuage en ACP

L'ACP vise à trouver le sous-espace qui fournit la meilleure représentation des données



**Figure 1** – Chameau ou dromadaire ? source J.P. Fenelon

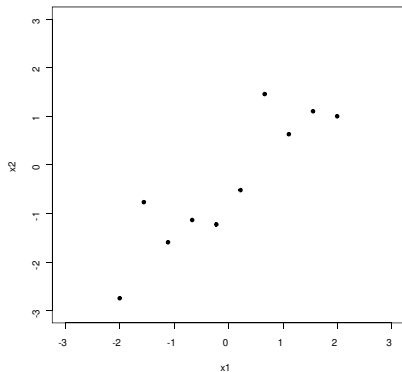
⇒ Meilleure approximation par projection

⇒ Meilleure représentation de la diversité, de la variabilité

# Ajustement du nuage en ACP

**X**

-2.00	-2.36
-1.56	-0.39
-1.11	-1.21
-0.67	-0.75
-0.22	-0.84
0.22	-0.14
0.67	1.84
1.11	1.01
1.56	1.48
2.00	1.38



**X** : données en 2 dimensions



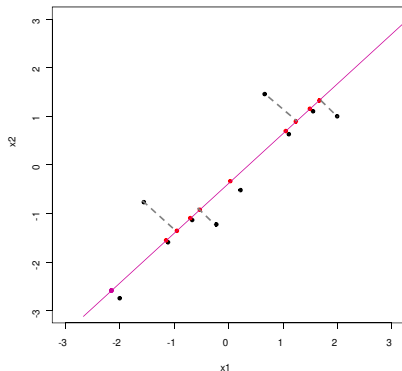
# Ajustement du nuage en ACP

**X**

-2.00	-2.36
-1.56	-0.39
-1.11	-1.21
-0.67	-0.75
-0.22	-0.84
0.22	-0.14
0.67	1.84
1.11	1.01
1.56	1.48
2.00	1.38

-2.16	-2.21
-0.96	-0.98
-1.15	-1.17
-0.70	-0.72
-0.53	-0.54
0.04	0.04
1.25	1.27
1.05	1.07
1.50	1.54
1.67	1.70

**$\hat{X}$**



**X** : données en 2 dimensions

Minimisation de la distance  
entre les individus et leur  
projection

# Reconstitution en ACP

**X**

-2.00	-2.36
-1.56	-0.39
-1.11	-1.21
-0.67	-0.75
-0.22	-0.84
0.22	-0.14
0.67	1.84
1.11	1.01
1.56	1.48
2.00	1.38

**D**

5.60

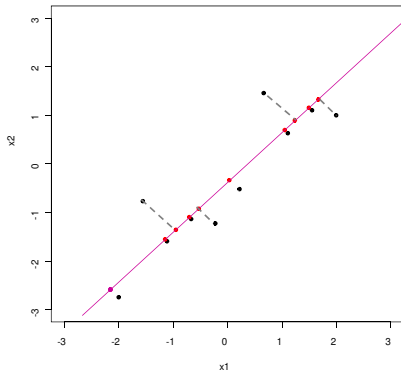
**V'**

0.699 0.715

**U**

-0.55  
-0.24  
-0.29  
-0.18  
-0.13  
0.01  
0.32  
0.27  
0.38  
0.43

-2.16 -2.21  
-0.96 -0.98  
-1.15 -1.17  
-0.70 -0.72  
-0.53 -0.54  
0.04 0.04  
1.25 1.27  
1.05 1.07  
1.50 1.54  
1.67 1.70



$$\hat{X} = U D V'$$

$\hat{X} = M + U D V'$  (produit matriciel utilisant les coordonnées des individus et les coordonnées des variables issues de l'ACP)

## ACP : cas complet

⇒ Point de vue géométrique : minimiser l'erreur de reconstitution

⇒ Approximation de  $\mathbf{X}$  par une matrice de rang  $S < p$  :

$$\|\mathbf{X}_{n \times p} - \hat{\mathbf{X}}_{n \times p}\|^2 \quad \text{SVD : } \hat{\mathbf{X}}^{\text{ACP}} = \mathbf{M}_{n \times p} + \mathbf{U}_{n \times S} \mathbf{D}_{S \times S} \mathbf{V}'_{p \times S}$$

$\mathbf{F} = \mathbf{U}\mathbf{D}$  composantes principales (scores)

$\mathbf{V}$  axes principaux (loadings)

## ACP : cas complet

⇒ Point de vue géométrique : minimiser l'erreur de reconstitution

⇒ Approximation de  $\mathbf{X}$  par une matrice de rang  $S < p$  :

$$\|\mathbf{X}_{n \times p} - \hat{\mathbf{X}}_{n \times p}\|^2 \quad \text{SVD : } \hat{\mathbf{X}}^{\text{ACP}} = \mathbf{M}_{n \times p} + \mathbf{U}_{n \times S} \mathbf{D}_{S \times S} \mathbf{V}_{p \times S}'$$

$\mathbf{F} = \mathbf{U}\mathbf{D}$  composantes principales (scores)

$\mathbf{V}$  axes principaux (loadings)

⇒ Point de vue modèle à effets fixes (Caussinus, 1986)

$$\mathbf{X}_{n \times p} = \tilde{\mathbf{X}}_{n \times p} + \varepsilon_{n \times p}$$

$$x_{ij} = m_j + \sum_{s=1}^S d_s u_{is} v_{js} + \varepsilon_{ij} \quad \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

Estimateurs de maximum de vraisemblance = estimateurs des moindres carrés

- ① Introduction
- ② ACP et reconstitution de données
- ③ Algorithme d'ACP itérative**
- ④ Régularisation de l'ACP itérative
- ⑤ Mise en œuvre pratique
- ⑥ Conclusion

⇒ ACP : moindres carrés

$$\left\| \mathbf{X}_{n \times p} - \left( \mathbf{M}_{n \times p} + \mathbf{U}_{n \times S} \mathbf{D}_{S \times S} \mathbf{V}_{p \times S}' \right) \right\|^2$$

⇒ ACP : moindres carrés

$$\left\| \mathbf{X}_{n \times p} - \left( \mathbf{M}_{n \times p} + \mathbf{U}_{n \times S} \mathbf{D}_{S \times S} \mathbf{V}_{p \times S}' \right) \right\|^2$$

⇒ ACP avec données manquantes : moindres carrés pondérés

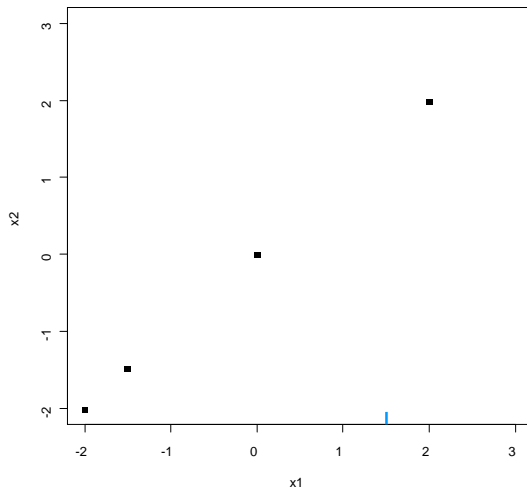
$$\left\| \mathbf{R}_{n \times p} * \left( \mathbf{X}_{n \times p} - \left( \mathbf{M}_{n \times p} + \mathbf{U}_{n \times S} \mathbf{D}_{S \times S} \mathbf{V}_{p \times S}' \right) \right) \right\|^2$$

with  $r_{ij} = 0$  si  $x_{ij}$  manquant,  $r_{ij} = 1$  sinon

Beaucoup d'algorithmes : moindres carrés pondérés alterné (Gabriel & Zamir, 1979) ; ACP iterative (Kiers, 1997)

# ACP itérative

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

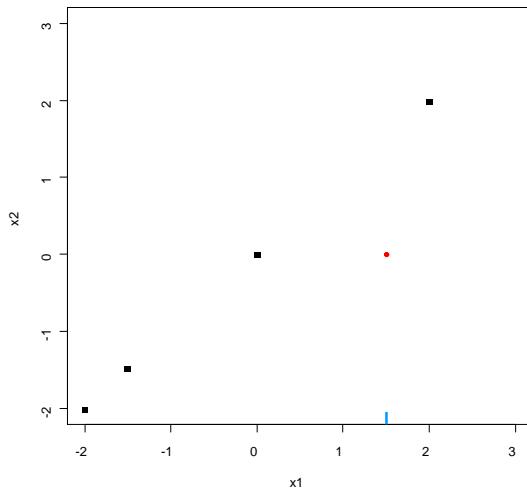




# ACP itérative

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



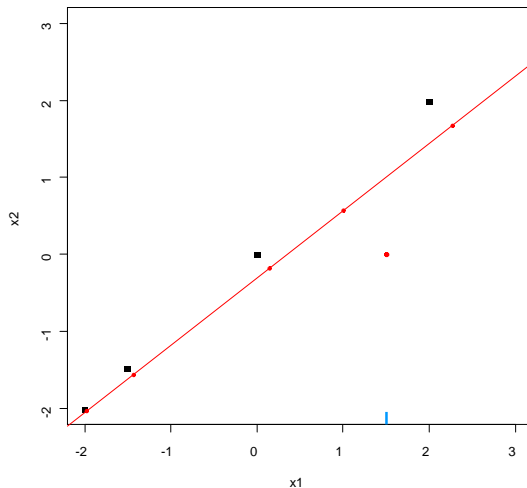
Initialisation  $\ell = 0$  :  $\mathbf{X}^0$  (imputation par la moyenne)

# ACP itérative

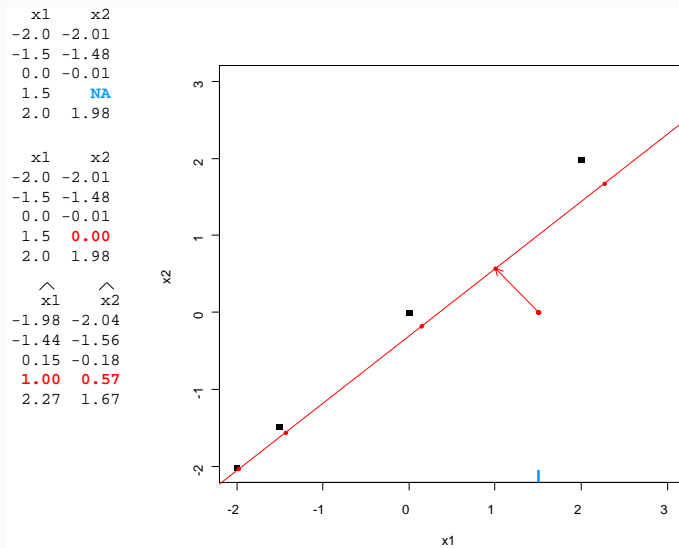
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98

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

$\hat{x}_1$	$\hat{x}_2$
-1.98	-2.04
-1.44	-1.56
0.15	-0.18
1.00	0.57
2.27	1.67



ACP sur le jeu de données complété  $\rightarrow (\mathbf{U}^\ell, \mathbf{D}^\ell, \mathbf{V}^\ell)$ ;



Valeurs manquantes imputées par le modèle  $\hat{\mathbf{X}}^\ell = \mathbf{M}^\ell + \mathbf{U}^\ell \mathbf{D}^\ell \mathbf{V}^{\ell T}$

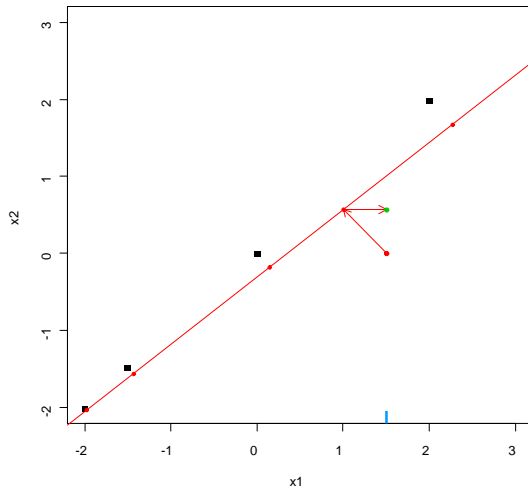
# ACP itérative

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

$\hat{x}_1$	$\hat{x}_2$
-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



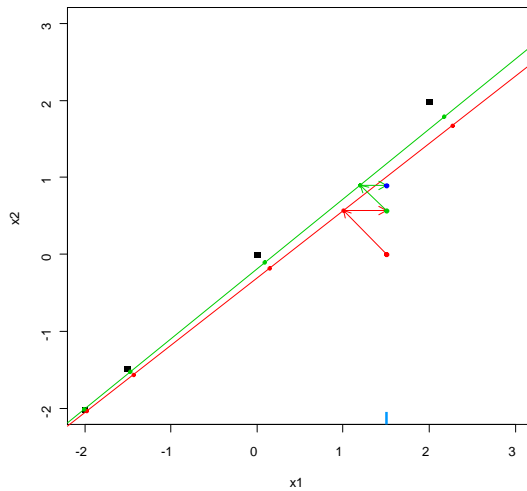
Nouveau jeu de données imputé  $\mathbf{X}^\ell = \mathbf{R} * \mathbf{X} + (1 - \mathbf{R}) * \hat{\mathbf{X}}^\ell$

# 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	0.57
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



# ACP itérative

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

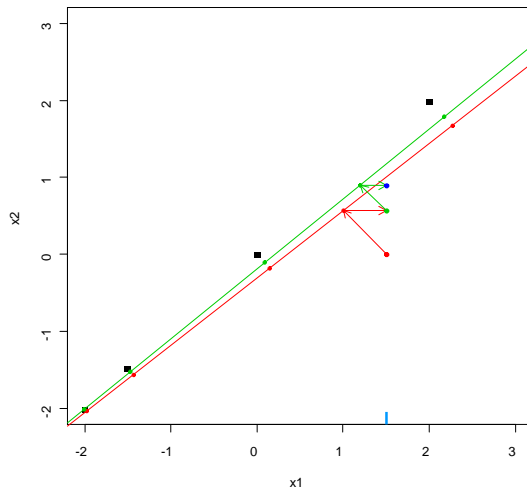
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	0.57
2.0	1.98

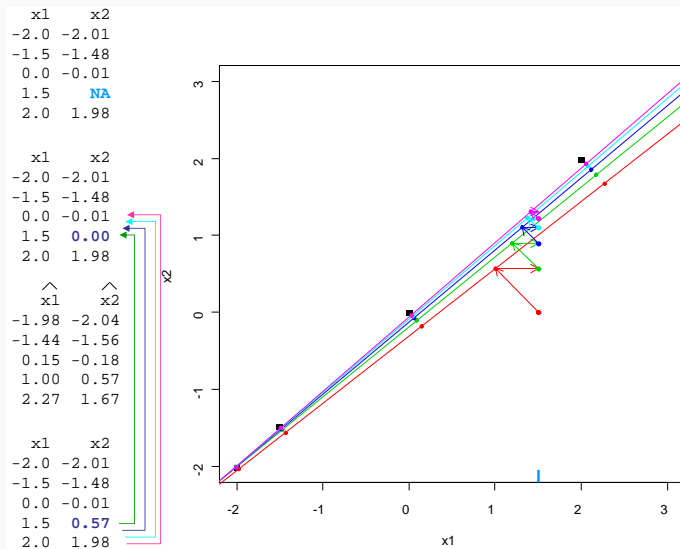
$\hat{x}_1$	$\hat{x}_2$
-2.00	-2.01
-1.47	-1.52
0.09	-0.11
1.20	0.90
2.18	1.78

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



# ACP itérative

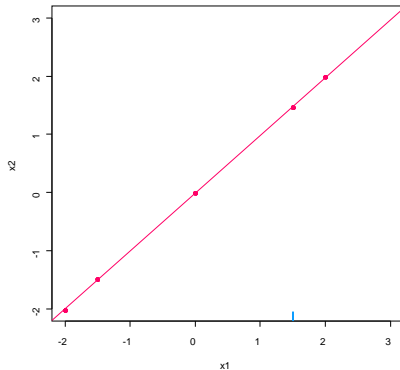


Les étapes sont répétées jusqu'à convergence

# 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	1.46
2.0	1.98



ACP sur le jeu de données complété  $\rightarrow (\mathbf{U}^\ell, \mathbf{D}^\ell, \mathbf{V}^\ell)$

Valeurs manquantes imputées par le modèle  $\hat{\mathbf{X}}^\ell = \mathbf{M}^\ell + \mathbf{U}^\ell \mathbf{D}^\ell \mathbf{V}^{\ell'}$



- ① initialisation  $\ell = 0$  :  $\mathbf{X}^0$  (imputation par la moyenne)
- ② step  $\ell$  :
  - (a) ACP sur le tableau complété  $\rightarrow (\mathbf{U}^\ell, \mathbf{D}^\ell, \mathbf{V}^\ell)$ ;  
 $S$  dimensions conservées
  - (b) valeurs manquantes imputées par  $\hat{\mathbf{X}}^\ell = \mathbf{M}^\ell + \mathbf{U}^\ell \mathbf{D}^\ell \mathbf{V}^{\ell'}$ ;  
nouveau tableau imputé  $\mathbf{X}^\ell = \mathbf{R} * \mathbf{X} + (1 - \mathbf{R}) * \hat{\mathbf{X}}^\ell$
- ③ étapes répétées jusqu'à convergence

- ① initialisation  $\ell = 0$  :  $\mathbf{X}^0$  (imputation par la moyenne)
- ② step  $\ell$  :
  - (a) ACP sur le tableau complété  $\rightarrow (\mathbf{U}^\ell, \mathbf{D}^\ell, \mathbf{V}^\ell)$ ;  
 $S$  dimensions conservées
  - (b) valeurs manquantes imputées par  $\hat{\mathbf{X}}^\ell = \mathbf{M}^\ell + \mathbf{U}^\ell \mathbf{D}^\ell \mathbf{V}^{\ell'}$ ;  
nouveau tableau imputé  $\mathbf{X}^\ell = \mathbf{R} * \mathbf{X} + (1 - \mathbf{R}) * \hat{\mathbf{X}}^\ell$
  - (c) moyennes (et écarts-types) sont mis à jour
- ③ étapes répétées jusqu'à convergence

- ① initialisation  $\ell = 0$  :  $\mathbf{X}^0$  (imputation par la moyenne)
- ② step  $\ell$  :
  - (a) ACP sur le tableau complété  $\rightarrow (\mathbf{U}^\ell, \mathbf{D}^\ell, \mathbf{V}^\ell)$ ;  
*S dimensions conservées*
  - (b) valeurs manquantes imputées par  $\hat{\mathbf{X}}^\ell = \mathbf{M}^\ell + \mathbf{U}^\ell \mathbf{D}^\ell \mathbf{V}^{\ell'}$ ;  
nouveau tableau imputé  $\mathbf{X}^\ell = \mathbf{R} * \mathbf{X} + (1 - \mathbf{R}) * \hat{\mathbf{X}}^\ell$
  - (c) moyennes (et écarts-types) sont mis à jour
- ③ étapes répétées jusqu'à convergence

- ① initialisation  $\ell = 0$  :  $\mathbf{X}^0$  (imputation par la moyenne)
- ② step  $\ell$  :
  - (a) ACP sur le tableau complété  $\rightarrow (\mathbf{U}^\ell, \mathbf{D}^\ell, \mathbf{V}^\ell)$ ;  
**S dimensions conservées**
  - (b) valeurs manquantes imputées par  $\hat{\mathbf{X}}^\ell = \mathbf{M}^\ell + \mathbf{U}^\ell \mathbf{D}^\ell \mathbf{V}^{\ell'}$ ;  
nouveau tableau imputé  $\mathbf{X}^\ell = \mathbf{R} * \mathbf{X} + (1 - \mathbf{R}) * \hat{\mathbf{X}}^\ell$
  - (c) moyennes (et écarts-types) sont mis à jour
- ③ étapes répétées jusqu'à convergence

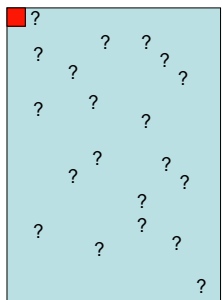
$\Rightarrow$  algorithme EM pour le modèle à effets fixes

$\Rightarrow$  Imputation (complétion de matrice, Netflix)

$\Rightarrow$  Réduction de la variabilité (imputation par  $\mathbf{M} + \mathbf{UDV}'$ )



## Choix du nombre de composantes

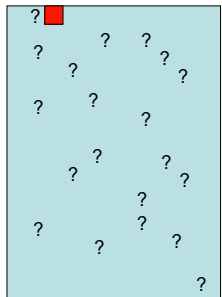


⇒ EM-CV (Bro *et al.* 2008)

$$\text{MSEP}(s) = \frac{1}{np} \sum_{i=1}^n \sum_{j=1}^p (x_{ij} - \hat{x}_{ij}^{s; -\{ij\}})^2$$

⇒ Très coûteux en temps de calcul

## Choix du nombre de composantes



⇒ EM-CV (Bro *et al.* 2008)

$$\text{MSEP}(s) = \frac{1}{np} \sum_{i=1}^n \sum_{j=1}^p (x_{ij} - \hat{x}_{ij}^{s; -\{ij\}})^2$$

⇒ Très coûteux en temps de calcul

## Choix du nombre de composantes



⇒ EM-CV (Bro *et al.* 2008)

$$\text{MSEP}(s) = \frac{1}{np} \sum_{i=1}^n \sum_{j=1}^p (x_{ij} - \hat{x}_{ij}^{s; -\{ij\}})^2$$

⇒ Très coûteux en temps de calcul



## Choix du nombre de composantes



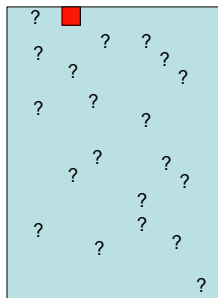
⇒ EM-CV (Bro *et al.* 2008)

$$\text{MSEP}(s) = \frac{1}{np} \sum_{i=1}^n \sum_{j=1}^p (x_{ij} - \hat{x}_{ij}^{s; -\{ij\}})^2$$

⇒ Très coûteux en temps de calcul

Ajouter plusieurs valeurs manquantes supplémentaires simultanément

## Choix du nombre de composantes



⇒ EM-CV (Bro *et al.* 2008)

$$\text{MSEP}(s) = \frac{1}{np} \sum_{i=1}^n \sum_{j=1}^p (x_{ij} - \hat{x}_{ij}^{s; -\{ij\}})^2$$

⇒ Très coûteux en temps de calcul

Ajouter plusieurs valeurs manquantes supplémentaires simultanément

Approximation possible par validation croisée généralisée  $\implies$  gain en temps de calcul

- Résultats de l'ACP obtenus à partir des données observées uniquement : graphe des individus et graphe des variables

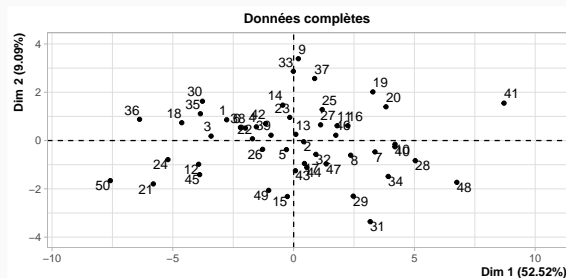
⇒ On "saute" les données manquantes, l'ACP itérative minimise

$$\| \mathbf{R} * (\mathbf{X} - (\mathbf{M} + \mathbf{UDV}')) \|^2$$

- Imputation :
  - prend en compte les ressemblances entre individus et les liaisons entre variables
  - le tableau imputé peut être utilisé (avec précaution) pour réaliser d'autres analyses
- Problème de surajustement

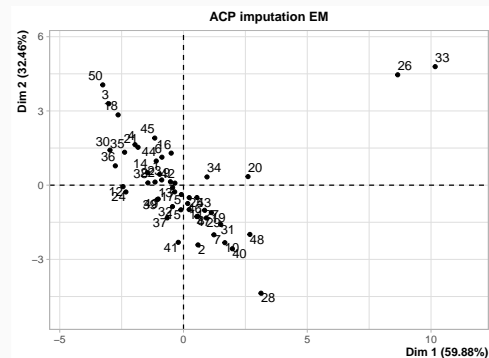
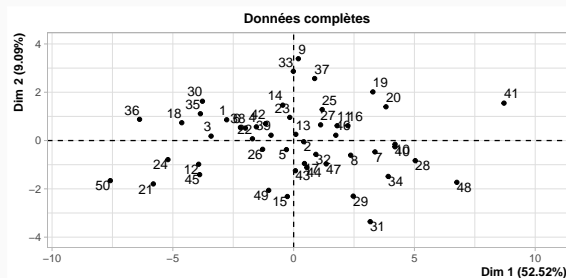
- ① Introduction
- ② ACP et reconstitution de données
- ③ Algorithme d'ACP itérative
- ④ Régularisation de l'ACP itérative**
- ⑤ Mise en œuvre pratique
- ⑥ Conclusion

$$X_{50 \times 10} = \mathbf{U}_{50 \times 2} \mathbf{D}^{1/2} \mathbf{V}'_{10 \times 2} + \mathcal{N}(0, 0.5);$$



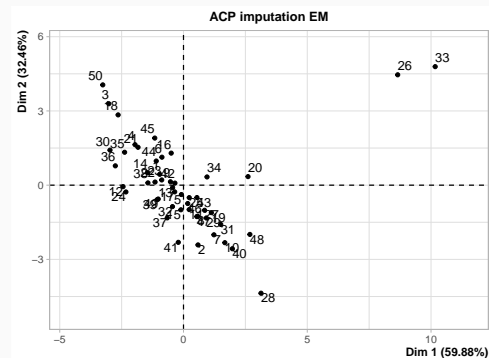
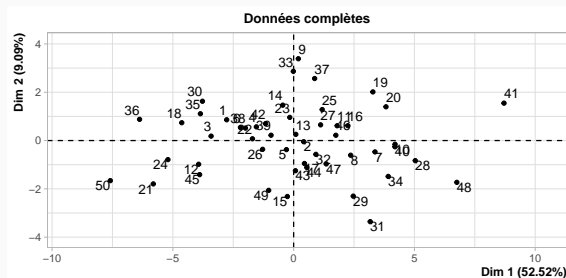
# Surajustement

$$X_{50 \times 10} = \mathbf{U}_{50 \times 2} \mathbf{D}^{1/2} \mathbf{V}'_{10 \times 2} + \mathcal{N}(0, 0.5); \text{ 50\% of NA}$$



# Surajustement

$$\mathbf{X}_{50 \times 10} = \mathbf{U}_{50 \times 2} \mathbf{D}^{1/2} \mathbf{V}'_{10 \times 2} + \mathcal{N}(0, 0.5); \text{ 50\% of NA}$$



⇒ erreur d'ajustement faible :  $\|\mathbf{R} * (\mathbf{X} - \hat{\mathbf{X}})\|^2 = 0.50$

⇒ erreur de prédiction élevée :  $\|(1 - \mathbf{R}) * (\mathbf{X} - \hat{\mathbf{X}})\|^2 = 16.98$

⇒ Bon ajustement et mauvaise prédiction

- Trop de paramètres sont estimés par rapport au nombre de données observées : le nombre de dimension  $S$  et le nombre de données manquantes sont grands
  - Faibles liaisons entre variables
- ① Diminuer le nombre  $S$
  - ② Early stopping
  - ③ Régularisation ⇒ ACP itérative régularisée



## ACP itérative régularisée (Josse *et al.*, 2009)

⇒ Initialisation - étape d'estimation - étape d'imputation

L'étape d'imputation :

$$\hat{x}_{ij}^{\text{ACP}} = \sum_{s=1}^S d_s u_{is} v_{js}$$

est remplacée par une étape d'imputation régularisée :

$$\hat{x}_{ij}^{\text{rACP}} = \sum_{s=1}^S \left( \frac{d_s^2 - \hat{\sigma}^2}{d_s^2} \right) d_s u_{is} v_{js} = \sum_{s=1}^S \left( d_s - \frac{\hat{\sigma}^2}{d_s} \right) u_{is} v_{js}$$

## ACP itérative régularisée (Josse *et al.*, 2009)

⇒ Initialisation - étape d'estimation - étape d'imputation

L'étape d'imputation :

$$\hat{x}_{ij}^{\text{ACP}} = \sum_{s=1}^S d_s u_{is} v_{js}$$

est remplacée par une étape d'imputation régularisée :

$$\hat{x}_{ij}^{\text{rACP}} = \sum_{s=1}^S \left( \frac{d_s^2 - \hat{\sigma}^2}{d_s^2} \right) d_s u_{is} v_{js} = \sum_{s=1}^S \left( d_s - \frac{\hat{\sigma}^2}{d_s} \right) u_{is} v_{js}$$

$$\hat{\sigma}^2 = \frac{RSS}{ddl} = \frac{n \sum_{s=S+1}^p d_s^2}{(n-1-S)(p-S)}$$

# ACP itérative régularisée (Josse *et al.*, 2009)

⇒ Initialisation - étape d'estimation - étape d'imputation

L'étape d'imputation :

$$\hat{x}_{ij}^{\text{ACP}} = \sum_{s=1}^S d_s u_{is} v_{js}$$

est remplacée par une étape d'imputation régularisée :

$$\hat{x}_{ij}^{\text{rACP}} = \sum_{s=1}^S \left( \frac{d_s^2 - \hat{\sigma}^2}{d_s^2} \right) d_s u_{is} v_{js} = \sum_{s=1}^S \left( d_s - \frac{\hat{\sigma}^2}{d_s} \right) u_{is} v_{js}$$

$$\hat{\sigma}^2 = \frac{RSS}{ddl} = \frac{n \sum_{s=S+1}^p d_s^2}{(n-1-S)(p-S)}$$

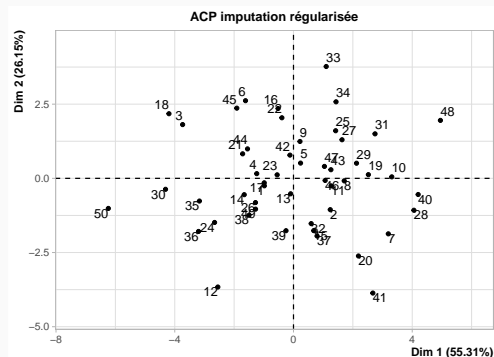
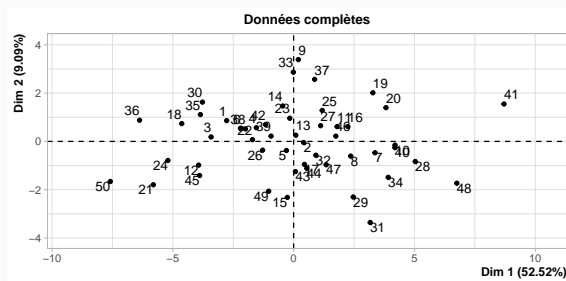
Compromis seuillage doux/dur (Mazumder, Hastie & Tibshirani, 2010)

$\sigma^2$  petit → ACP régularisée  $\approx$  ACP

$\sigma^2$  grand → imputation par la moyenne

# Surajustement

$$\mathbf{X}_{50 \times 10} = \mathbf{U}_{50 \times 2} \mathbf{D}^{1/2} \mathbf{V}'_{10 \times 2} + \mathcal{N}(0, 0.5); \text{ 50\% of NA}$$



⇒ erreur d'ajustement :  $\|\mathbf{R} * (\mathbf{X} - \hat{\mathbf{X}})\|^2 = 0.56$  (EM= 0.50)

⇒ erreur de prédiction :  $\|(1 - \mathbf{R}) * (\mathbf{X} - \hat{\mathbf{X}})\|^2 = 2.28$  (EM= 16.98)

- Bonne qualité d'imputation quand la structure dans le jeu de données est forte (imputation utilisant les ressemblances entre individus et les liaisons entre variables)
- Bien meilleur que l'algorithme Nipals (encore trop utilisé)
- Compétitif par rapport aux forêts aléatoires

- ① Introduction
- ② ACP et reconstitution de données
- ③ Algorithme d'ACP itérative
- ④ Régularisation de l'ACP itérative
- ⑤ Mise en œuvre pratique**
- ⑥ Conclusion

# Imputation par ACP en pratique

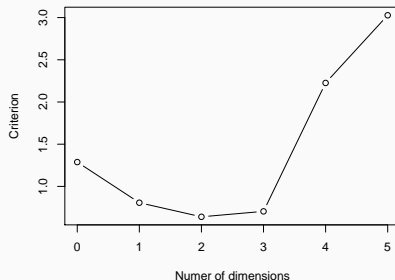
## Tutoriel sur l'ACP avec données manquantes

(données ozone, lignes de code)

⇒ Etape 1 : Estimation du nombre de dimensions

(Validation croisée, Bro, 2008 ; GCV, Josse & Husson, 2011)

```
> library(missMDA)
> nb <- estim_ncpPCA(don, method.cv="Kfold")
> nb$ncp      #2
> plot(0:5, nb$criterion, xlab="nb dim", ylab="MSEP")
```



⇒ Etape 2 : Imputation des données manquantes

```
> res.comp <- imputePCA(don, ncp = 2)
```

```
> res.comp$completeObs[1:3,]
```

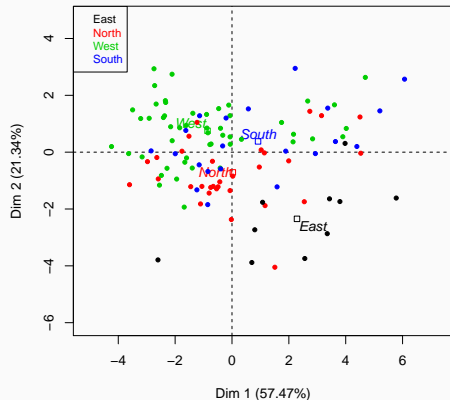
	max03	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	max03v
0601	87	15.60	18.50	20.47	4	4.00	8.00	0.69	-1.71	-0.69	84
0602	82	18.51	20.88	21.81	5	5.00	7.00	-4.33	-4.00	-3.00	87
0603	92	15.30	17.60	19.50	2	3.98	3.81	2.95	1.97	0.52	82



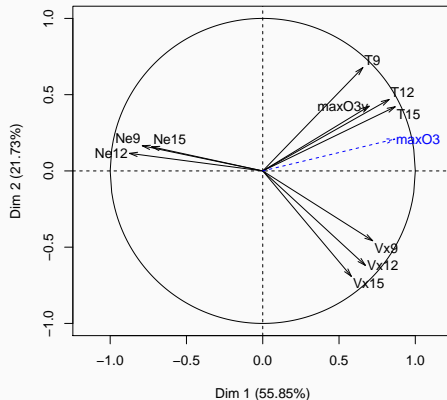
# ACP sur le tableau complété

⇒ Etape 3 : ACP sur le tableau complété

Individuals factor map (PCA)



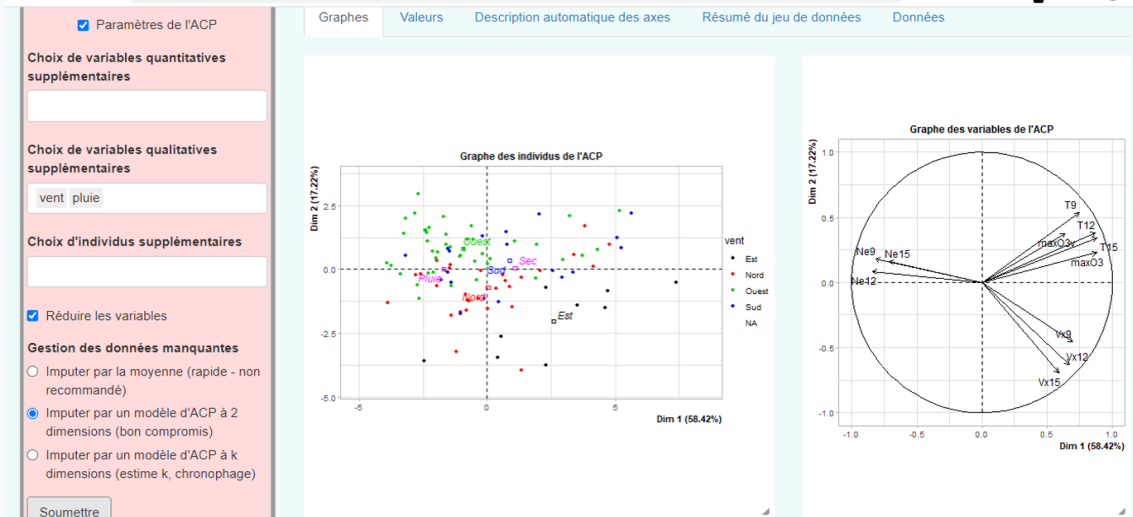
Variables factor map (PCA)



```
> imp <- cbind.data.frame(res.comp$completeObs, ozone[,12])  
> res.pca <- PCA(imp, quanti.sup=1, quali.sup=12)  
> plot(res.pca, hab=12, lab="quali")  
> plot(res.pca, choix="var")
```

# 3 en 1 avec le package Factoshiny

- > library(Factoshiny)
- > Factoshiny(ozone)



## Quid des éléments supplémentaires ?

Idée : pondérer les éléments supplémentaires (variables quantitatives, individus supplémentaires)

- ❶ Mettre un poids aux éléments supplémentaires qui ne contribueront pas à la construction des dimensions
- ❷ Lancer l'algorithme d'ACP itérative régularisée avec ces poids : l'imputation n'utilise pas l'information portée par les éléments supplémentaires
- ❸ Lancer ensuite l'ACP sur le tableau complété en utilisant la fonction classique d'ACP avec éléments supplémentaires

- ① Introduction
- ② ACP et reconstitution de données
- ③ Algorithme d'ACP itérative
- ④ Régularisation de l'ACP itérative
- ⑤ Mise en œuvre pratique
- ⑥ Conclusion

## Bilan

- L'ACP itérative régularisée permet d'imputer les données manquantes d'un jeu de données incomplet
- Le tableau imputé peut être directement utilisé avec un algorithme classique d'ACP
- Les imputations n'ont pas de poids dans le critère utilisé pour construire axes et composantes d'une ACP

## Remarque

*"The idea of imputation is both seductive and dangerous. It is seductive because it can lull the user into the pleasurable state of believing that the data are complete after all, and it is dangerous because it lumps together situations where the problem is sufficiently minor that it can be legitimately handled in this way and situations where standard estimators applied to the real and imputed data have substantial biases." (Dempster & Rubin, 1983)*

## Remarque

*"The idea of imputation is both seductive and dangerous. It is seductive because it can lull the user into the pleasurable state of believing that the data are complete after all, and it is dangerous because it lumps together situations where the problem is sufficiently minor that it can be legitimately handled in this way and situations where standard estimators applied to the real and imputed data have substantial biases."* (Dempster & Rubin, 1983)

Quelques problèmes pratiques sur l'imputation :

- Imputation de  $X$  et  $X^2$
- Problèmes de bornes ( $> 0$ )  $\Rightarrow$  tronquer ?
- Comment faire avec des données de grandes dimensions ?

[http://factominer.free.fr/missMDA/index\\_fr.html](http://factominer.free.fr/missMDA/index_fr.html)

Accueil	Méthodes FactoMineR	Enseignement MOOC, livres	Améliorations graphiques	Valeurs manquantes missMDA	Rapport automatique	Google group	Plus
---------	---------------------	---------------------------	--------------------------	----------------------------	---------------------	--------------	------



## Le package missMDA

Le package **missMDA** est complémentaire de FactoMineR. Il permet de gérer les données manquantes pour les méthodes d'analyses factorielles (ACP, AFC, ACM, AFDM, AFM). Il permet de faire de l'imputation simple et multiple.

L'imputation simple consiste à remplacer les valeurs manquantes par des valeurs plausibles. Cela revient à compléter le jeu de données qui peut ensuite être analysé par n'importe quelle méthode d'analyse factorielle.

**missMDA** impute les valeurs manquantes de sorte que les valeurs imputées n'ont aucune influence sur les résultats de l'analyse factorielle (pas d'influence dans le sens où les valeurs imputées n'ont aucun poids, et donc les résultats de l'analyse factorielle sont obtenus uniquement avec les valeurs observées).

**missMDA** utilise des méthodes de réduction de données, ce qui lui permet d'imputer de façon satisfaisante de gros jeux de données contenant des variables quantitatives et/ou qualitatives. En effet, il impute par ACP (ou ACM, ou AFDM ou AFM) en prenant en compte à la fois les similarités entre individus et les liens entre variables.

Voir cette vidéo si vous voulez comprendre le principe de missMDA quelque soit les jeux de données (quantitatifs et/ou qualitatifs).

Les imputations sont très bonnes comparées aux méthodes classiques permettant d'imputer des tableaux incomplets (forêts aléatoires par exemple).

- **missMDA** gère les données manquantes dans:
  - les jeux de données avec variables quantitatives grâce à l'ACP (Voir la vidéo)
  - les jeux de données avec variables qualitatives grâce à l'ACM (Voir la vidéo)
  - les tableaux de contingence grâce à l'AFC
  - les données mixtes grâce à l'AFDM
  - les jeux de données où les variables sont structurées par groupe grâce à l'AFM
- **missMDA** permet de faire de l'imputation multiple:
  - pour les variables quantitatives grâce à l'ACP: Voir la vidéo
  - pour les variables qualitatives grâce à l'ACM

## Menu sur les données manquantes

### Le package missMDA

ACP avec données manquantes

ACM avec données manquantes

Imputation multiple

Peut-on croire dans les valeurs imputées ?

Références - Conférences

## Les auteurs de missMDA

François Husson

Julie Josse



⇒ Logiciels :

- Package missMDA (utilisé avec FactoMineR ou Factoshiny)
- [R CRAN task View: Missing Data](#)
- [R-miss-tastic](#)

⇒ Articles :

- Imbert, A., & Vialaneix, N. (2018). Décrire, prendre en compte, imputer et évaluer les valeurs manquantes dans les études statistiques : une revue des approches existantes. *Journal de la SFdS*, **159(2)**, 1-55.
- Josse J, Husson F. & Pagès J (2009) Gestion des données manquantes en Analyse en Composantes Principales. *Journal de la SFdS*. **150 (2)**, 28-51.