# Multiple imputation based on principal components methods

Vincent Audigier

CNAM, Paris

Paris Dauphine
2019

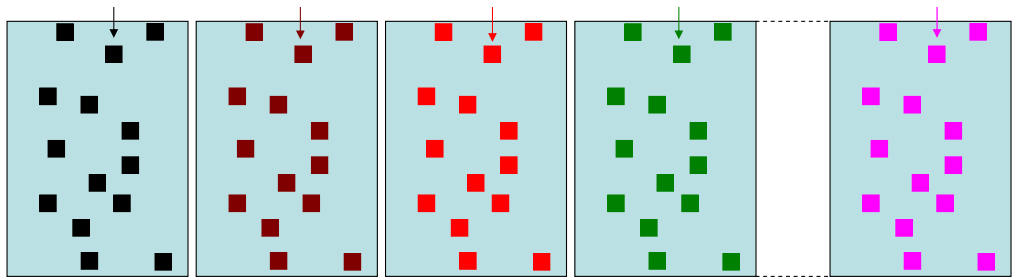
# Missing data

- Missing data are very frequent
- An issue for applying statistical methods
- Three kinds of missing data mechanisms (MCAR, MAR, MNAR)
- Multiple imputation is one solution

# Multiple imputation (Rubin, 1987)

1. Generate a set of $M$ parameters $(\theta_m)_{1\leq m\leq M}$ of an imputation model to generate $M$ plausible imputed data sets

$P\left(X^{miss}|X^{obs},\theta_1\right)$ . . . . . $P\left(X^{miss}|X^{obs},\theta_M\right)$



2. Fit the analysis model on each imputed data set: $\hat{\beta}_m, \widehat{\text{Var}}\left(\hat{\beta}_m\right)$

3. Combine the results: $\hat{\beta} = \frac{1}{M}\sum_{m=1}^{M}\hat{\beta}_m$

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

$\Rightarrow$ Provide estimation of the parameters and of their variability

# Steps for data imputation

1. Choose an imputation model
   - a model for all variables (JM)
   - a model for each conditionnal distribution (FCS)
2. Generate *M* values for its parameters
   - Bayesian (or Approximate Bayesian) or Booststrap
   - Require handling missing values
3. Impute missing data according to each ones (conditionally to observed values)

# Example: JM multivarate Gaussian distribution (King et al., 2001)

1. Imputation model $X_{n\times p}$ : $X_{i.} \sim \mathcal{N}\left(\mu_{p\times 1}, \Sigma_{p\times p}\right)$ (Honaker et al., 2011)
2. Generate $M$ values for $\theta = (\mu, \Sigma)$
   - Bootstrap rows $X^1, \ldots, X^M$
   - Fit the Gaussian model on each one using EM algorithm:

$$\longrightarrow \theta^1 = (\mu^1, \Sigma^1), \ldots, \theta^M = (\mu^M, \Sigma^M)$$

3. Imputation
   - for a $\theta^m$, derive the conditional distribution of $X^{miss}|X^{obs}$ for an incomplete individual $x_i = \left(x_i^{obs}, x_i^{miss}\right)$
   - make a draw from it to impute the individual
   - repeat the procedure over all incomplete individuals to obtain one imputed table
   - repeat over all $\theta^m$ $(1 \leq m \leq M)$

# Challenges

MI can be tricky with

- high dimensionality ($n < p$)
- high dependence
- categorical variables
- large data
- complex data
- ...

MI methods based on principal components methods can provide solutions

# Outline

# PCA

$\Rightarrow$ Geometrical point of view

$$\|\mathbf{X}_{n\times p} - \hat{\mathbf{X}}^{S}_{n\times p}\|^2 \qquad \hat{\mathbf{X}}^{S} = \widehat{\mathbf{U}}_{n\times S}\widehat{\mathbf{\Lambda}}^{\frac{1}{2}}_{S\times S}\widehat{\mathbf{V}}^{\top}_{p\times S}$$

- $\widehat{\mathbf{F}} = \widehat{\mathbf{U}}\widehat{\mathbf{\Lambda}}^{\frac{1}{2}}$ principal components - scores
- $\widehat{\mathbf{V}}$ principal axes - loadings

$\Rightarrow$ Model point of view: $\mathbf{X}_{n\times p} = \tilde{\mathbf{X}}_{n\times p} + \varepsilon_{n\times p}$

$$\begin{aligned} x_{ij} &= \tilde{x}_{ij} + \varepsilon_{ij} \text{ , } \varepsilon_{ij} \sim \mathcal{N}(0,\sigma^2) \\ &= \sum_{s=1}^{S} \sqrt{\lambda_s} u_{is} v_{js} + \varepsilon_{ij} \end{aligned}$$

Max Likelihood = Min least squares

# Imputation with PCA

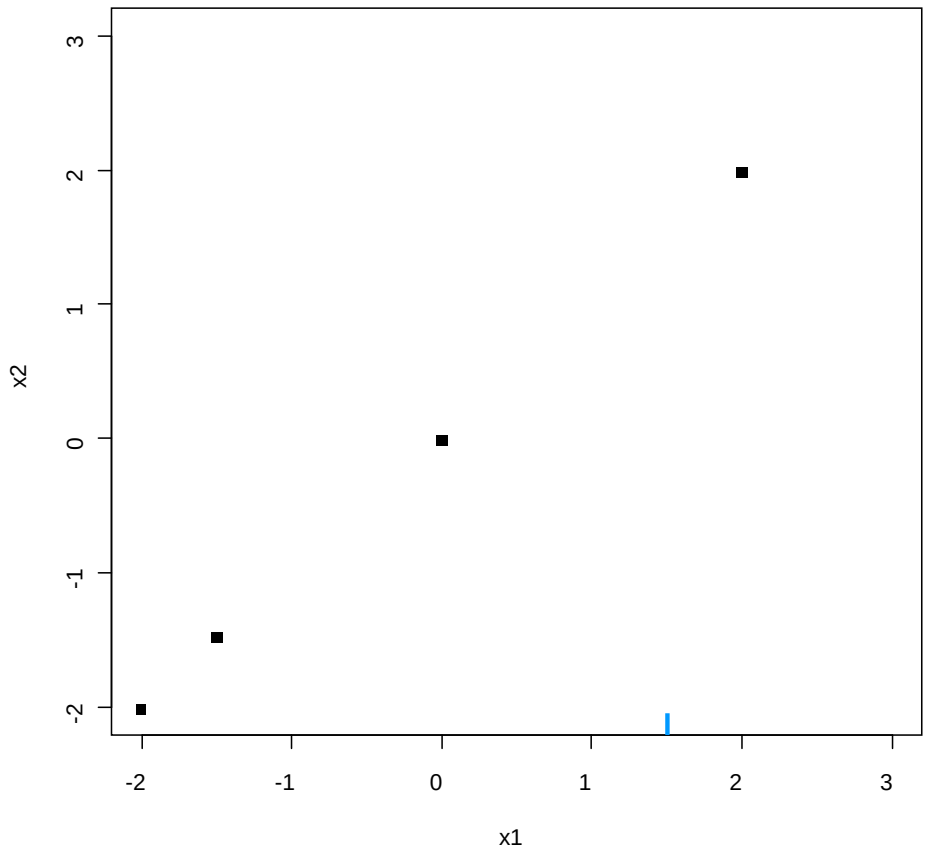
$$
\begin{aligned}
x_{ij} &= \tilde{x}_{ij} + \varepsilon_{ij} \text{ , } \varepsilon_{ij} \sim \mathcal{N}(0,\sigma^2) \\
&= \sum_{s=1}^{S} \sqrt{\lambda_s} u_{is} v_{js} + \varepsilon_{ij}
\end{aligned}
$$

- An EM algorithm to estimate the parameters: iterative PCA (Kiers 1997)
- Fitted values as imputed values (imputation using conditionnal expectation)
- Add a Gaussian noise for stochastic single imputation

# Algorithm: iterative PCA

Data set

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

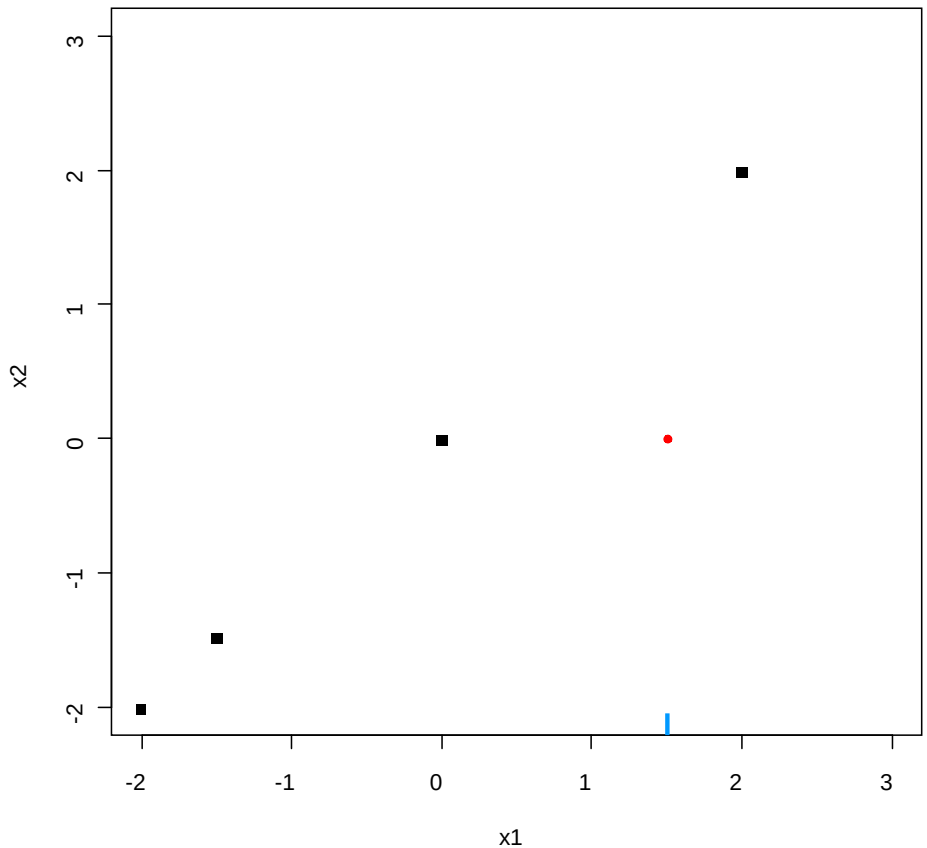


# Algorithm: iterative PCA

| 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: mean imputation



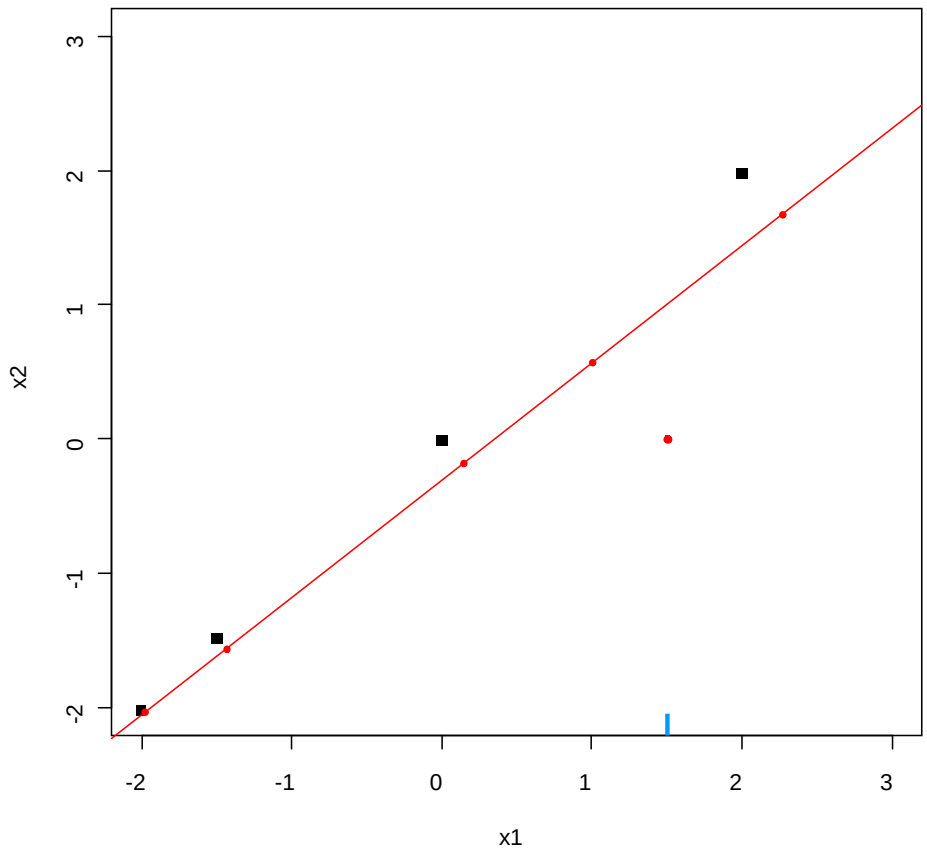
# Algorithm: iterative PCA

PCA on the imputed data set using 1 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 | **0.00** |
| 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 |



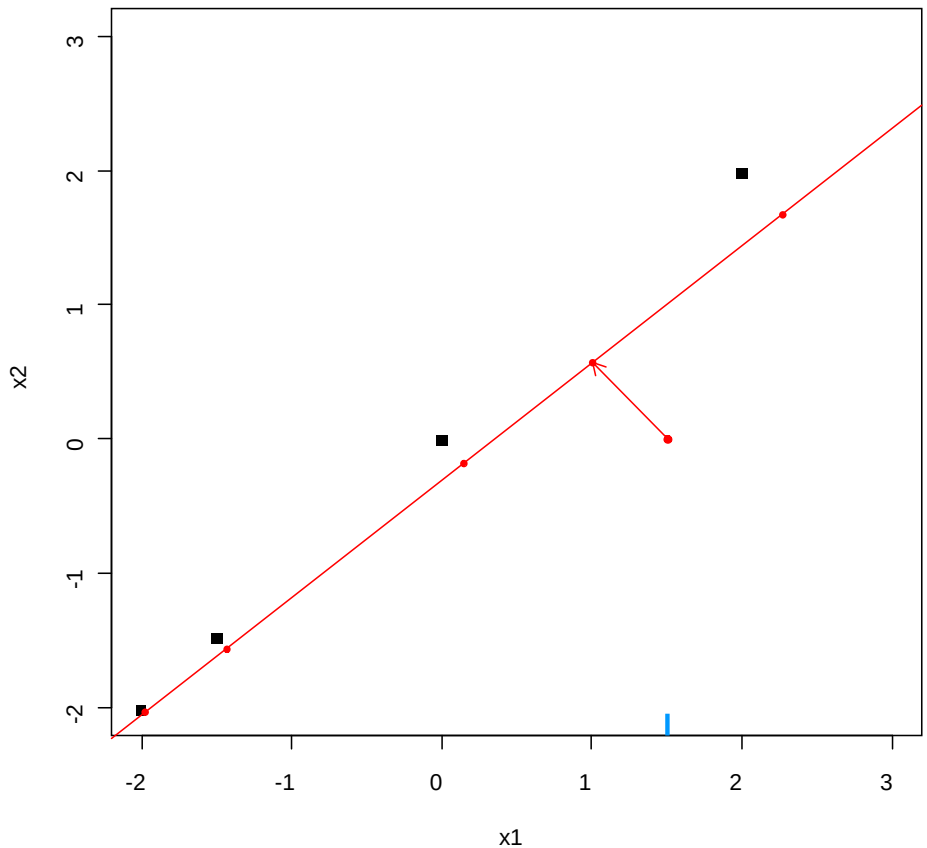
# Algorithm: iterative PCA

Find the predicted values given by the model

| 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 |



# Algorithm: iterative PCA

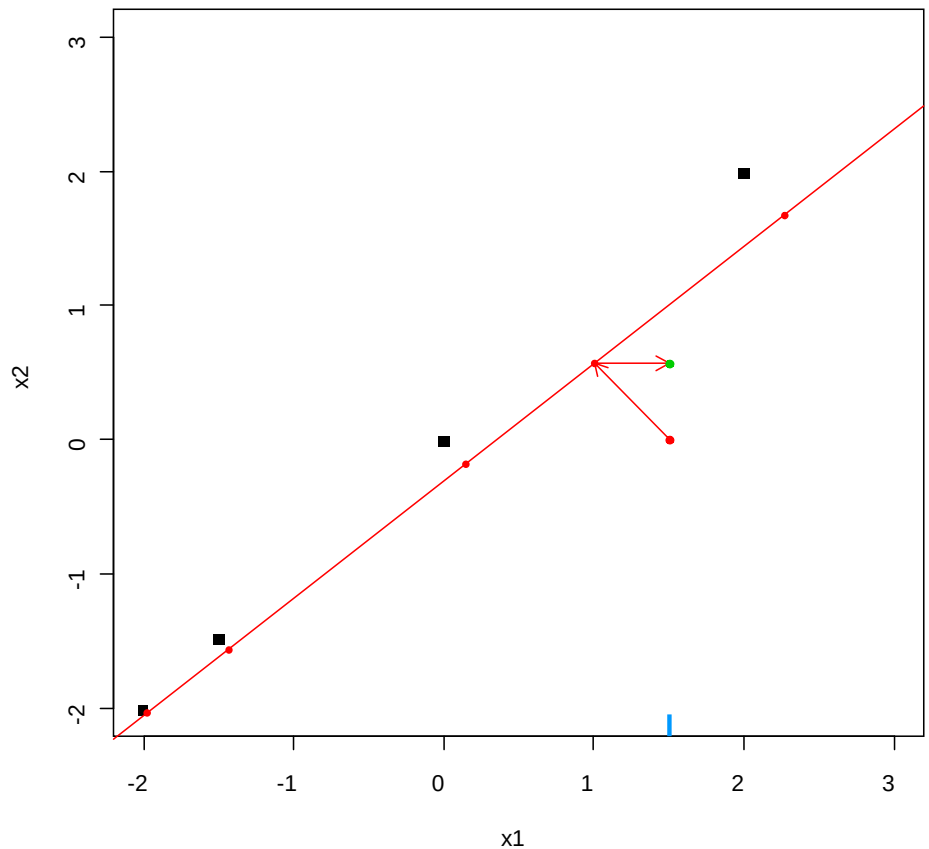
Imputation step: $X^{\ell} = W * X + (1 - W) * \hat{X}^{\ell}$

| 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 |



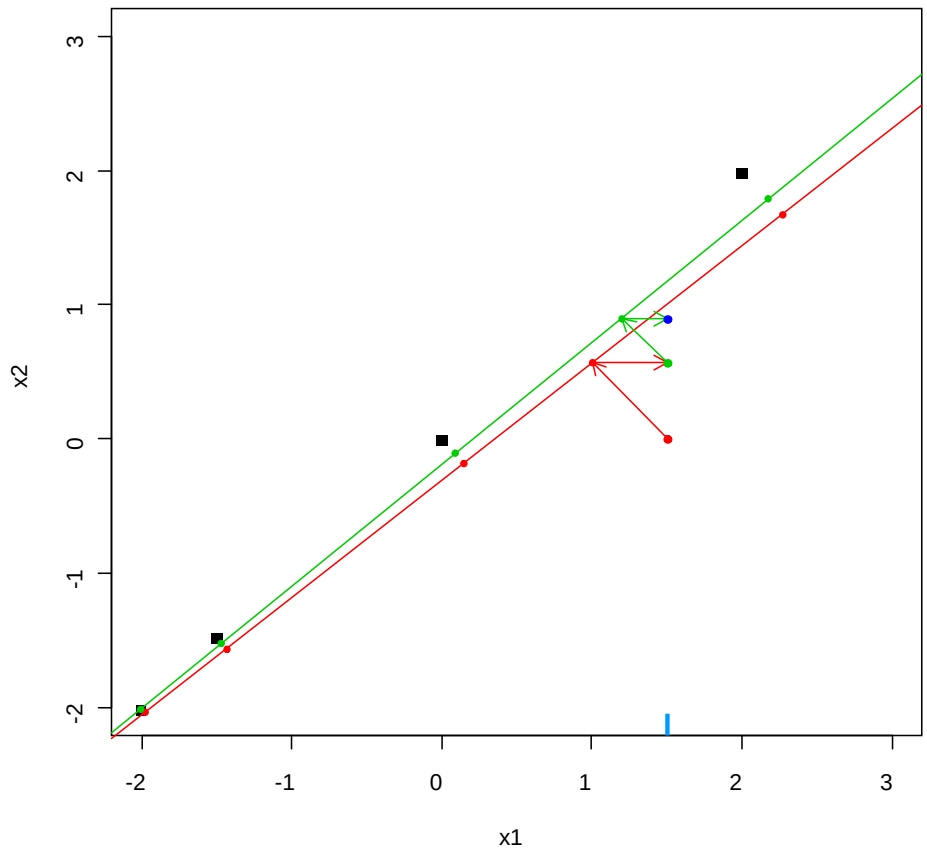
# Algorithm: iterative PCA

PCA on the imputed data set using 1 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 |



# Algorithm: iterative PCA

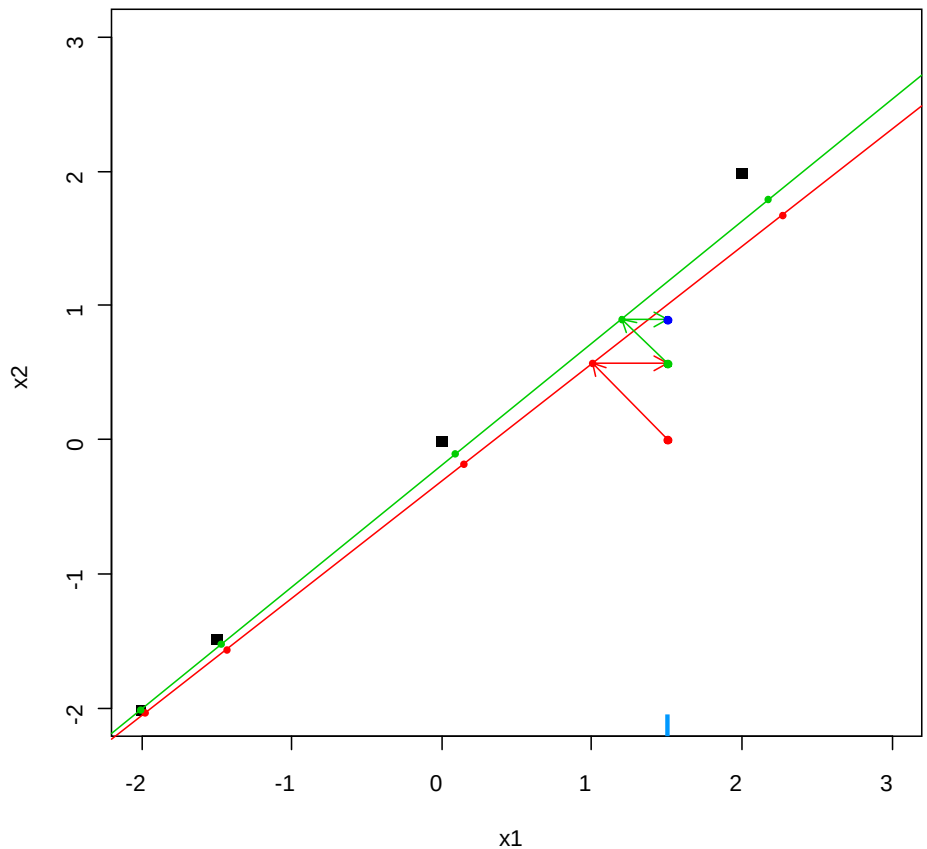
Imputation step: $X^{\ell} = W * X + (1 - W) * \hat{X}^{\ell}$

```
  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
```

```
   ^     ^
  x1    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
```



# Algorithm: iterative PCA

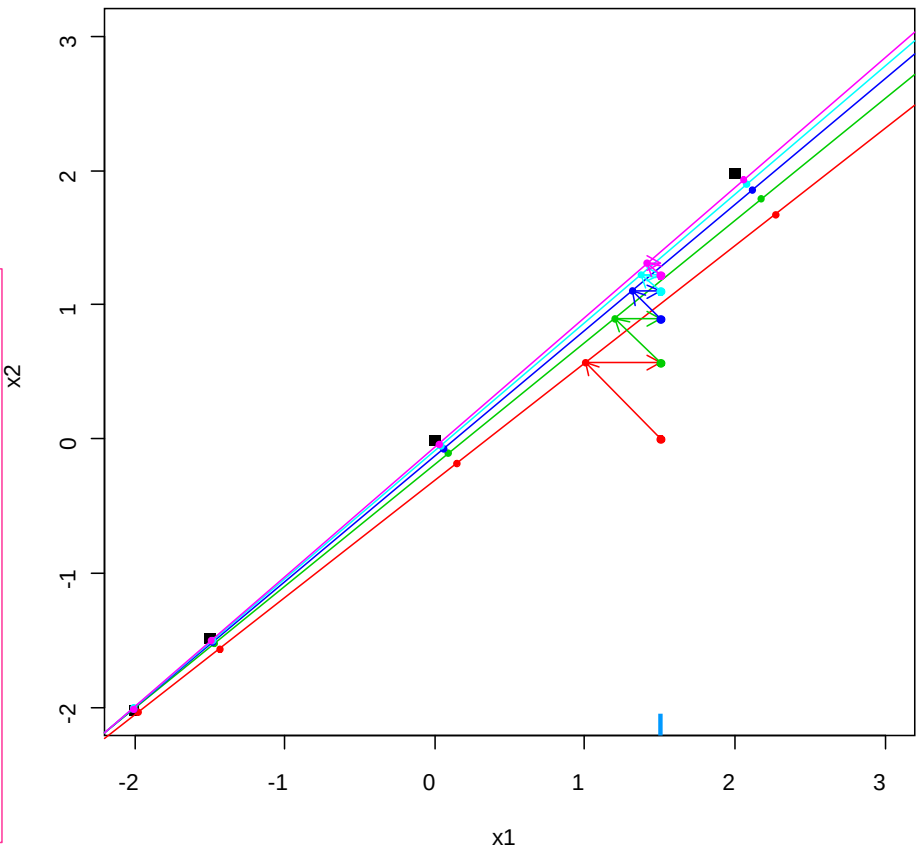
Repeated until convergence

| 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 |

| $\hat{x}1$ | $\hat{x}2$ |
|---|---|
| -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 |



# Algorithm: iterative PCA

1. initialisation $\ell = 0$ : $X^0$ (mean imputation)
2. step $\ell$ :
   (a) Estimate $\widehat{\mathbf{U}}^\ell$, $\widehat{\mathbf{\Lambda}}^\ell$, $\widehat{\mathbf{V}}^\ell$
   (b) Imputation with $\hat{x}_{ij}^\ell = \sum_{s=1}^{S} \left(\sqrt{\hat{\lambda}_s}\right) \hat{u}_{is}^\ell \hat{v}_{js}^\ell$
3. (a) and (b) are repeated until convergence

# Regularization

- Iterative PCA can suffer from instability
  - when the data structure is weak
  - when the proportion of missing values is high

- Regularization: weighting axes according to the proportion of inertia they support
  - high weight if the proportion is large
  - low weight otherwise

1. initialisation $\ell = 0$ : $X^0$ (mean imputation)
2. step $\ell$ :
   (a) Estimate $\widehat{\mathbf{U}}^\ell$ , $\widehat{\mathbf{\Lambda}}^\ell$ , $\widehat{\mathbf{V}}^\ell$
   (b) Imputation with $\hat{x}_{ij}^\ell = \sum_{s=1}^{S} \left( \sqrt{\hat{\lambda}_s} - \frac{\hat{\sigma}^2}{\hat{\lambda}_s} \right) \hat{u}_{is}^\ell \hat{v}_{js}^\ell$
3. (a) and (b) are repeated until convergence

# How to choose the number of dimensions?

By cross-validation procedures:

- adding missing values on the incomplete data set
- predicting each of them using PCA for several number of dimensions
- calculating the prediction error

Several ways:

- Leave-one-out (Bro et al., 2008)
- Repeated cross-validation
- GCV criterion (Josse and Husson, 2011)

# Outline

# MIPCA algorithm

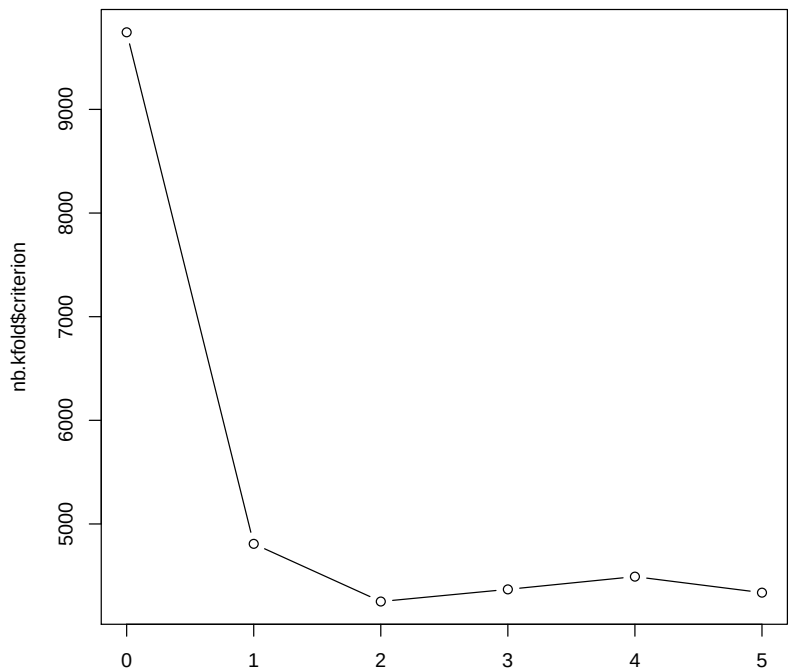
1. Initialisation
   - Estimate of PCA parameters
     $X_{n\times p} = \mathbf{U}_{n\times s}\mathbf{\Lambda}^{\frac{1}{2}}_{s\times s}\mathbf{V}^{\top}_{p\times s} + E_{n\times p}$ avec $E = (\epsilon_{ij})_{ij}$ et
     $\epsilon_{ij} \sim \mathcal{N}(0,\sigma^2) \Rightarrow \widehat{\mathbf{U}}_{n\times s}$, $\widehat{\mathbf{\Lambda}}_{s\times s}$, $\widehat{\mathbf{V}}_{p\times s}$, $\hat{\sigma}^2$
2. Repeat $M$ times
   - Generate values for $\theta^m = (\mathbf{U}^m_{n\times s}, \mathbf{\Lambda}^m_{s\times s}, \mathbf{V}^m_{p\times s})$
     - Create $X^{\star}_m = \widehat{\mathbf{U}}_{n\times s}\widehat{\mathbf{\Lambda}}^{\frac{1}{2}}_{s\times s}\widehat{\mathbf{V}}^{\top}_{p\times s} + E^{\star}$ with $E^{\star} \sim \mathcal{N}(0,\hat{\sigma}^2)$
     - PCA on $X^{\star}_m \Rightarrow \widehat{X}_m = \widehat{\mathbf{U}}^{\star}_{n\times s}\widehat{\mathbf{\Lambda}^{\frac{1}{2}}}^{\star}_{s\times s}\widehat{\mathbf{V}}^{\star\top}_{p\times s}$, $\hat{\sigma}^{\star 2}_m$
   - Impute data
     - according to the conditional expectation :
       $X_m \leftarrow W * X + (1 - W) * \hat{X}^{\star}_m$
     - add a Gaussian noise: $X_m \leftarrow X_m + \tilde{E}$ draw from $\mathcal{N}\left(0,\hat{\sigma}^{\star 2}_m\right)$

# MIPCA algorithm

- A Bayesian version has been also developed
  - more technical
  - inherits from the same properties
  - a assessed methodology
- The algorithm is implemented in the R package missMDA

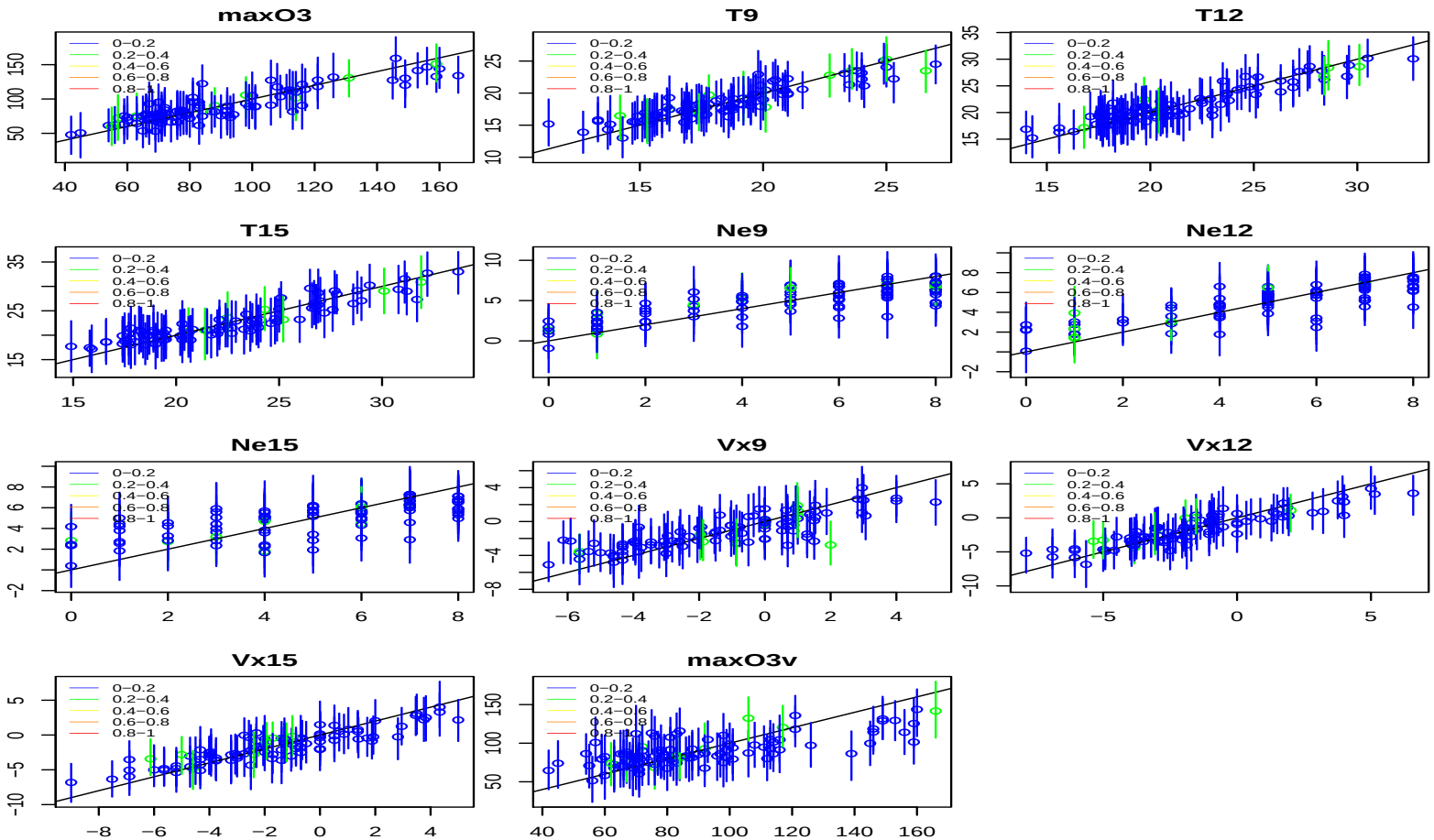
# Example: number of dimensions

```
library(missMDA);library(mice);data(ozone)

nb.kfold <- estim_ncpPCA(ozone[,1:11],
method.cv = "Kfold")

plot(names(nb.kfold$criterion),
nb.kfold$criterion,type="b")
```



# Example: imputation and fit of the imputation model

```
res.BayesMIPCA<-MIPCA(ozone[,1:11],ncp=nb.kfold$ncp,
verbose=TRUE,method.mi = "Bayes")
res.over<-Overimpute(res.BayesMIPCA)
```



# Example: fit an analysis model

```
imp<-prelim(res.mi=res.BayesMIPCA,X=ozone[,1:11]
fit <- with(data=imp,
exp=lm(maxO3~T9+T12+T15+Ne9+Ne12+Ne15+Vx9+Vx12+Vx15+maxO3v))
res.pool<-pool(fit);summary(res.pool)
```

| | estimate | std.error | statistic | df | p.value |
|---|---|---|---|---|---|
| (Intercept) | 2.81 | 16.08 | 0.17 | 79.44 | 0.86 |
| T9 | 0.35 | 1.28 | 0.28 | 60.00 | 0.78 |
| T12 | 1.40 | 1.28 | 1.09 | 52.26 | 0.28 |
| T15 | 1.64 | 0.99 | 1.65 | 60.13 | 0.10 |
| Ne9 | -1.59 | 1.15 | -1.38 | 72.21 | 0.17 |
| Ne12 | -1.03 | 1.49 | -0.69 | 66.40 | 0.49 |
| Ne15 | 0.62 | 1.06 | 0.58 | 70.94 | 0.56 |
| Vx9 | 0.86 | 1.01 | 0.85 | 71.43 | 0.40 |
| Vx12 | 0.48 | 1.18 | 0.41 | 64.15 | 0.68 |
| Vx15 | 0.02 | 1.05 | 0.02 | 67.91 | 0.98 |
| maxO3v | 0.29 | 0.07 | 3.91 | 75.31 | 0.00 |

# MI methods for continuous data

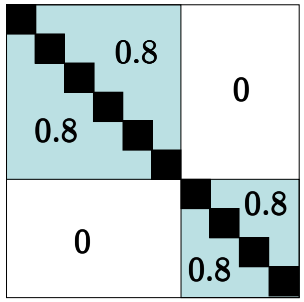
Generally based on normal distribution:

- JM: $X_{n\times p} : X_{i.} \sim \mathcal{N}(\mu,\Sigma)$ (Honaker et al., 2011)
  1. Bootstrap rows: $X^1,\ldots,X^M$
     EM algorithm: $\theta^1 = (\mu^1,\Sigma^1),\ldots,\theta^M = (\mu^M,\Sigma^M)$
  2. Imputation: $x_{i.}^m$ drawn from $\mathcal{N}\left(\mu^m_{X^{miss}|X^{obs}},\Sigma^m_{X^{miss}|X^{obs}}\right)$

- FCS: $\mathcal{N}\left(\mu_{X_j|X_{(-j)}},\Sigma_{X_j|X_{(-j)}}\right)$ (Van Buuren, 2012)
  1. Bayesian approach: $\theta^m = (\beta^m,\sigma^m)$
  2. Imputation: stochastic regression $x_{ij}^m$ drawn from $\mathcal{N}\left(X_{(-j)}\beta^m,\sigma^m\right)$

# Simulations

- Quantities of interest: $\theta_1 = \mathbb{E}[Y], \theta_2 = \beta_1, \theta_3 = \rho$
- 1000 simulations
  - data set drawn from $\mathcal{N}_p(\mu, \Sigma)$ with a two-block structure, varying $n$ (30 or 200), $p$ (6 or 60) and $\rho$ (0.3 or 0.9)




  - 10% or 30% of missing values using a MCAR mechanism
  - multiple imputation using $M = 20$ imputed arrays
- Criteria
  - bias
  - CI width, coverage

# Results for the expectation

| | parameters | | | | confidence interval width | | | coverage | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $n$ | $p$ | $\rho$ | % | *JM* | *FCS* | *BayesMIPCA* | *JM* | *FCS* | *BayesMIPCA* |
| 1 | 30 | 6 | 0.3 | 0.1 | 0.803 | 0.805 | 0.781 | 0.955 | 0.953 | 0.950 |
| 2 | 30 | 6 | 0.3 | 0.3 | | 1.010 | 0.898 | | 0.971 | 0.949 |
| 3 | 30 | 6 | 0.9 | 0.1 | 0.763 | 0.759 | 0.756 | 0.952 | 0.95 | 0.949 |
| 4 | 30 | 6 | 0.9 | 0.3 | | 0.818 | 0.783 | | 0.965 | 0.953 |
| 5 | 30 | 60 | 0.3 | 0.1 | | | 0.775 | | | 0.955 |
| 6 | 30 | 60 | 0.3 | 0.3 | | | 0.864 | | | 0.952 |
| 7 | 30 | 60 | 0.9 | 0.1 | | | 0.742 | | | 0.953 |
| 8 | 30 | 60 | 0.9 | 0.3 | | | 0.759 | | | 0.954 |
| 9 | 200 | 6 | 0.3 | 0.1 | 0.291 | 0.294 | 0.292 | 0.947 | 0.947 | 0.946 |
| 10 | 200 | 6 | 0.3 | 0.3 | 0.328 | 0.334 | 0.325 | 0.954 | 0.959 | 0.952 |
| 11 | 200 | 6 | 0.9 | 0.1 | 0.281 | 0.281 | 0.281 | 0.953 | 0.95 | 0.952 |
| 12 | 200 | 6 | 0.9 | 0.3 | 0.288 | 0.289 | 0.288 | 0.948 | 0.951 | 0.951 |
| 13 | 200 | 60 | 0.3 | 0.1 | | 0.304 | 0.289 | | 0.957 | 0.945 |
| 14 | 200 | 60 | 0.3 | 0.3 | | 0.384 | 0.313 | | 0.981 | 0.958 |
| 15 | 200 | 60 | 0.9 | 0.1 | | 0.282 | 0.279 | | 0.951 | 0.948 |
| 16 | 200 | 60 | 0.9 | 0.3 | | 0.296 | 0.283 | | 0.958 | 0.952 |

# Properties for BayesMIPCA

A MI method based on a Bayesian treatment of the PCA model

advantages

- captures the structure of the data: good inferences for regression coefficient, correlation, mean
- a dimensionality reduction method: ($n < p$ or $n > p$, low or high percentage of missing values)
- no inversion issue: strong or weak relationships
- a regularization strategy improving stability

remains competitive if:

- the low rank assumption is not verified
- the Gaussian assumption is not true

# Imputation with MCA for categorical variables

MCA can be seen as the PCA on an indicator matrix **X** with specific weights depending on $\mathbf{D}_\Sigma$

$$X = \begin{array}{|ccc|cc|cc|c|}\hline 1&0&0&1&0&0&1&\ldots&0&1\\ 1&0&0&1&0&1&0&\ldots&\text{NA}&\text{NA}\\ \text{NA}&\text{NA}&\text{NA}&0&1&0&0&\ldots&0&1\\ 1&0&0&1&0&0&1&\ldots&0&1\\ & & & & & & & & & \\ & & & & & & & & & \\ & & & & & & & & & \\ & & & & & & & & & \\ & & & & & & & & & \\ 0&0&1&\text{NA}&\text{NA}&0&\ldots&0&1&\\ 1&0&0&1&0&0&1&\ldots&0&1\\ \hline \end{array} \begin{array}{c} J \\ \\ \\ \\ \\ J \\ \\ \\ \\ \\ J \end{array}$$

$$\begin{array}{ccc} I_1 & I_k & I_K \end{array} \quad IJ$$

$$D_\Sigma = \begin{bmatrix} I_1 & & & 0 \\ & \ddots & & \\ & & I_k & \\ 0 & & & \ddots \\ & & & & I_K \end{bmatrix}$$

# Multiple imputation for categorical data using MCA

MI for categorical data is very challenging for a moderate number of variables

- estimation issues
- storage issues

MI with MCA

1. Variability on the parameters of the imputation model

$$\left(\left(\hat{\mathbf{U}}_{n\times S},\hat{\Lambda}^{1/2}_{S\times S},\hat{\mathbf{V}}^{\top}_{p\times S}\right)_1,\ldots,\left(\hat{\mathbf{U}}_{n\times S},\hat{\Lambda}^{1/2}_{S\times S},\hat{\mathbf{V}}^{\top}_{p\times S}\right)_M\right)$$

$\rightarrow$ A non-parametric bootstrap approach

2. Add a disturbance on the MCA prediction $\hat{X}_m = \hat{\mathbf{U}}_m\hat{\Lambda}^{1/2}_m\hat{\mathbf{V}}^{\top}_m$

# Multiple imputation with MCA

1. Variability of the parameters of MCA ($\hat{\mathbf{U}}_{n\times S}, \hat{\Lambda}^{1/2}_{S\times S}, \hat{\mathbf{V}}^{\top}_{p\times S}$) using a non-parametric bootstrap
2. Imputation:

$\hat{\mathbf{X}}_1$ $\hat{\mathbf{X}}_2$ $\hat{\mathbf{X}}_M$

| | | | | |
|---|---|---|---|---|
| 1 | 0 | ... | 1 | 0 |
| 1 | 0 | ... | 1 | 0 |
| 1 | 0 | ... | **0.81** | **0.19** |
| **0.25** | **0.75** | | 0 | 1 |
| 0 | 1 | | 0 | 1 |

| | | | | |
|---|---|---|---|---|
| 1 | 0 | ... | 1 | 0 |
| 1 | 0 | ... | 1 | 0 |
| 1 | 0 | ... | **0.60** | **0.40** |
| **0.26** | **0.74** | | 0 | 1 |
| 0 | 1 | | 0 | 1 |

...

| | | | | |
|---|---|---|---|---|
| 1 | 0 | ... | 1 | 0 |
| 1 | 0 | ... | 1 | 0 |
| 1 | 0 | ... | **0.74** | **0.16** |
| **0.20** | **0.80** | | 0 | 1 |
| 0 | 1 | | 0 | 1 |

Draw categories from the values of $\left(\hat{\mathbf{X}}_m\right)_{1\leq m\leq M}$

| | | |
|---|---|---|
| A | ... | A |
| A | ... | A |
| A | ... | **B** |
| **B** | ... | C |
| B | ... | B |

| | | |
|---|---|---|
| A | ... | A |
| A | ... | A |
| A | ... | **A** |
| **B** | ... | C |
| B | ... | B |

...

| | | |
|---|---|---|
| A | ... | A |
| A | ... | A |
| A | ... | **B** |
| **B** | ... | C |
| B | ... | B |

# Example

```
data(TitanicNA)
nb <- estim_ncpMCA(TitanicNA)
res.mi <- MIMCA(TitanicNA, ncp=5, verbose=TRUE)
imp<-prelim(res.mi,TitanicNA)
fit <- with(data=imp,exp=glm(SURV~CLASS+AGE+SEX,family = "binomial"))
res.pool<-pool(fit)
summary(res.pool)
```

| | estimate | std.error | statistic | df | p.value |
|---|---|---|---|---|---|
| (Intercept) | 2.39 | 0.46 | 5.21 | 237.85 | 0.00 |
| CLASS1 | 0.84 | 0.20 | 4.21 | 502.47 | 0.00 |
| CLASS2 | -0.17 | 0.22 | -0.75 | 448.99 | 0.46 |
| CLASS3 | -0.98 | 0.19 | -5.27 | 566.77 | 0.00 |
| AGE1 | -1.04 | 0.40 | -2.61 | 197.42 | 0.01 |
| SEX1 | -2.64 | 0.18 | -14.84 | 509.34 | 0.00 |

# MI methods for categorical data

- Log-linear model (Schafer, 1997)
  - Hypothesis on $X = (x_{ijk})_{i,j,k}$: $X|\theta \sim \mathcal{M}(n,\theta)$

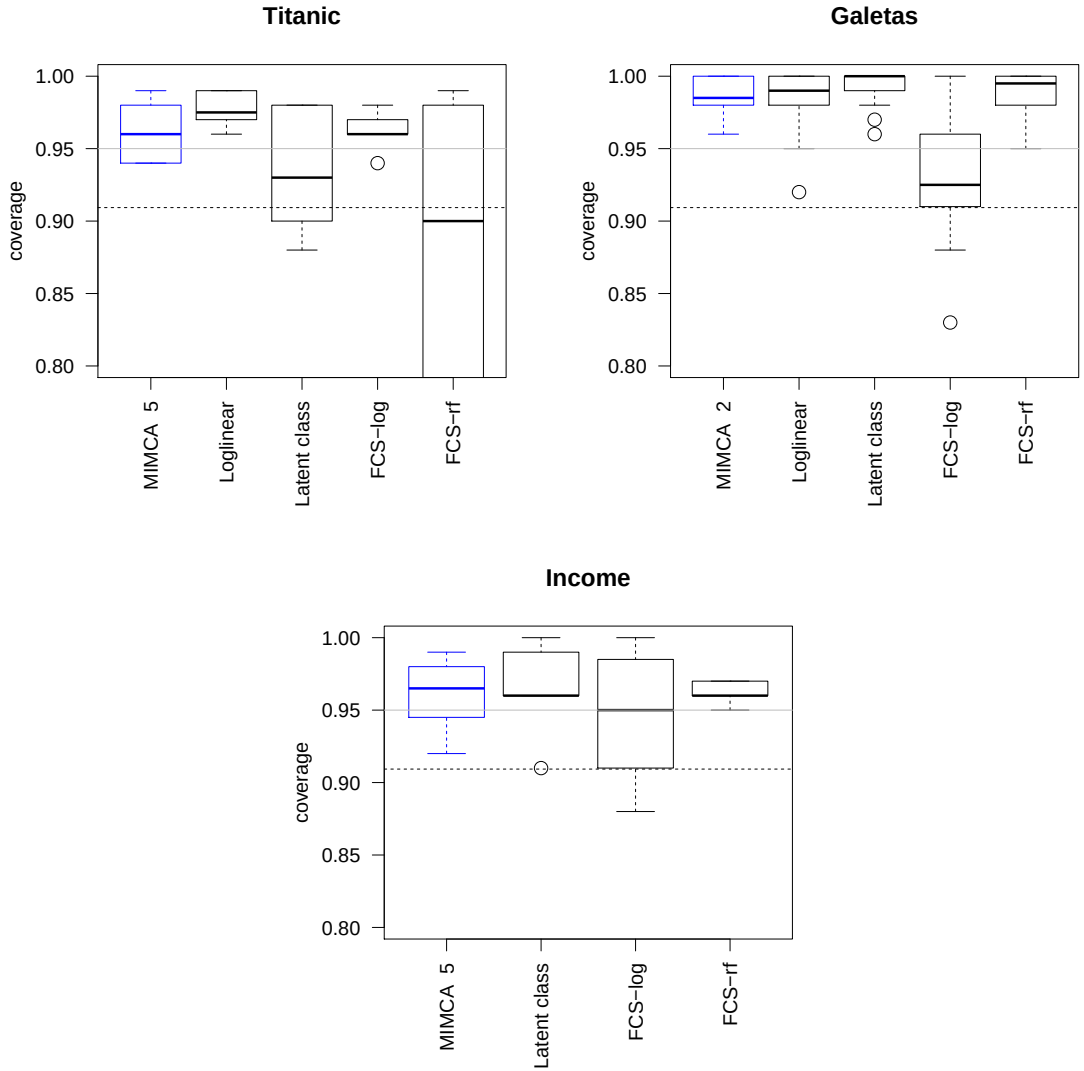
    $$log(\theta_{ijk}) = \lambda_0 + \lambda_i^A + \lambda_j^B + \lambda_k^C + \lambda_{ij}^{AB} + \lambda_{ik}^{AC} + \lambda_{jk}^{BC} + \lambda_{ijk}^{ABC}$$

  1. Variability of the parameter $\theta$: Bayesian formulation
  2. Imputation using the set of $M$ parameters
- Latent class model (Si and Reiter, 2013)
  - Hypothesis: $\mathbb{P}(X = (x_1, \ldots, x_p); \theta) = \sum_{\ell=1}^{L} \left( \theta_\ell \prod_{j=1}^{p} \theta_{x_j}^{(\ell)} \right)$
  1. Variability of the parameters $\theta_L$ and $\theta_X$ : Bayesian formulation
  2. Imputation using the set of $M$ parameters
- FCS: GLM (Van Buuren, 2012) or Random Forests (Doove et al., 2014; Shah et al., 2014)

# Simulations from real data sets

- Quantities of interest: $\theta =$ parameters of a logistic model
- Simulation design (repeated 200 times)
  - the real data set is considered as a population
  - drawn one sample from the data set
  - generate 20% of missing values
  - multiple imputation using $M = 5$ imputed arrays
- Criteria
  - bias
  - CI width, coverage
- Comparison with :
  - JM: log-linear model, latent class model
  - FCS: logistic regression, random forests

# Results - Inference



# Results - Time

| | Titanic | Galetas | Income |
|---|---|---|---|
| MIMCA | 2.750 | 8.972 | **58.729** |
| Loglinear | 0.740 | 4.597 | NA |
| Latent class model | 10.854 | 17.414 | 143.652 |
| FCS logistic | 4.781 | 38.016 | 881.188 |
| FCS forests | 265.771 | 112.987 | 6329.514 |

TAB.: Time consumed in second

| | Titanic | Galetas | Income |
|---|---|---|---|
| Number of individuals | 2201 | 1192 | 6876 |
| Number of variables | 4 | 4 | 14 |

# Conclusion

MI methods using dimensionality reduction method

- captures the relationships between variables
- captures the similarities between individuals
- requires a small number of parameters

Address some imputation issues:

- can be applied on various data sets
- provide correct inferences for analysis model based on relationships between pairs of variables

Available in the R package missMDA

# References I


D. B. Rubin. *Multiple Imputation for Non-Response in Survey*. Wiley, New-York, 1987.

G. King, J. Honaker, A. Joseph, and K. Scheve. Analyzing incomplete political science data: An alternative algorithm for multiple imputation. *American Political Science Review*, 95(1):49–69, 2001.

J. Josse and F. Husson. Selecting the number of components in PCA using cross-validation approximations. *Computational Statistics and Data Analysis*, 56 (6):1869–1879, 2011.

J. L. Schafer. *Analysis of Incomplete Multivariate Data*. Chapman & Hall/CRC, London, 1997.

V. Audigier, F. Husson, and J. Josse. Mimca: multiple imputation for categorical variables with multiple correspondence analysis. *Statistics and Computing*, 27(2): 501–518, Mar 2017.

V. Audigier, F. Husson, and J. Josse. Multiple imputation for continuous variables using a bayesian principal component analysis. *Journal of Statistical Computation and Simulation*, 86(11):2140–2156, 2016.

`factominer.free.fr/missMDA/appendix_These_Audigier.pdf`

`http://www.stefvanbuuren.nl/mi/docs/MNAR.pdf`

`http://www.stefvanbuuren.nl/mi/Software.html`