Dealing With Missing Values in R

Julie Josse

Zurich R Courses

ETH Zurich February 27-28 2020

Presentation

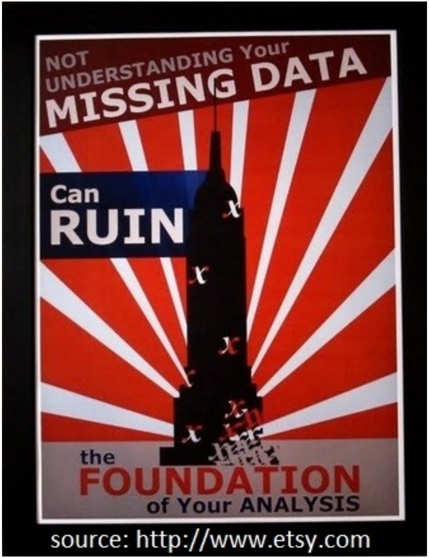
- Dimensionality reduction methods to visualize complex data (PCA based): multi-sources data, textual data, arrays
- Latent variables models
- **Missing values** - matrix completion
- Low rank estimation, selection of regularization parameters
- **Causal inference** (estimating ATE, HTE) - with missing values
- Fields of application: bio-sciences (agronomy, sensory analysis), health data (hospital APHP)
- R community: book R for Statistics, R foundation, R Forwards (widen the participation of minorities), R packages and JSS papers, R taskview on missing values, plateform rmisstastic

  FactoMineR explore continuous, categorical, multiple contingency tables (correspondence analysis), combine clustering and PC, ..

  MissMDA for single and multiple imputation, PCA with missing

  denoiseR to denoise data

Dealing with missing values



Outline

- Day 1: Morning
  - Introduction
  - Single imputation
  - Matrix completion with PCA
- Day 1: Afternoon
  - Multiple imputation

- Day 2: Morning
  - Categorical variables, mixed data
  - EM algorithms
- Day 2: Afternoon
  - Supervised learning with missing values
  - Informative missing values mechanism

Outline

Missing values

are everywhere: unanswered questions in a survey, lost data, damaged plants, machines that fail...

"The best thing to do with missing values is not to have any"

⇒ Still an issue in the "big data" area

Data integration: data from different sources

Traumabase

- 20000 patients
- 250 continuous and categorical variables: **heterogeneous**
- 11 hospitals: **multilevel data**
- 4000 new patients/ year

Center	Accident	Age	Sex	Weight	Lactactes	BP	shock	...
Beaujon	fall	54	m	85	NM	180	yes	
Pitie	gun	26	m	NR	NA	131	no	
Beaujon	moto	63	m	80	3.9	145	yes	
Pitie	moto	30	w	NR	Imp	107	no	
HEGP	knife	16	m	98	2.5	118	no	
⋮								⋱

Traumabase

- 20000 patients
- 250 continuous and categorical variables: **heterogeneous**
- 11 hospitals: **multilevel data**
- 4000 new patients/ year

Center	Accident	Age	Sex	Weight	Lactactes	BP	shock	...
Beaujon	fall	54	m	85	NM	180	yes	
Pitie	gun	26	m	NR	NA	131	no	
Beaujon	moto	63	m	80	3.9	145	yes	
Pitie	moto	30	w	NR	Imp	107	no	
HEGP	knife	16	m	98	2.5	118	no	
⋮								⋱

⇒ **Estimate causal effect**: Administration of the **treatment** "tranexamic acid" (within 3 hours after the accident) on the **outcome** mortality for traumatic brain injury patients

Traumabase

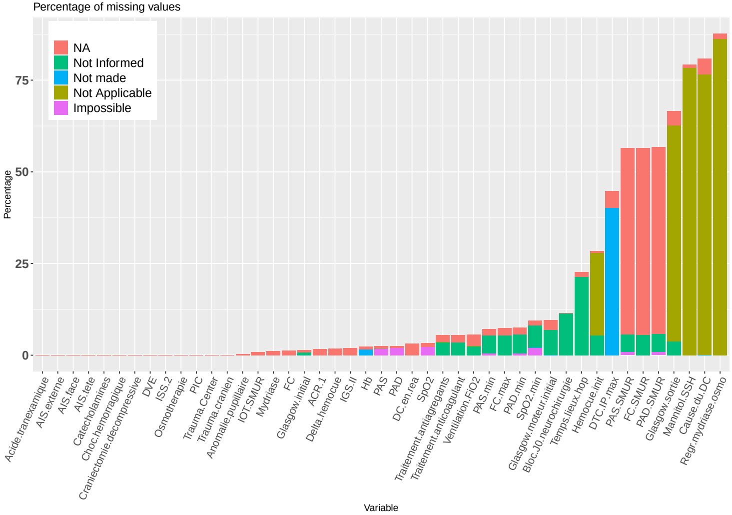
- 20000 patients
- 250 continuous and categorical variables: **heterogeneous**
- 11 hospitals: **multilevel data**
- 4000 new patients/ year

Center	Accident	Age	Sex	Weight	Lactactes	BP	shock	...
Beaujon	fall	54	m	85	NM	180	yes	
Pitie	gun	26	m	NR	NA	131	no	
Beaujon	moto	63	m	80	3.9	145	yes	
Pitie	moto	30	w	NR	Imp	107	no	
HEGP	knife	16	m	98	2.5	118	no	
⋮								⋱

⇒ **Predict**: the risk of hemorrhagic shock given pre-hospital features

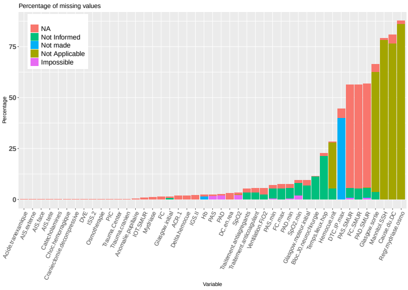
Ex random forests/logistic regression with covariates with missing values

Missing values



Multilevel data/ data integration: Systematic missing variable in one hospital

Complete-case analysis




```
?lm, ?glm, na.action = na.omit
```

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

An $n \times p$ matrix, each entry is missing with probability 0.01

$p = 5 \quad \implies \approx 95\%$ of rows kept

$p = 300 \implies \approx \ 5\%$ of rows kept

Missing values mechanisms

Dealing with missing values depends on the pattern of missing values and the **mechanism** leading to missing values (Rubin, 1976)

Ex: Two variables Income and Age with missing values on Income.

Missing Completely at Random (MCAR)

The probability to have missing values on income is independent of the values of age and the values of income. Each entry has the same probability to be observed.

Missing at Random (MAR)

The probability to have missing values on income depends on the values of age: older people are less encline to reveal their income

Missing not at Random (MNAR)

The probability to have missing values on income depends on the values of income: rich people are less encline to reveal their income

Missing values mechanisms

- $X \in \mathbb{R}^{n \times p}$ the data, $(X_{\text{obs}}, X_{\text{mis}})$ the observed and missing values,
- $M \in \mathbb{R}^{n \times p}$ the missing-data pattern:

$$M_{ij} = \begin{cases} 1 & \text{if } X_{ij} \text{ is observed,} \\ 0 & \text{otherwise.} \end{cases}$$

MCAR mechanism

$$g(M|X;\phi) = g(M;\phi), \quad \forall X, \phi.$$

ϕ: the unknown parameters of the missingness.

MAR mechanism

$$g(M|X;\phi) = g(M|X_{\text{obs}};\phi), \quad \forall X_{\text{mis}}, \phi.$$

MNAR mechanism

Other cases, i.e.

$$g(M|X;\phi) = g(M|X_{\text{obs}}, X_{\text{mis}};\phi), \quad \forall \phi.$$

Classical definitions

Missing value mechanisms (Rubin, 1976)

MCAR $\forall \phi, \forall \mathbf{m}, \mathbf{x}, g_\phi(\mathbf{m}|\mathbf{x}) = g_\phi(\mathbf{m})$

MAR $\forall \phi, \forall i, \forall \mathbf{x}', o(\mathbf{x}', \mathbf{m}_i) = o(\mathbf{x}_i, \mathbf{m}_i) \Rightarrow g_\phi(\mathbf{m}_i|\mathbf{x}') = g_\phi(\mathbf{m}_i|\mathbf{x}_i)$

(*e.g.* $g_\phi((0,0,1,0) \mid (3,2,4,8)) = g_\phi((0,0,1,0) \mid (3,2,7,8))$)

MNAR Not MAR

$\rightarrow$ useful for likelihoods

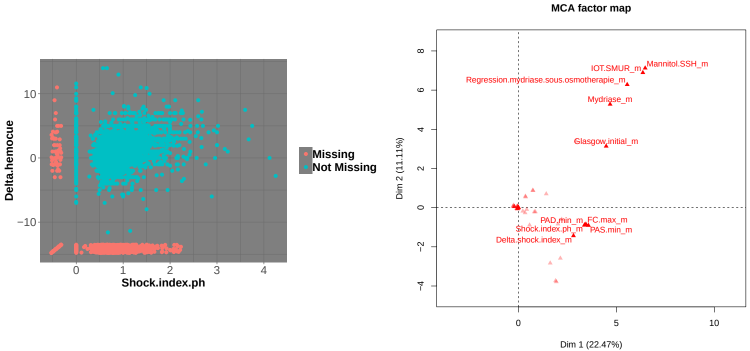
- $f_\theta(X)$, the distribution for the complete data
- $g_\phi(M|X)$, the missing values mechanism

$\Rightarrow$ Assume MAR: ignore $g_\phi(M|X)$ when doing (likelihood) inference on θ. Maximizing likelihood for observed data while ignoring (marginalizing) the unobserved values gives maximum likelihood estimates.

Visualization

The first thing to do with missing values (as for any analysis) is descriptive statistics: Visualization of patterns to get hints on how and why they occur

`VIM` (M. Templ), `naniar` (N. Tierney), `FactoMineR` (Husson *et al.*)



Right: *PAS_m* close to *PAD_m*: Often missing on both *PAS* & *PAD*

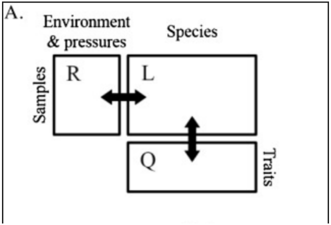
IOT: nested questions. Q1: yes/no, if yes Q2 - Q4, if no Q2 - Q4 "missing"

Note: Crucial **before** starting any treatment of missing values and **after**

Contingency tables with side information

National agency for wildlife and hunting management (ONCFS)

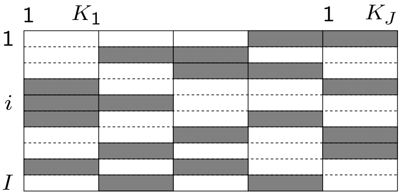
Data: Water-bird count data, 1990-2016 from 722 wetland sites in 5 countries in North Africa
Sites and years info: meteorological, geographical (altitude, long)



⇒ Aims: Assess the effect of time on species abundances
Monitor the population and assess wetlands conservation policies.

⇒ 70% of missing values in contingency tables

Multi-blocks data set

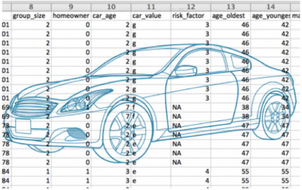


L'OREAL: 100 000 women in different countries - 300 questions

- Self-assessment questionnaire: life style, skin and hair characteristics, care and consumer habits
- Clinical assessments by a dermatologist: facial skin complexion, wrinkles, scalp dryness, greasiness
- Hair assessments by a hair dresser: abundance, volume, breakage, curliness
- Skin and Hair photographs and measurements: sebum quantity, etc.

You can use any type of data to create a Kaggle competition, for example:

- **Numerical**
- Text
- Media
- Multiple formats



Allstate ran a competition to predict a customer's purchase based on a limited amount of shopping history data.

airbnb

Jobs • 1,429 teams

Airbnb New User Bookings

Wed 25 Nov 2015 — Thu 11 Feb 2016 (40 hours to go)

Predict in which country a new user will make his first booking:

age: 42.4 %

date first booking: 6.7 %

first affiliate tracked: 2.2 %

gender: 46 %

Research works

- F. Husson (Agrocampus), G. Robin (PhD student), B. Narasimhan (Stanford): distributed matrix completion for multilevel medical data
- G. Robin, R. Tibshirani (Stanford): imputation of contingency tables with side information
- W. Jiang (PhD student), M. Lavielle (Inria), G. Bogdan (Wroclaw): glm with missing values and variable selection controlling FDR
- E. Scornet (X), Marine Le Morvan (Postdoc), G. Varoquaux (inria): random forest with missing values - MLP with missing values
- I. Mayer (PhD student), S. Wager (Stanford), J.P. Vert (Google Brain): Causal inference, deep-latent variables models with missing values

Solutions to handle missing values

Books: Schafer (2002), Little & Rubin (2002); Kim & Shao (2013); Carpenter & Kenward (2013); van Buuren (2018), etc.

Modify the estimation process to deal with missing values

Maximum likelihood: **EM algorithm** to obtain point estimates +
Supplemented EM (Meng & Rubin, 1991) / Louis formulae for their variability
Ex logistic regression: EM to get $\hat{\beta}$ + Louis to get $\hat{V}(\hat{\beta})$

Aim: **Estimate parameters & their variance** from an incomplete data
$\Rightarrow$ Inferential framework

Solutions to handle missing values

Books: Schafer (2002), Little & Rubin (2002); Kim & Shao (2013); Carpenter & Kenward (2013); van Buuren (2018), etc.

Modify the estimation process to deal with missing values

Maximum likelihood: **EM algorithm** to obtain point estimates +
Supplemented EM (Meng & Rubin, 1991) / Louis formulae for their variability
Ex logistic regression: EM to get $\hat{\beta}$ + Louis to get $\hat{V}(\hat{\beta})$
Cons: Difficult to establish - not many softwares even for simple models
One specific algorithm for each statistical method...

Aim: **Estimate parameters & their variance** from an incomplete data
$\Rightarrow$ Inferential framework

Solutions to handle missing values

Books: Schafer (2002), Little & Rubin (2002); Kim & Shao (2013); Carpenter & Kenward (2013); van Buuren (2018), etc.

Modify the estimation process to deal with missing values

Maximum likelihood: **EM algorithm** to obtain point estimates +
Supplemented EM (Meng & Rubin, 1991) / Louis formulae for their variability
Ex logistic regression: EM to get $\hat{\beta}$ + Louis to get $\hat{V}(\hat{\beta})$
Cons: Difficult to establish - not many softwares even for simple models
One specific algorithm for each statistical method...

Imputation (multiple) to get a complete data set

Any analysis can be performed
Ex logistic regression: Impute and apply logistic model to get $\hat{\beta}$, $\hat{V}(\hat{\beta})$

Aim: **Estimate parameters & their variance** from an incomplete data
$\Rightarrow$ Inferential framework

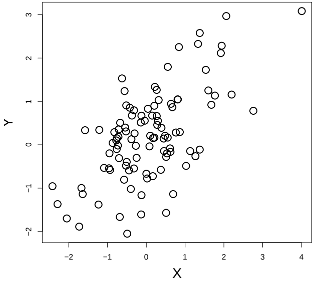
Outline

Outline

Mean imputation

- $(x_i, y_i) \underset{\text{i.i.d.}}{\sim} \mathcal{N}_2((\mu_x, \mu_y), \Sigma_{xy})$

X	Y
-0.56	-1.93
-0.86	-1.50
.....	...
2.16	0.7
0.16	0.74



$\mu_y = 0$

$\sigma_y = 1$

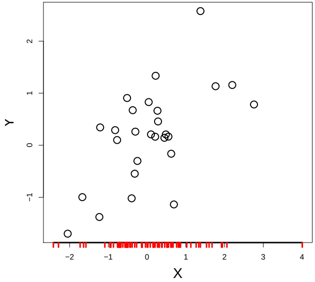
$\rho = 0.6$

$\hat{\mu}_y = -0.01$
$\hat{\sigma}_y = 1.01$
$\hat{\rho} = 0.66$

Mean imputation

- $(x_i, y_i) \underset{\text{i.i.d.}}{\sim} \mathcal{N}_2((\mu_x, \mu_y), \Sigma_{xy})$
- 70 % of missing entries completely at random on Y

X	Y
-0.56	NA
-0.86	NA
.....	...
2.16	0.7
0.16	NA



$\mu_y = 0$

$\sigma_y = 1$

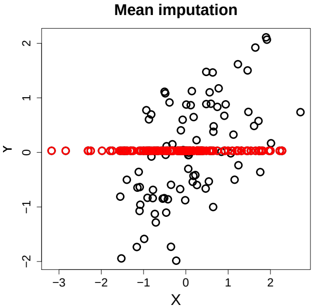
$\rho = 0.6$

$\hat{\mu}_y = 0.18$
$\hat{\sigma}_y = 0.9$
$\hat{\rho} = 0.6$

Mean imputation

- $(x_i, y_i) \underset{\text{i.i.d.}}{\sim} \mathcal{N}_2((\mu_x, \mu_y), \Sigma_{xy})$
- 70 % of missing entries completely at random on Y
- Estimate parameters on the mean imputed data

X	Y
-0.56	**0.01**
-0.86	**0.01**
.....	...
2.16	0.7
0.16	**0.01**



$\mu_y = 0$

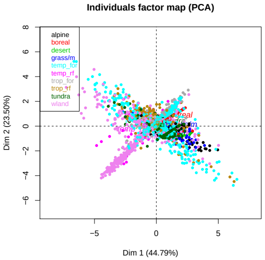
$\sigma_y = 1$

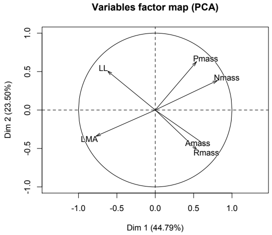
$\rho = 0.6$

$\hat{\mu}_y = 0.01$
$\hat{\sigma}_y = 0.5$
$\hat{\rho} = 0.30$

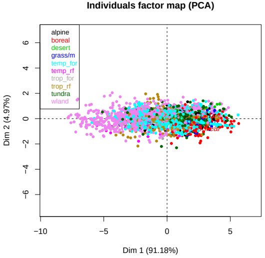
Mean imputation deforms joint and marginal distributions

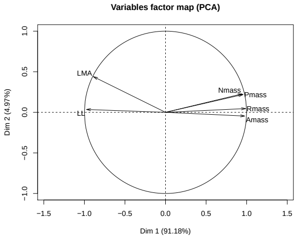
Mean imputation is bad for estimation






```
library(FactoMineR)
PCA(ecolo)
Warning message: Missing
are imputed by the mean
of the variable:
You should use imputePCA
from missMDA
```






```
library(missMDA)
imp <- imputePCA(ecolo)
PCA(imp$comp)
```

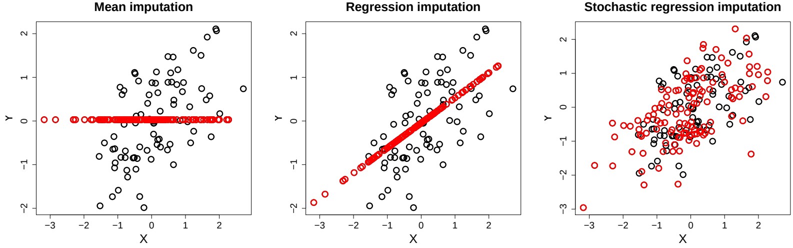
Ecological data: [1] $n = 69000$ species - 6 traits. Estimated correlation between Pmass & Rmass ≈ 0 (mean imputation) or ≈ 1 (EM PCA)

[1] Wright, I. et al. (2004). The worldwide leaf economics spectrum. *Nature*.

Imputation methods

- by regression takes into account the relationship: Estimate β - impute $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \Rightarrow$ variance underestimated and correlation overestimated
- by stochastic reg: Estimate β and σ - impute from the predictive $y_i \sim \mathcal{N}\left(x_i\hat{\beta}, \hat{\sigma}^2\right) \Rightarrow$ preserve distributions

Here $\hat{\beta}, \hat{\sigma}^2$ estimated with complete data, but MLE can be obtained with EM



	Mean imputation	Regression imputation	Stochastic regression imputation
$\mu_y = 0$	0.01	0.01	0.01
$\sigma_y = 1$	0.5	0.72	0.99
$\rho = 0.6$	0.30	0.78	0.59

Imputation with joint model with gaussian distribution

$\Rightarrow$ Hypothesis $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$
Bivariate case with missing values on $x_{.1}$ (stochastic regression):

- estimate β and σ
- impute from the predictive $y_i \sim \mathcal{N}\left(x_i\hat{\beta}, \hat{\sigma}^2\right)$

Extension to the multivariate case:

- Estimate μ and Σ from an incomplete data with EM
- Impute by drawing from the conditional distribution $X_{\text{MIS}}|X_{\text{OBS}} \sim \mathcal{N}(\mu_{\text{MIS}|\text{OBS}}, \Sigma_{\text{MIS}|\text{OBS}})$

$$\mu_{\text{MIS}|\text{OBS}} = \mathbb{E}[X_{\text{MIS}}] + \Sigma_{\text{MIS,OBS}}\Sigma^{-1}_{\text{OBS,OBS}}\left(X_{\text{OBS}} - \mathbb{E}[X_{\text{OBS}}]\right).$$

$\Rightarrow$ Corresponds to imputation by regression

$\Rightarrow$ Schur complements:

$$\Sigma_{\text{MIS}|\text{OBS}} = \Sigma_{\text{MIS}} - \Sigma_{\text{MIS,OBS}}\Sigma^{-1}_{\text{OBS,OBS}}\Sigma_{\text{OBS,MIS}}.$$

```
> pre <- prelim.norm(as.matrix(don))
> thetahat <- em.norm(pre)
> imp <- imp.norm(pre, thetahat, don)
```

Imputation methods for multivariate data

Assuming a joint model

- Gaussian distribution: $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$ (`Amelia` Honaker, King, Blackwell)
- low rank: $X_{n \times d} = \mu_{n \times d} + \varepsilon$ $\varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$ with μ of low rank k (`softimpute` Hastie & Mazuder; `missMDA` J. & Husson)
- latent class - nonparametric Bayesian (`dpmpm` Reiter)
- deep learning using variational autoencoders (MIWAE, Mattei, 2018)

Using conditional models (joint implicitly defined)

- with logistic, multinomial, poisson regressions (`mice` van Buuren)
- iterative impute each variable by random forests (`missForest` Stekhoven)

Imputation for categorical, mixed, blocks/multilevel data [2], etc.

⇒ Missing values taskview[3] J., Mayer., Tierney, Vialaneix

[2] J., Husson, Robin & Narasimhan. (2018). Imputation of mixed data with multilevel SVD.
[3] https://cran.r-project.org/web/views/MissingData.html

Outline

PCA (complete)

Find the subspace that best represents the data

Figure 1: Camel or dromedary?

⇒ Best approximation with projection

⇒ Best representation of the variability

⇒ Do not distort the distances between individuals

PCA (complete)

Find the subspace that best represents the data

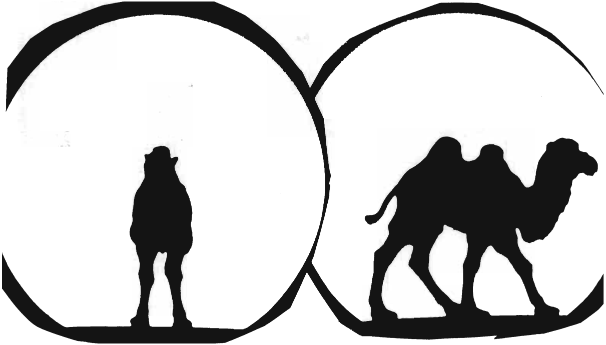


Figure 1: Camel or dromedary? source J.P. Fénelon

⇒ Best approximation with projection

⇒ Best representation of the variability

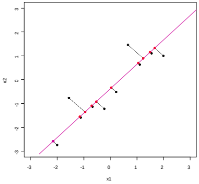
⇒ Do not distort the distances between individuals

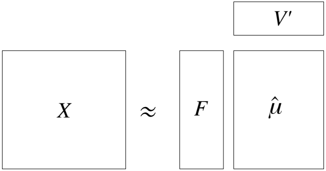
X

-2.00	-2.74
-1.56	-0.77
-1.11	-1.59
-0.67	-1.13
-0.22	-1.22
0.22	-0.52
0.67	1.46
1.11	0.63
1.56	1.10
2.00	1.00

$\hat{\mu}$

-2.16	-2.58
-0.96	-1.35
-1.15	-1.55
-0.70	-1.09
-0.53	-0.92
0.04	-0.34
1.24	0.89
1.05	0.69
1.50	1.15
1.67	1.33





⇒ Minimizes distance between observations and their projection

⇒ Approx $X_{n\times p}$ with a low rank matrix $S < p$ $\|A\|_2^2 = \mathrm{tr}(AA^\top)$:

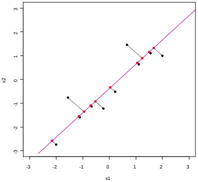
$$\operatorname{argmin}_\mu \left\{ \|X - \mu\|_2^2 : \operatorname{rank}(\mu) \leq S \right\}$$

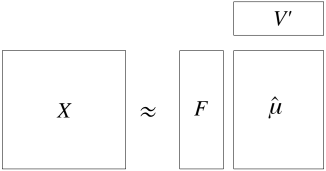
X

-2.00	-2.74
NA	-0.77
-1.11	-1.59
-0.67	-1.13
-0.22	NA
0.22	-0.52
0.67	1.46
NA	0.63
1.56	1.10
2.00	1.00

$\hat{\mu}$

-2.16	-2.58
-0.96	-1.35
-1.15	-1.55
-0.70	-1.09
-0.53	-0.92
0.04	-0.34
1.24	0.89
1.05	0.69
1.50	1.15
1.67	1.33





⇒ Minimizes distance between observations and their projection

⇒ Approx $X_{n\times p}$ with a low rank matrix $S < p$ $\|A\|_2^2 = \text{tr}(AA^\top)$:

$$\text{argmin}_\mu \left\{ \|X - \mu\|_2^2 : \text{rank}(\mu) \leq S \right\}$$

SVD X: $\hat{\mu}^{\text{PCA}} = U_{n\times S}\Lambda^{\frac{1}{2}}_{S\times S}V'_{p\times S}$ $\quad F = U\Lambda^{\frac{1}{2}}$ PC - scores

$= F_{n\times S}V'_{p\times S}$ $\quad V$ principal axes - loadings

Missing values in PCA

$\Rightarrow$ PCA: least squares

$$\text{argmin}_\mu \left\{ \|X_{n\times p} - \mu_{n\times p}\|_2^2 : \text{rank}\,(\mu) \leq S \right\}$$

$\Rightarrow$ PCA with missing values: weighted least squares

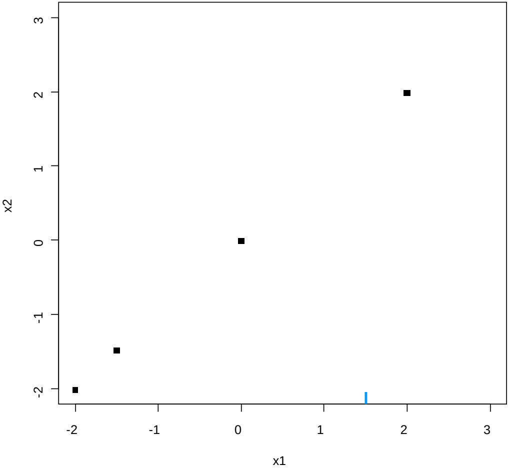
$$\text{argmin}_\mu \left\{ \|W_{n\times p} * (X - \mu)\|_2^2 : \text{rank}\,(\mu) \leq S \right\}$$

with $W_{ij} = 0$ if X_{ij} is missing, $W_{ij} = 1$ otherwise; $*$ elementwise multiplication

Many algorithms: weighted alternating least squares (Gabriel & Zamir, 1979); iterative PCA (Kiers, 1997)

Iterative PCA

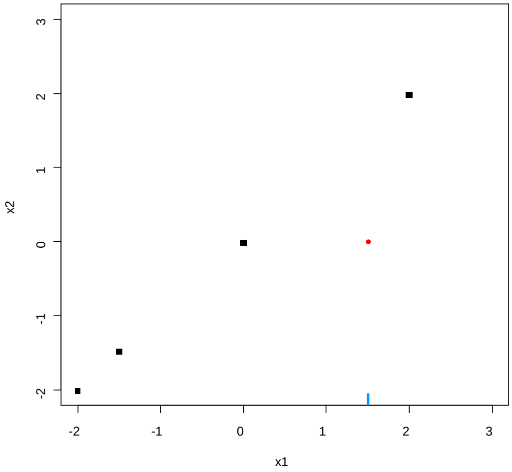
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**NA**
2.0	1.98



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



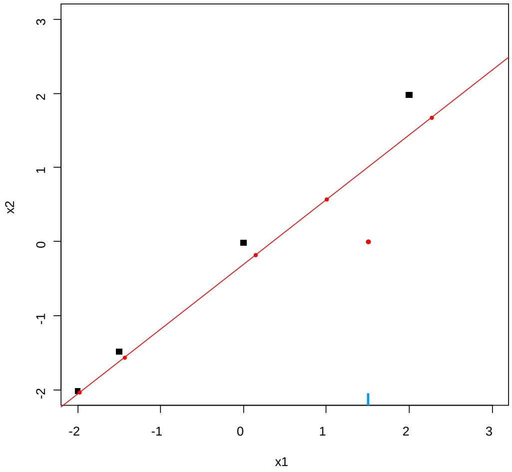
Initialization $\ell = 0$: X^0 (mean imputation)

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

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

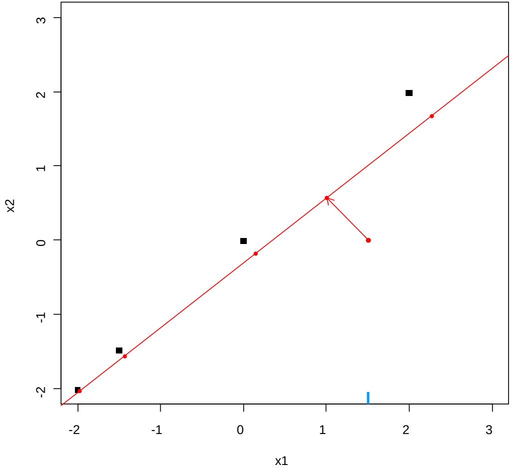


PCA on the completed data set $\rightarrow (U^{\ell}, \Lambda^{\ell}, V^{\ell})$;

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



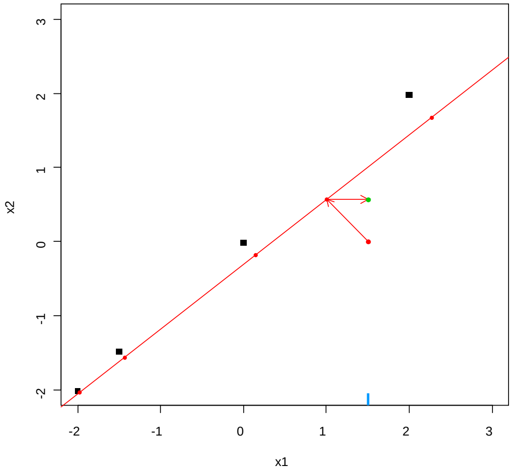
Missing values imputed with the fitted matrix $\hat{\mu}^{\ell} = U^{\ell} \Lambda^{1/2^{\ell}} V^{\ell\prime}$

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

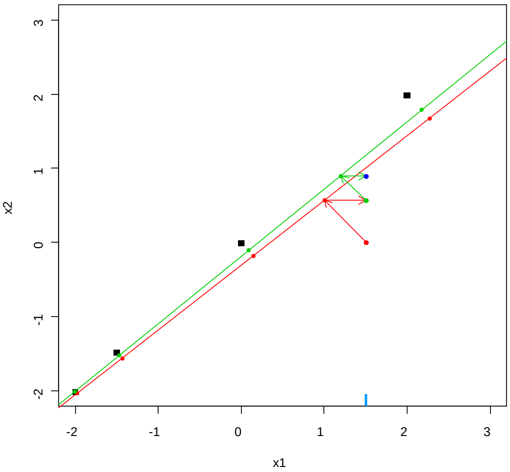


The new imputed dataset is $\hat{X}^{\ell} = W * X + (\mathbf{1} - W) * \hat{\mu}^{\ell}$

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



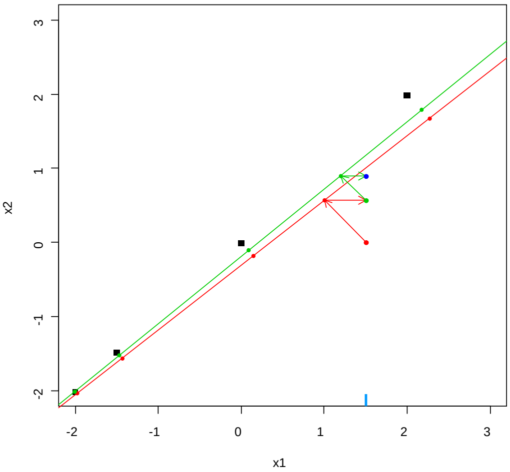
Iterative PCA

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	**0.57**
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



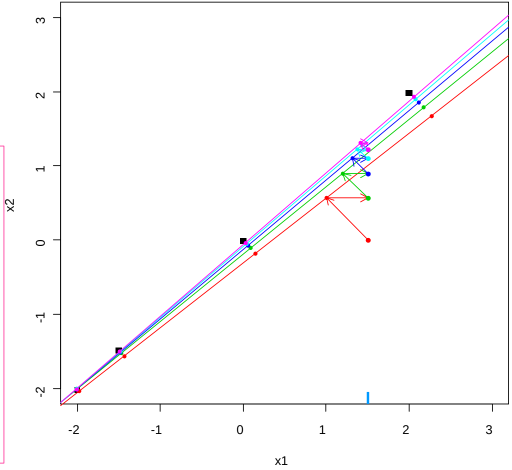
Iterative PCA

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

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

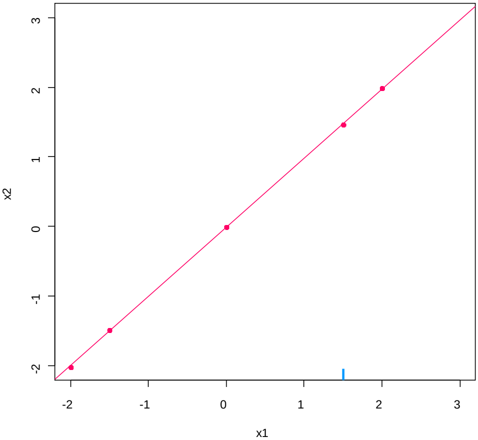
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98



Steps are repeated until convergence

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	**1.46**
2.0	1.98

PCA on the completed data set $\rightarrow (U^\ell, \Lambda^\ell, V^\ell)$

Missing values imputed with the fitted matrix $\hat{\mu}^\ell = U^\ell \Lambda^{1/2^\ell} V^{\ell\prime}$

Iterative PCA

1. initialization $\ell = 0$: X^0 (mean imputation)
2. step ℓ:
   (a) PCA on the completed data $\rightarrow (U^\ell, \Lambda^\ell, V^\ell)$;
   S dimensions kept
   (b) missing values are imputed with $(\hat{\mu}^S)^\ell = U^\ell {\Lambda^{1/2}}^\ell V^{\ell\prime}$
   the new imputed data is $\hat{X}^\ell = W * X + (\mathbf{1} - W) * (\hat{\mu}^S)^\ell$
3. steps of estimation and imputation are repeated

Iterative PCA

1. initialization $\ell = 0$: X^0 (mean imputation)
2. step ℓ:
   (a) PCA on the completed data $\rightarrow (U^\ell, \Lambda^\ell, V^\ell)$;
   S dimensions kept
   (b) missing values are imputed with $(\hat{\mu}^S)^\ell = U^\ell {\Lambda^{1/2}}^\ell V^{\ell\prime}$
   the new imputed data is $\hat{X}^\ell = W * X + (\mathbf{1} - W) * (\hat{\mu}^S)^\ell$
3. steps of estimation and imputation are repeated

$\Rightarrow$ $\hat{\mu}$ from incomplete data: EM algo $X = \mu + \varepsilon$, $\varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma^2\right)$
with μ of low rank , $x_{ij} = \sum_{s=1}^{S} \sqrt{\tilde{\lambda}_s} \tilde{u}_{is} \tilde{v}_{js} + \varepsilon_{ij}$

$\Rightarrow$ Completed data: good imputation (matrix completion, Netflix)

Iterative PCA

1. initialization $\ell = 0$: X^0 (mean imputation)
2. step ℓ:
   (a) PCA on the completed data $\rightarrow (U^\ell, \Lambda^\ell, V^\ell)$;
   S dimensions kept
   (b) missing values are imputed with $(\hat{\mu}^S)^\ell = U^\ell {\Lambda^{1/2}}^\ell V^{\ell\prime}$
   the new imputed data is $\hat{X}^\ell = W * X + (\mathbf{1} - W) * (\hat{\mu}^S)^\ell$
3. steps of estimation and imputation are repeated

$\Rightarrow$ $\hat{\mu}$ from incomplete data: EM algo $X = \mu + \varepsilon$, $\varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma^2\right)$
with μ of low rank , $x_{ij} = \sum_{s=1}^{S} \sqrt{\tilde{\lambda}_s} \tilde{u}_{is} \tilde{v}_{js} + \varepsilon_{ij}$

$\Rightarrow$ Completed data: good imputation (matrix completion, Netflix)

Reduction of variability (imputation by $U\Lambda^{1/2}V'$)

Selecting S? Generalized cross-validation (J. & Husson, 2012)

Cross-validation to select S

$\Rightarrow$ EM-CV (Bro *et al.* 2008)
$\Rightarrow$ $\text{MSEP}_S = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}(X_{ij} - (\hat{\mu}_{ij}^{S})^{-ij})^2$
$\Rightarrow$ Computational costly

Cross-validation to select S

$\Rightarrow$ EM-CV (Bro *et al.* 2008)

$\Rightarrow$ $\text{MSEP}_S = \frac{1}{np} \sum_{i=1}^{n} \sum_{j=1}^{p} (X_{ij} - (\hat{\mu}_{ij}^{S})^{-ij})^2$

$\Rightarrow$ Computational costly

Cross-validation to select S

$\Rightarrow$ EM-CV (Bro *et al.* 2008)

$\Rightarrow \text{MSEP}_S = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}(X_{ij} - (\hat{\mu}_{ij}^{S})^{-ij})^2$

$\Rightarrow$ Computational costly

Cross-validation to select S

$\Rightarrow$ EM-CV (Bro *et al.* 2008)

$\Rightarrow$ $\text{MSEP}_S = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}(X_{ij} - (\hat{\mu}_{ij}^{S})^{-ij})^2$

$\Rightarrow$ Computational costly

$\Rightarrow$ In regression $\hat{y} = Py$ (Craven & Whaba, 1979)

$$\hat{y}_i^{-i} - y_i = \frac{\hat{y}_i - y_i}{1 - P_{i,i}}$$

Cross-validation to select S

$\Rightarrow$ EM-CV (Bro *et al.* 2008)
$\Rightarrow$ $\text{MSEP}_S = \frac{1}{np}\sum_{i=1}^{n}\sum_{j=1}^{p}(X_{ij} - (\hat{\mu}_{ij}^{S})^{-ij})^2$
$\Rightarrow$ Computational costly

$\Rightarrow$ In regression $\hat{y} = Py$ (Craven & Whaba, 1979)

$$\hat{y}_i^{-i} - y_i = \frac{\hat{y}_i - y_i}{1 - P_{i,i}}$$

$\Rightarrow$ Aim: write PCA as $\hat{\mu}^{(S)} = PX$

$$(\hat{\mu}_{ij}^{S})^{-ij} - x_{ij} \simeq \frac{(\hat{\mu}_{ij}^{S}) - X_{ij}}{1 - P_{ij,ij}}$$

PCA as a smoother

2 projection matrices: $\|X_{n\times p} - F_{n\times S} V'_{S\times p}\|_2^2$

$$\begin{cases} V' = (F'F)^{-1}F'X & \Rightarrow P_F = F(F'F)^{-1}F' \\ F = XV(V'V)^{-1} & \Rightarrow P_V = V(V'V)^{-1}V' \end{cases}$$

$$\hat{\mu}^S = FV' = XP_V = P_F X$$

$$\mathrm{vec}(\hat{\mu}^{(S)}) = P^{(S)}\mathrm{vec}(X) \quad P^{(S)}_{np\times np} = (P_V^{'} \otimes \mathbb{I}_n) + (\mathbb{I}_p^{'} \otimes P_F) - (P_V^{'} \otimes P_F)$$

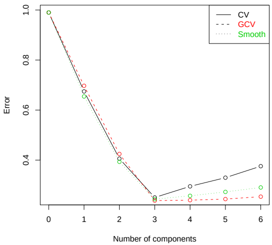
Pazman & Denis, 2002; Candes & Tao, 2009

$\Rightarrow$ Number of independent parameters:

$$\hat{\sigma}^2 = \frac{RSS}{\mathrm{tr}\left(\mathbb{I}_{np} - P^{(S)}\right)} = \frac{n\sum_{s=S+1}^{min(n,p)} \lambda_s}{np - (nS + pS - S^2)}$$

Cross-validation approximations

```
> nb <- estim_ncp(don)
> nb$criterion
        0         1         2         3         4         5
1.2884873 0.8069719 0.6400517 0.7045074 2.2257738 3.0274337
```



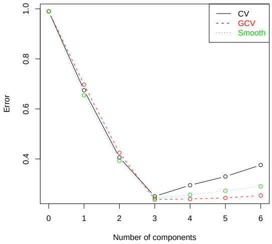
$$\mathrm{CV}_S = \frac{1}{np}\sum_{i,j}\left(X_{ij} - (\hat{\mu}_{ij}^{S})^{-ij}\right)^2$$

$$\mathrm{ACV}_S = \frac{1}{np}\sum_{i,j}\left(\frac{X_{ij} - (\hat{\mu}_{ij}^{S})}{1 - P_{ij,ij}}\right)^2$$

Josse, J. & Husson, F. Selecting the number of components in PCA using cross-validation approximations. *Computational Statistics and Data Analysis*.

Cross-validation approximations

```
> nb <- estim_ncp(don)
> nb$criterion
        0         1         2         3         4         5
1.2884873 0.8069719 0.6400517 0.7045074 2.2257738 3.0274337
```



$$\mathrm{CV}_S = \frac{1}{np}\sum_{i,j}\left(X_{ij} - (\hat{\mu}_{ij}^S)^{-ij}\right)^2$$

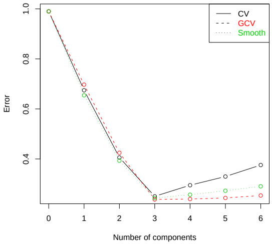
$$\mathrm{ACV}_S = \frac{1}{np}\sum_{i,j}\left(\frac{X_{ij} - (\hat{\mu}_{ij}^S)}{1 - P_{ij,ij}}\right)^2$$

$$\mathrm{GCV}_S = \frac{1}{np} \times \frac{\sum_{i,j}(X_{ij} - \hat{\mu}_{ij}^S))^2}{(1 - \mathrm{tr}(P^{(S)})/np)^2}$$

Josse, J. & Husson, F. Selecting the number of components in PCA using cross-validation approximations. *Computational Statistics and Data Analysis*.

Cross-validation approximations

```
> nb <- estim_ncp(don)
> nb$criterion
        0         1         2         3         4         5
1.2884873 0.8069719 0.6400517 0.7045074 2.2257738 3.0274337
```



$$\text{CV}_S = \frac{1}{np}\sum_{i,j}\left(X_{ij} - (\hat{\mu}_{ij}^S)^{-ij}\right)^2$$

$$\text{ACV}_S = \frac{1}{np}\sum_{i,j}\left(\frac{X_{ij} - (\hat{\mu}_{ij}^S)}{1 - P_{ij,ij}}\right)^2$$

$$\text{GCV}_S = \frac{np\sum_{i,j}(X_{ij} - \hat{\mu}_{ij}^S))^2}{(np - \text{tr}(P^{(S)}))^2}$$

$$\text{GCV NA}_S = \frac{np\|W * (X - \hat{\mu}^S))\|_2^2}{(np - |NA| - (nS + pS - S^2))^2}$$

Josse, J. & Husson, F. Selecting the number of components in PCA using cross-validation approximations. *Computational Statistics and Data Analysis*.

Overfitting

Overfitting when:

- many parameters / the number of observed values (the number of dimensions S and of missing values are important)
- data are very noisy

$\Rightarrow$ Trust too much the relationship between variables

Remarks:

- missing values: special case of small data set
- iterative PCA: prediction method

Solution:
$\Rightarrow$ Shrinkage methods

Soft thresholding iterative SVD

⇒ Overfitting issues of iterative PCA: many parameters ($U_{n\times S}$, $V_{S\times p}$)/observed values (S large - many NA); noisy data

⇒ Regularized versions. Init - estimation - imputation steps:

imputation $\hat{\mu}_{ij}^{\text{PCA}} = \sum_{s=1}^{S} \sqrt{\lambda_s} u_{is} v_{js}$ is replaced by

a "shrunk" impute $\hat{\mu}_{ij}^{\text{Soft}} = \sum_{s=1}^{p} \left(\sqrt{\lambda_s} - \lambda\right)_+ u_{is} v_{js}$

$$X = \mu + \varepsilon \qquad \text{argmin}_\mu \left\{ \|W * (X - \mu)\|_2^2 + \lambda \|\mu\|_* \right\}$$

SoftImpute for large matrices. T. Hastie, R. Mazumber, 2015, Matrix Completion and Low-Rank SVD via Fast Alternating Least Squares. *JMLR* Implemented in `softImpute`

Regularized iterative PCA

⇒ Init. - estimation - imputation steps. In missMDA (Youtube)

The imputation step:

$$\hat{\mu}_{ij}^{\text{PCA}} = \sum_{s=1}^{S} \sqrt{\lambda_s} u_{is} v_{js}$$

is replaced by a "shrunk" imputation step (Efron & Morris 1972):

$$\hat{\mu}_{ij}^{\text{rPCA}} = \sum_{s=1}^{S} \left(\frac{\lambda_s - \hat{\sigma}^2}{\lambda_s} \right) \sqrt{\lambda_s} u_{is} v_{js} = \sum_{s=1}^{S} \left(\sqrt{\lambda_s} - \frac{\hat{\sigma}^2}{\sqrt{\lambda_s}} \right) u_{is} v_{js}$$

σ^2 small → regularized PCA ≈ PCA
σ^2 large → mean imputation

$$\hat{\sigma}^2 = \frac{RSS}{\text{ddl}} = \frac{n \sum_{s=S+1}^{p} \lambda_s}{np - p - nS - pS + S^2 + S} \qquad (X_{n \times p}; U_{n \times S}; V_{p \times S})$$

Properties

$\Rightarrow$ Results of PCA obtained from an incomplete data set: graph of observations and correlation circle. Missing values are skipped $||W * (X - \mu)||^2$

$\Rightarrow$ Very good quality of imputation. Using similarities between individuals and relationship between variables. Popular in machine learning with recommandation systems (Netflix: 99% missing).

Model makes sense: Data = structure of rank S + noise

(Udell & Townsend Nice Latent Variable Models Have Log-Rank, 2017)

$\Rightarrow$ Different noise regime

- low noise: iterative PCA (tuning S: cross-validation, GCV)
- moderate: iterative regularized PCA (tuning σ, S)
- high noise (SNR low, S large): soft thresholding (tuning λ, σ)
  Implemented in R packages `denoiseR` (Josse, Wager, Sardy)

The imputed data set should be analysed with caution with other methods

Random Forests versus PCA

	Feat1	Feat2	Feat3	Feat4	Feat5...
C1	1	1	1	1	1
C2	1	1	1	1	1
C3	2	2	2	2	2
C4	2	2	2	2	2
C5	3	3	3	3	3
C6	3	3	3	3	3
C7	4	4	4	4	4
C8	4	4	4	4	4
C9	5	5	5	5	5
C10	5	5	5	5	5
C11	6	6	6	6	6
C12	6	6	6	6	6
C13	7	7	7	7	7
C14	7	7	7	7	7
Igor	8	NA	NA	8	8
Frank	8	NA	NA	8	8
Bertrand	9	NA	NA	9	9
Alex	9	NA	NA	9	9
Yohann	10	NA	NA	10	10
Jean	10	NA	NA	10	10

Iterative Random Forests imputation

1. Initial imputation: mean imputation - random category
   Sort the variables according to the amount of missing values
2. Fit a RF X_j^{obs} on variables X_{-j}^{obs} and then predict X_j^{miss}
3. Cycling through variables
4. Repeat step 2.2 and 3 until convergence

- number of trees: 100
- number of variables randomly selected at each node $\sqrt{p}$
- number of iterations: 4-5

Implemented in the R package `missForest` (paper) `missForest` (Daniel J. Stekhoven, Peter Buhlmann, 2011)

Random forests versus PCA

	Feat1	Feat2	Feat3	Feat4	Feat5...
C1	1	1	1	1	1
C2	1	1	1	1	1
C3	2	2	2	2	2
C4	2	2	2	2	2
C5	3	3	3	3	3
C6	3	3	3	3	3
C7	4	4	4	4	4
C8	4	4	4	4	4
C9	5	5	5	5	5
C10	5	5	5	5	5
C11	6	6	6	6	6
C12	6	6	6	6	6
C13	7	7	7	7	7
C14	7	7	7	7	7
Igor	8	NA	NA	8	8
Frank	8	NA	NA	8	8
Bertrand	9	NA	NA	9	9
Alex	9	NA	NA	9	9
Yohann	10	NA	NA	10	10
Jean	10	NA	NA	10	10

Missing

Feat1	Feat2	Feat3	Feat4	Feat5
1	1.0	1.00	1	1
1	1.0	1.00	1	1
2	2.0	2.00	2	2
2	2.0	2.00	2	2
3	3.0	3.00	3	3
3	3.0	3.00	3	3
4	4.0	4.00	4	4
4	4.0	4.00	4	4
5	5.0	5.00	5	5
5	5.0	5.00	5	5
6	6.0	6.00	6	6
6	6.0	6.00	6	6
7	7.0	7.00	7	7
7	7.0	7.00	7	7
8	6.87	6.87	8	8
8	6.87	6.87	8	8
9	6.87	6.87	9	9
9	6.87	6.87	9	9
10	6.87	6.87	10	10
10	6.87	6.87	10	10

`missForest`

Feat1	Feat2	Feat3	Feat4	Feat5
1	1	1	1	1
1	1	1	1	1
2	2	2	2	2
2	2	2	2	2
3	3	3	3	3
3	3	3	3	3
4	4	4	4	4
4	4	4	4	4
5	5	5	5	5
5	5	5	5	5
6	6	6	6	6
6	6	6	6	6
7	7	7	7	7
7	7	7	7	7
8	8	8	8	8
8	8	8	8	8
9	9	9	9	9
9	9	9	9	9
10	10	10	10	10
10	10	10	10	10

`imputePCA`

$\Rightarrow$ Imputation inherits from the method: RF (computationaly costly) good for non linear relationships / PCA good for linear relationships

Outline

Incomplete ozone

	O3	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	O3v
0601	87	15.6	18.5	18.4	4	4	8	NA	-1.7101	-0.6946	84
0602	82	NA	18.4	17.7	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
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
0919	NA	14.8	16.3	15.9	7	7	7	-4.3301	-6.0622	-5.1962	42
0920	71	15.5	18	17.4	7	7	6	-3.9392	-3.0642	0	NA
0921	96	NA	NA	NA	3	3	3	NA	NA	NA	71
0922	98	NA	NA	NA	2	2	2	4	5	4.3301	96
0923	92	14.7	17.6	18.2	1	4	6	5.1962	5.1423	3.5	98
0924	NA	13.3	17.7	17.7	NA	NA	NA	-0.9397	-0.766	-0.5	92
0925	84	13.3	17.7	17.8	3	5	6	0	-1	-1.2856	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

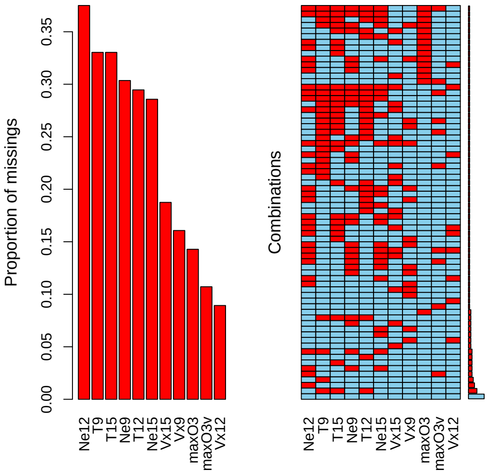
```
        maxO3    T9   T12   T15   Ne9  Ne12  Ne15    Vx9   Vx12   Vx15  maxO3v
20010601  87.000 15.600 18.500 20.471 4.000 4.000 8.000  0.695 -1.710 -0.695  84.000
20010602  82.000 18.505 20.870 21.799 5.000 5.000 7.000 -4.330 -4.000 -3.000  87.000
20010603  92.000 15.300 17.600 19.500 2.000 3.984 3.812  2.954  1.951  0.521  82.000
20010604 114.000 16.200 19.700 24.693 1.000 1.000 0.000  2.044  0.347 -0.174  92.000
20010605  94.000 18.968 20.500 20.400 5.294 5.272 5.056 -0.500 -2.954 -4.330 114.000
20010606  80.000 17.700 19.800 18.300 6.000 7.020 7.000 -5.638 -5.000 -6.000  94.000
20010607  79.000 16.800 15.600 14.900 7.000 8.000 6.556 -4.330 -1.879 -3.759  80.000
20010610  79.000 14.900 17.500 18.900 5.000 5.000 5.016  0.000 -1.042 -1.389  99.000
20010611 101.000 16.100 19.600 21.400 2.000 4.691 4.000 -0.766 -1.026 -2.298  79.000
20010612 106.000 18.300 22.494 22.900 5.000 4.627 4.495  1.286 -2.298 -3.939 101.000
20010613 101.000 17.300 19.300 20.200 7.000 7.000 3.000 -1.500 -1.500 -0.868 106.000
.....

20010915  69.000 17.100 17.700 17.500 6.000 7.000 8.000 -5.196 -2.736 -1.042  71.000
20010916  71.000 15.400 18.091 16.600 4.000 5.000 5.000 -3.830  0.000  1.389  69.000
20010917  60.000 15.283 18.565 19.556 4.000 5.000 4.000  0.000  3.214  0.000  71.000
20010918  42.000 14.091 14.300 14.900 8.000 7.000 7.000 -2.500 -3.214 -2.500  60.000
20010919  65.000 14.800 16.425 15.900 7.000 7.982 7.000 -4.341 -6.062 -5.196  42.000
20010920  71.000 15.500 18.000 17.400 7.000 7.000 6.000 -3.939 -3.064  0.000  65.000
20010924  76.000 13.300 17.700 17.700 5.631 5.883 5.453 -0.940 -0.766 -0.500  65.139
20010925  75.573 13.300 18.434 17.800 3.000 5.000 5.001  0.000 -1.000 -1.286  76.000
20010927  77.000 16.200 20.800 20.499 5.368 5.495 5.177 -0.695 -2.000 -1.473  71.000
20010928  99.000 18.074 22.169 23.651 3.531 3.610 3.561  1.500  0.868  0.868  93.135
20010929  83.000 19.855 22.663 23.847 5.374 5.000 3.000 -4.000 -3.759 -4.000  99.000
20010930  70.000 15.700 18.600 20.700 7.000 6.405 7.000 -2.584 -1.042 -4.000  83.000
```

```
> library(missMDA)
> res.comp <- imputePCA(ozo[, 1:11])
> res.comp$comp
```

Count missing values

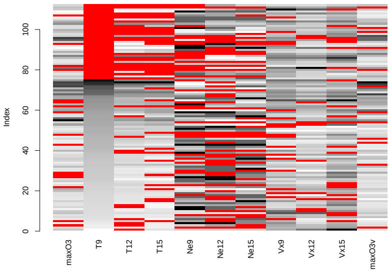
```
> library(missMDA)
> WindDirection <- ozo[,12]
> don <- ozo[,1:11]
> library(VIM)
> res <- summary(aggr(don, sortVar = TRUE))$combinations
> res[rev(order(res[, 2])),]

Variables sorted by
number of missings:                       Combinations Count    Percent
Variable      Count         0:0:0:0:0:0:0:0:0:0:0    13 11.6071429
    Ne12 0.37500000         0:1:1:1:0:0:0:0:0:0:0     7  6.2500000
      T9 0.33035714         0:0:0:0:0:1:0:0:0:0:0     5  4.4642857
     T15 0.33035714         0:1:0:0:0:0:0:0:0:0:0     4  3.5714286
     Ne9 0.30357143         0:1:0:0:1:1:1:0:0:0:0     3  2.6785714
     T12 0.29464286         0:0:1:0:0:0:0:0:0:0:0     3  2.6785714
    Ne15 0.28571429         0:0:0:1:0:0:0:0:0:0:0     3  2.6785714
    Vx15 0.18750000         0:0:0:0:1:1:1:0:0:0:0     3  2.6785714
     Vx9 0.16071429         0:0:0:0:0:1:0:0:0:0:1     3  2.6785714
   maxO3 0.14285714         0:1:1:1:1:0:0:0:0:0:0     2  1.7857143
  maxO3v 0.10714286         0:0:0:0:1:0:0:0:0:1:0     2  1.7857143
    Vx12 0.08928571         0:0:0:0:0:0:1:1:0:0:0     2  1.7857143
                            0:0:0:0:0:0:1:0:0:0:0     2  1.7857143
                            .....................     .  ...
```

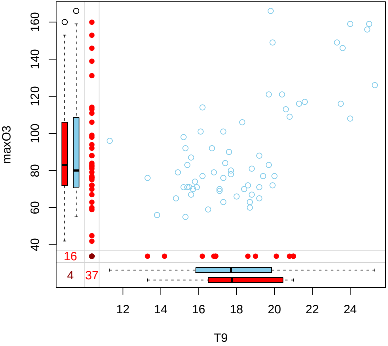
Pattern visualization




```
 #library(VIM)
> aggr(don, sortVar = TRUE)
```

Visualization





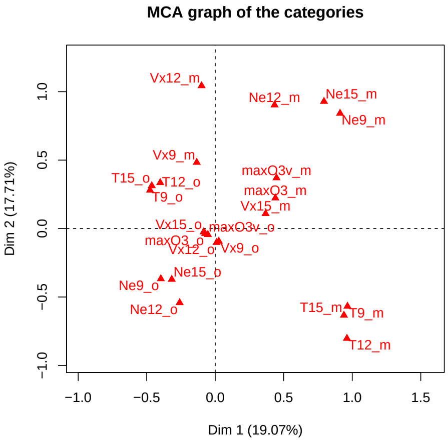

```
# library(VIM)
> matrixplot(don, sortby = 2)
> marginplot(don[ ,c("T9", "maxO3")])
```

Visualization with Multiple Correspondence Analysis

⇒ Create the missingness matrix

```
> mis.ind <- matrix("o", nrow = nrow(don), ncol = ncol(don))
> mis.ind[is.na(don)] = "m"
> dimnames(mis.ind) = dimnames(don)
> mis.ind

         maxO3 T9  T12 T15 Ne9 Ne12 Ne15 Vx9 Vx12 Vx15 maxO3v
20010601 "o"   "o" "o" "m" "o" "o"  "o"  "o" "o"  "o"  "o"
20010602 "o"   "m" "m" "m" "o" "o"  "o"  "o" "o"  "o"  "o"
20010603 "o"   "o" "o" "o" "o" "m"  "m"  "o" "m"  "o"  "o"
20010604 "o"   "o" "o" "m" "o" "o"  "o"  "m" "o"  "o"  "o"
20010605 "o"   "m" "o" "o" "m" "m"  "m"  "o" "o"  "o"  "o"
20010606 "o"   "o" "o" "o" "o" "m"  "o"  "o" "o"  "o"  "o"
20010607 "o"   "o" "o" "o" "o" "o"  "m"  "o" "o"  "o"  "o"
20010610 "o"   "o" "o" "o" "o" "o"  "m"  "o" "o"  "o"  "o"
```

Visualization with Multiple Correspondence Analysis

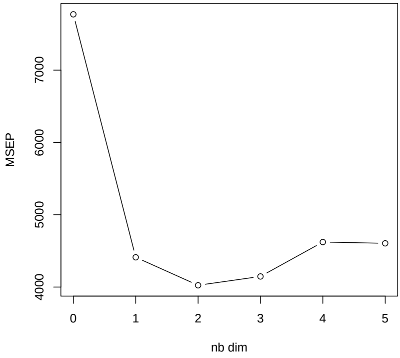



```
> library(FactoMineR)
> resMCA <- MCA(mis.ind)
> plot(resMCA, invis = "ind", title = "MCA graph of the categories")
```

Imputation with PCA in practice

⇒ Step 1: Estimation of the number of dimensions
(Cross Validation, 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")
```

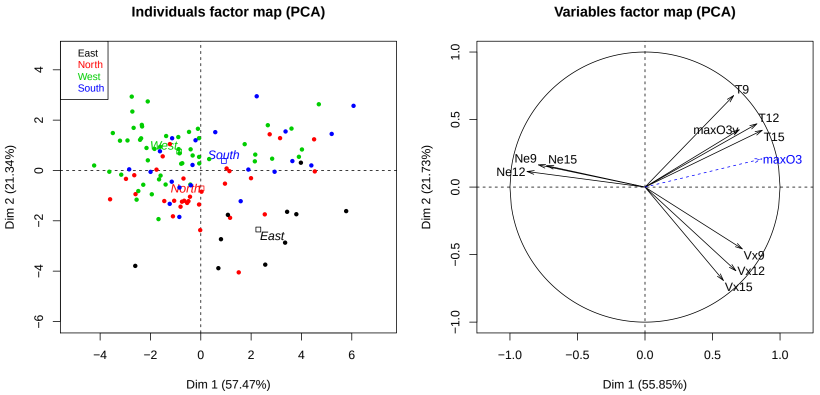


Imputation with PCA in practice

⇒ Step 2: Imputation of the missing values

```
> res.comp <- imputePCA(don, ncp = 2)
> res.comp$completeObs[1:3, ]
     maxO3    T9   T12   T15 Ne9 Ne12 Ne15   Vx9  Vx12  Vx15 maxO3v
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
```

Cherry on the cake: PCA on incomplete data!




```
> imp <- cbind.data.frame(res.comp$completeObs, ozo[, 12])
> res.pca <- PCA(imp, quanti.sup = 1, quali.sup = 12)
> plot(res.pca, hab = 12, lab = "quali"); plot(res.pca, choix = "var")
> res.pca$ind$coord #scores (principal components)
```

Imputation for continuous data

```
> library(softImpute)
> fit1 <- softImpute(XNA, rank = , lambda = )
> X.soft <- complete(XNA, fit1)

> library(denoiseR)
> adaNA <- imputeada(XNA,  gamma = 1) ## time consuming...
> X.ada <- adaNA$completeObs
```

An ecological data set

Glopnet data: 2494 species described by 6 quantitative variables

- LMA (leaf mass per area)
- LL (leaf lifespan)
- Amass (photosynthetic assimilation)
- Nmass (leaf nitrogen),
- Pmass (leaf phosphorus)
- Rmass (dark respiration rate)

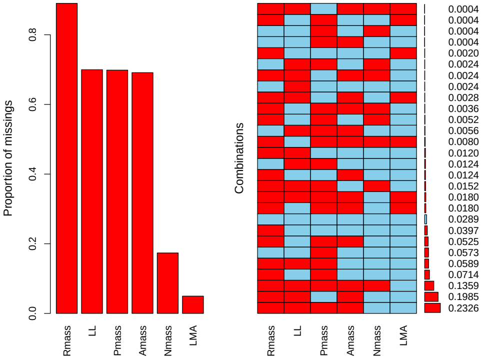
and 1 categorical variable: the biome

Reference: Wright IJ, et al. (2004) The worldwide leaf economics spectrum. Nature, 428:821.
`www.nature.com/nature/journal/v428/n6985/extref/nature02403-s2.xls`

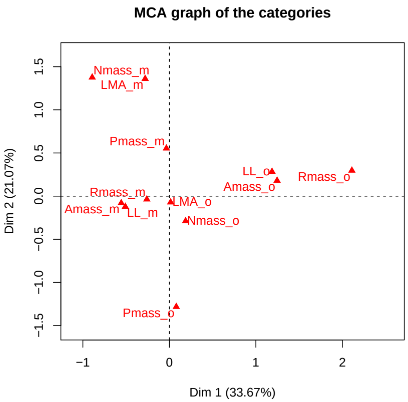
An ecological data set

```
> sum(is.na(don))/(nrow(don)*ncol(don)) # 53% of missing values
[1] 0.5338145
> dim(na.omit(don))  ## Delete species with missing values
[1] 72  6            ## only 72 remaining species!
```

```
> library(VIM)
> aggr(don,numbers=TRUE,sortVar=TRUE)
```



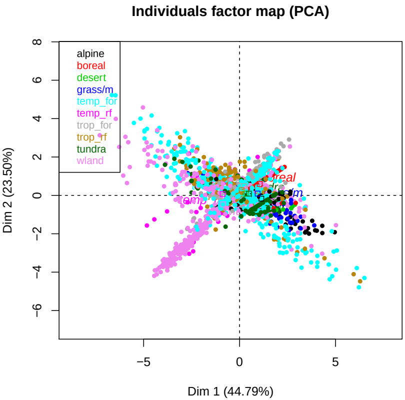
An ecological data set

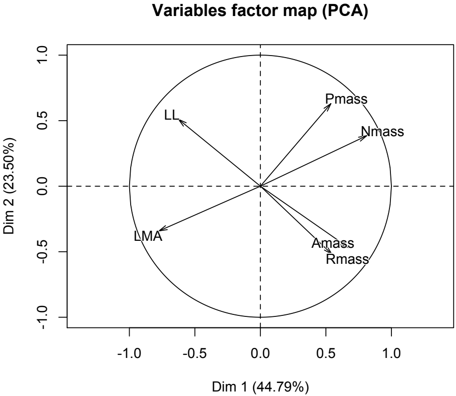



```
> mis.ind <- matrix("o",nrow=nrow(don),ncol=ncol(don))
> mis.ind[is.na(don)] <- "m"
> dimnames(mis.ind) <- dimnames(don)
> library(FactoMineR)
> resMCA <- MCA(mis.ind)
> plot(resMCA,invis="ind",title="MCA graph of the categories")
```

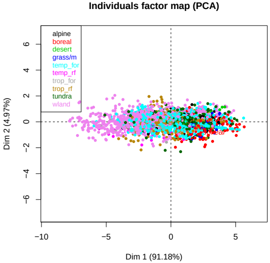
An ecological data set

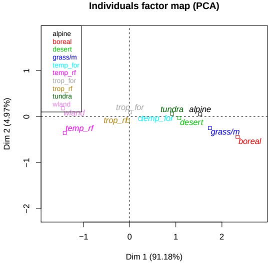
What about mean imputation?

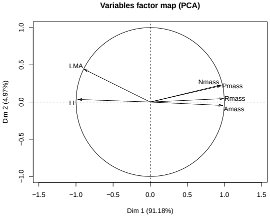




An ecological data set





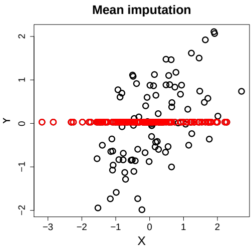



```
> library(missMDA)
> nb <- estim_ncpPCA(don,method.cv="Kfold",nbsim=100)
> res.comp <- imputePCA(don,ncp=2)
> imp <- cbind.data.frame(res.comp$completeObs,tab.init[,1:4])
> res.pca <- PCA(imp,quanti.sup=1,quali.sup=12)
> plot(res.pca, hab=12, lab="quali"); plot(res.pca, choix="var")
> res.pca$ind$coord #scores (principal components)
```

Outline

Outline

Single imputation methods: Danger!



$\mu_y = 0$	0.01
$\sigma_y = 1$	0.5
$\rho = 0.6$	0.30
$CI\mu_y 95\%$	

Confidence interval for a mean

Let $Y = (Y_1, \ldots, Y_n)'$ be i.i.d. independent Gaussian random with expectation μ_y and variance $\sigma_y^2 > 0$.

- The empirical mean $\bar{Y} = n^{-1} \sum_{i=1}^{n} Y_i$
- $\bar{Y} \sim \mathcal{N}(\mu_y, \sigma_y^2/n)$
- A confidence interval for μ

$$\mathbb{P}\left(\bar{Y} - \frac{\sigma_y}{\sqrt{n}} z_{1-\alpha/2} \ \leq \mu \leq \bar{Y} + \frac{\sigma_y}{\sqrt{n}} z_{1-\alpha/2}\right) = 1 - \alpha$$

Confidence interval for a mean

Let $Y = (Y_1, \dots, Y_n)'$ be i.i.d. independent Gaussian random with expectation μ_y and variance $\sigma_y^2 > 0$.

- The empirical mean $\bar{Y} = n^{-1} \sum_{i=1}^n Y_i$
- $\bar{Y} \sim \mathcal{N}(\mu_y, \sigma_y^2/n)$
- A confidence interval for μ

$$\mathbb{P}\left(\bar{Y} - \frac{\sigma_y}{\sqrt{n}} z_{1-\alpha/2} \ \leq \mu \leq \bar{Y} + \frac{\sigma_y}{\sqrt{n}} z_{1-\alpha/2}\right) = 1 - \alpha$$

Variance unknown:

$$\frac{\sqrt{n}}{\widehat{\sigma_y}} \left(\bar{Y} - \mu_y\right) \sim T(n-1)$$

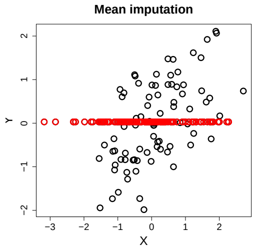
$$\left[\bar{y} - \frac{\widehat{\sigma}_y}{\sqrt{n}} qt_{1-\alpha/2}(n-1)\ ,\ \bar{y} + \frac{\hat{\sigma}_y}{\sqrt{n}} qt_{1-\alpha/2}(n-1)\right]$$

Simulation

- Generate bivariate Gaussian data ($\mu_y = 0, \sigma_y = 1, \rho = 0.6$)
- Put missing values on y
- Imput missing entries
- Compute the confidence interval of μ_y - count if the true value $\mu_y = 0$ is in the confidence interval
- Repeat the steps 10000 times
- Give the coverage

Single imputation methods

$$\left[\bar{y} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}; \bar{u} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}\right]$$

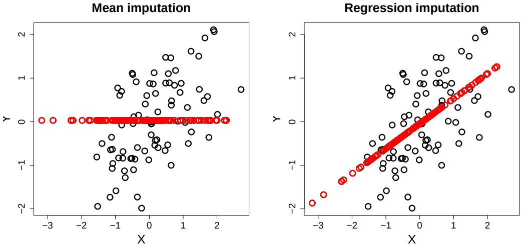


$\mu_y = 0$	0.01
$\sigma_y = 1$	0.5
$\rho = 0.6$	0.30
$CI \mu_y 95\%$	39.4

The idea of imputation is both seductive and dangerous *(Dempster and Rubin, 1983)*

Single imputation methods

$$\left[\bar{y} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}; \bar{u} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}\right]$$

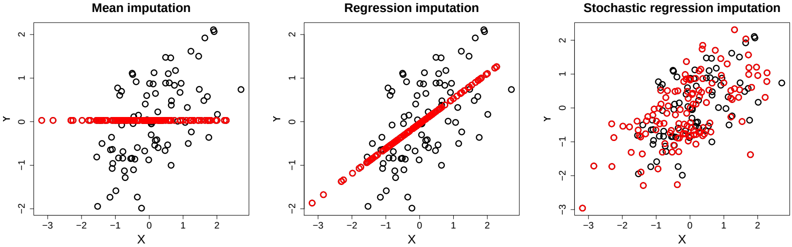


$\mu_y = 0$	0.01	0.01
$\sigma_y = 1$	0.5	0.72
$\rho = 0.6$	0.30	0.78
$CI\mu_y 95\%$	39.4	61.6

The idea of imputation is both seductive and dangerous *(Dempster and Rubin, 1983)*

Single imputation methods

$$\left[\bar{y} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}; \bar{u} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}\right]$$

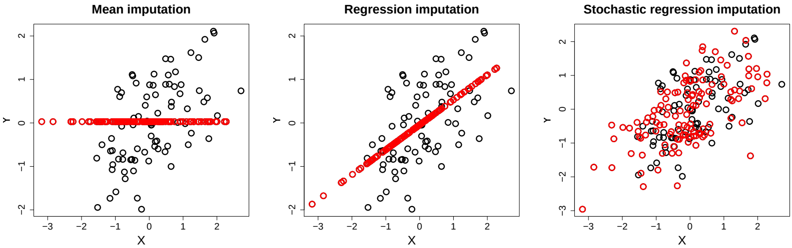


	Mean imputation	Regression imputation	Stochastic regression imputation
$\mu_y = 0$	0.01	0.01	0.01
$\sigma_y = 1$	0.5	0.72	0.99
$\rho = 0.6$	0.30	0.78	0.59
$CI\mu_y 95\%$	39.4	61.6	70.8

The idea of imputation is both seductive and dangerous *(Dempster and Rubin, 1983)*

Single imputation methods

$$\left[\bar{y} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}; \bar{u} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}\right]$$



	Mean imputation	Regression imputation	Stochastic regression imputation
$\mu_y = 0$	0.01	0.01	0.01
$\sigma_y = 1$	0.5	0.72	0.99
$\rho = 0.6$	0.30	0.78	0.59
$CI\mu_y 95\%$	39.4	61.6	70.8

The idea of imputation is both seductive and dangerous *(Dempster and Rubin, 1983)*

$\Rightarrow$ Standard errors of the parameters ($\hat{\sigma}_{\hat{\mu}_y}$) calculated from the imputed data set are underestimated

Underestimation of variance

Classical confidence interval for μ_y $\left[\bar{y} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}; \bar{Y} - qt_{n-1}\frac{\hat{\sigma}_y}{\sqrt{n}}\right]$

Asymptotic variance with missing values (Little & Rubin, p140):

$$\frac{\hat{\sigma}_y^2}{n_{obs}}\left(1 - \hat{\rho}^2\frac{n - n_{obs}}{n_{obs}}\right) = \frac{\hat{\sigma}_y^2}{n}\left(1 + \frac{n - n_{obs}}{n_{obs}}(1 - \hat{\rho}^2)\right)$$

$\Rightarrow$ When the $\rho = 1$, we trust the prediction and the coverage given by stochastic regression is OK.

$\Rightarrow$ Coverage of single imputation is too low: need to take into account the uncertainty associated to the predictions.

Single imputation: Underestimation of the variability

⇒ Incomplete Traumabase

X_1	X_2	X_3	...	Y
NA	20	10	...	shock
-6	45	NA	...	shock
0	NA	30	...	no shock
NA	32	35	...	shock
-2	NA	12	...	no shock
1	63	40	...	shock

Single imputation: Underestimation of the variability

⇒ Incomplete Traumabase

X_1	X_2	X_3	...	Y
NA	20	10	...	shock
-6	45	NA	...	shock
0	NA	30	...	no shock
NA	32	35	...	shock
-2	NA	12	...	no shock
1	63	40	...	shock

⇒ Completed Traumabase

X_1	X_2	X_3	...	Y
3	20	10	...	shock
-6	45	6	...	shock
0	4	30	...	no shock
-4	32	35	...	shock
-2	75	12	...	no shock
1	63	40	...	shock

Single imputation: Underestimation of the variability

⇒ Incomplete Traumabase

X_1	X_2	X_3	...	Y
NA	20	10	...	shock
-6	45	NA	...	shock
0	NA	30	...	no shock
NA	32	35	...	shock
-2	NA	12	...	no shock
1	63	40	...	shock

⇒ Completed Traumabase

X_1	X_2	X_3	...	Y
3	20	10	...	shock
-6	45	6	...	shock
0	4	30	...	no shock
-4	32	35	...	shock
-2	75	12	...	no shock
1	63	40	...	shock

A single value can't reflect the uncertainty of prediction

Multiple impute 1) Generate M plausible values for each missing value

X_1	X_2	X_3	Y
3	20	10	s
-6	45	6	s
0	4	30	no s
-4	32	35	s
-2	75	12	no s
1	63	40	s

X_1	X_2	X_3	Y
-7	20	10	s
-6	45	9	s
0	12	30	no s
13	32	35	s
-2	10	12	no s
1	63	40	s

X_1	X_2	X_3	Y
7	20	10	s
-6	45	12	s
0	-5	30	no s
2	32	35	s
-2	20	12	no s
1	63	40	s

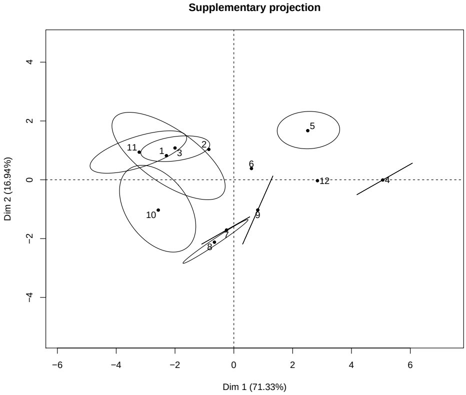
```
library(mice); mice(traumadata)
library(missMDA); MIPCA(traumadata)
```

Visualization of the imputed values

X_1	X_2	X_3	Y
3	20	10	s
-6	45	6	s
0	4	30	no s
-4	32	35	s
-2	15	12	no s
1	63	40	s

X_1	X_2	X_3	Y
-7	20	10	s
-6	45	9	s
0	12	30	no s
13	32	35	s
-2	10	12	no s
1	63	40	s

X_1	X_2	X_3	Y
7	20	10	s
-6	45	12	s
0	-5	30	no s
2	32	35	s
-2	20	12	no s
1	63	40	s




```
library(missMDA)
MIPCA(traumadata)
```

Percentage of NA?

Multiple imputation

1) Generate M plausible values for each missing value

X_1	X_2	X_3	Y
3	20	10	s
-6	45	6	s
0	4	30	no s
-4	32	35	s
1	63	40	s
-2	15	12	no s

X_1	X_2	X_3	Y
-7	20	10	s
-6	45	9	s
0	12	30	no s
13	32	35	s
1	63	40	s
-2	10	12	no s

X_1	X_2	X_3	Y
7	20	10	s
-6	45	12	s
0	-5	30	no s
2	32	35	s
1	63	40	s
-2	20	12	no s

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

3) Combine the results (Rubin's rules):

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

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

```
imp.mice <- mice(traumadata)
lm.mice.out <- with(imp.mice, glm(Y ~ ., family = "binomial"))
```

$\Rightarrow$ Variability of missing values taken into account

Outline

Multiple imputation: bivariate case

1. Generating M imputed data sets

   First idea: several stochastic regression
   for $m = 1, ..., M$, draw y_i from the predictive $\mathcal{N}(x_i\hat{\beta}, \hat{\sigma}^2)$

2. Performing the analysis on each imputed data set
3. Combining: variance = within + between imputation variance

	$M = 1$	$M = 50$
$\mu_y = 0$	0.01	0.01
$\sigma_y = 1$	0.99	0.99
$\rho = 0.6$	0.59	0.59
$CI\mu_y 95\%$	70.8	81.8

Multiple imputation: bivariate case

1. Generating M imputed data sets

   First idea: several stochastic regression
   for $m = 1, ..., M$, draw y_i from the predictive $\mathcal{N}(x_i\hat{\beta}, \hat{\sigma}^2)$

2. Performing the analysis on each imputed data set
3. Combining: variance = within + between imputation variance

	$M = 1$	$M = 50$
$\mu_y = 0$	0.01	0.01
$\sigma_y = 1$	0.99	0.99
$\rho = 0.6$	0.59	0.59
$CI\mu_y 95\%$	70.8	81.8

$\Rightarrow$ Variability of the parameters is missing: "improper" imputation

Multiple imputation: bivariate case

1. Generating M imputed data sets

   First idea: several stochastic regression
   for $m = 1, ..., M$, draw y_i from the predictive $\mathcal{N}(x_i\hat{\beta}, \hat{\sigma}^2)$

2. Performing the analysis on each imputed data set
3. Combining: variance = within + between imputation variance

	$M = 1$	$M = 50$
$\mu_y = 0$	0.01	0.01
$\sigma_y = 1$	0.99	0.99
$\rho = 0.6$	0.59	0.59
$CI\mu_y 95\%$	70.8	81.8

⇒ Variability of the parameters is missing: "improper" imputation

⇒ Prediction variance = estimation variance plus noise

Regression: variance of prediction

$y_{n+1} = x'_{n+1}\beta + \varepsilon_{n+1}$
$\hat{y}_{n+1} = x'_{n+1}\hat{\beta}$
$\hat{\beta} = (X'X)^{-1}X'Y$

$$\begin{aligned} V[\hat{y}_{n+1} - y_{n+1}] &= V[x'_{n+1}(\hat{\beta} - \beta) - \varepsilon_{n+1}] \\ &= x'_{n+1}V(\hat{\beta} - \beta)x_{n+1} + \sigma^2] \\ &= \hat{\sigma}^2\left(x'_{n+1}(X'X)^{-1}x_{n+1} + 1\right) \end{aligned}$$

CI for the prediction

$$\left[x'_{n+1}\hat{\beta} + -t_{n-p}(1-\alpha/2)\hat{\sigma}\sqrt{\left(x'_{n+1}(X'X)^{-1}x_{n+1} + 1\right)}\right]$$

Multiple imputation continuous data: bivariate case

$\Rightarrow$ Proper multiple imputation with $y_i = x_i\beta + \varepsilon_i$

1. Variability of the parameters, M plausible: $(\hat{\beta})^1, ..., (\hat{\beta})^M$

   $\Rightarrow$ Bootstrap
   $\Rightarrow$ Posterior distribution: Data Augmentation (Tanner & Wong, 1987)

2. Noise: for $m = 1, ..., M$, missing values y_i^m are imputed by drawing from the predictive distribution $\mathcal{N}(x_i\hat{\beta}^m, (\hat{\sigma}^2)^m)$

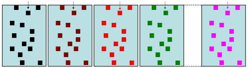
	Improper	Proper
$CI\mu_y 95\%$	0.818	0.935

Multiple imputation

⇒ Aim: provide estimation of the parameters and of their variability (taken into account the variability due to missing values)

Single imputation: a single value can't reflect the uncertainty of prediction ⇒ underestimate the standard errors

1. Generating M imputed data sets: variance of prediction



2. Performing the analysis on each imputed data set
3. Combining: variance = within + between imputation variance

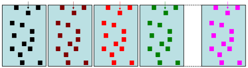
$$\hat{\beta} = \frac{1}{M}\sum_{m=1}^{M}\hat{\beta}_m \quad T = \frac{1}{M}\sum \widehat{Var}\left(\hat{\beta}_m\right) + \left(1+\frac{1}{M}\right)\frac{1}{M-1}\sum\left(\hat{\beta}_m - \hat{\beta}\right)^2$$

Multiple imputation

$\Rightarrow$ Aim: provide estimation of the parameters and of their variability (taken into account the variability due to missing values)

Single imputation: a single value can't reflect the uncertainty of prediction $\Rightarrow$ underestimate the standard errors

1. Generating M imputed data sets: variance of prediction



   1) Variance of estimation of the parameters + 2) Noise

2. Performing the analysis on each imputed data set
3. Combining: variance = within + between imputation variance

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

Joint modeling

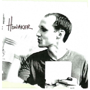
$\Rightarrow$ Hypothesis $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$

Algorithm Expectation Maximization Bootstrap:

1. Bootstrap rows: $X^1, \ldots, X^M$
   EM algorithm: $(\hat{\mu}^1, \hat{\Sigma}^1), \ldots, (\hat{\mu}^M, \hat{\Sigma}^M)$
2. Imputation: x_{ij}^m drawn from $\mathcal{N}\left(\hat{\mu}^m, \hat{\Sigma}^m\right)$

Easy to parallelized. Implemented in `Amelia` (website)

Amelia Earhart



James Honaker

Gary King

Matt Blackwell

Fully conditional modeling

⇒ Hypothesis: one model/variable

1. Initial imputation: mean imputation
2. For a variable j

   2.2 Imputation of the missing values in variable j with a model of X_j on the other X_{-j}: stochastic regression x_{ij} from $\mathcal{N}\left((x_{i,-j})'\hat{\beta}^{-j}, \hat{\sigma}^{-j}\right)$
3. Cycling through variables

⇒ Iteratively refine the imputation.

⇒ With continuous variables and a regression/variable: $\mathcal{N}(\mu, \Sigma)$

Implemented in `mice` (website) and Python

"*There is no clear-cut method for determining whether the MICE algorithm has converged*"

Stef van Buuren

Fully conditional modeling

⇒ Hypothesis: one model/variable

1. Initial imputation: mean imputation
2. For a variable j
   2.1 $(\hat{\beta}^{-j}, \hat{\sigma}^{-j})$ drawn from a Bootstrap: $(\hat{\beta}^{-j}, \hat{\sigma}^{-j})^1, ..., (\hat{\beta}^{-j}, \hat{\sigma}^{-j})^M$
   2.2 Imputation of the missing values in variable j with a model of X_j on the other X_{-j}: stochastic regression x_{ij} from $\mathcal{N}\left((x_{i,-j})'\hat{\beta}^{-j}, \hat{\sigma}^{-j}\right)$
3. Cycling through variables

Get M imputed data

⇒ Iteratively refine the imputation.

⇒ With continuous variables and a regression/variable: $\mathcal{N}(\mu, \Sigma)$

Implemented in `mice` (website) and Python

"*There is no clear-cut method for determining whether the MICE algorithm has converged*"

Stef van Buuren

Proof gibbs

Monte Carlo and Quasi-Monte Carlo Methods 2012, page 353

Monte Carlo statistical methods (Robert, Christian and Casella, George, 2004) (page 344)

The EM algorithm and extensions (McLachlan, Geoffrey J and Krishnan, Thriyambakam, 1998) (page 243) Example 6.7: Why Does Gibbs Sampling Work?

Joint / Conditional modeling

⇒ Both seen imputed values are drawn from a Joint distribution (even if joint does not exist)
⇒ Conditional modeling takes the lead?

- Flexible: one model/variable. Easy to deal with interactions and variables of different nature (binary, ordinal, categorical...)
- Many statistical models are conditional models!
- Tailor to your data
- Appears to work quite well in practice

⇒ Drawbacks: one model/variable... tedious...

Joint / Conditional modeling

⇒ Both seen imputed values are drawn from a Joint distribution (even if joint does not exist)

⇒ Conditional modeling takes the lead?

- Flexible: one model/variable. Easy to deal with interactions and variables of different nature (binary, ordinal, categorical...)
- Many statistical models are conditional models!
- Tailor to your data
- Appears to work quite well in practice

⇒ Drawbacks: one model/variable... tedious...

⇒ What to do with high correlation or when $n < p$?

- JM shrinks the covariance $\Sigma + k\mathbb{I}$ (selection of k?)
- CM: ridge regression or predictors selection/variable ⇒ a lot of tuning ... not so easy ...

Multiple imputation with Bootstrap PCA

$$x_{ij} = \mu_{ij} + \varepsilon_{ij} = \sum_{s=1}^{S} \sqrt{\tilde{\lambda}_s} \tilde{u}_{is} \tilde{v}_{js} + \varepsilon_{ij} \ , \ \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

1. Variability of the parameters, M plausible: $(\hat{\mu}_{ij})^1, ..., (\hat{\mu}_{ij})^M$
2. Noise: for $m = 1, ..., M$, missing values x_{ij}^m drawn $\mathcal{N}(\hat{\mu}_{ij}^m, \hat{\sigma}^2)$

Implemented in missMDA (website)

François Husson

Joint, conditional and PCA

⇒ Good estimates of the parameters and their variance from an incomplete data (coverage close to 0.95)
The variability due to missing values is well taken into account

`Amelia` & `mice` have difficulties with large correlations or $n < p$
`missMDA` does not but requires a tuning parameter: number of dim.

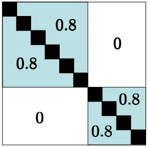
`Amelia` & `missMDA` are based on linear relationships
`mice` is more flexible (one model per variable)

MI based on PCA works in a large range of configuration, $n < p$, $n > p$ strong or weak relationships, low or high percentage of missing values

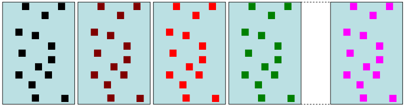
Simulations

The simulated data $\mathcal{N}(\mu, \Sigma)$

- 2 underlying dimensions (control k)
- n (30,200), p (6,60), ρ (0.3,0.8), %NA (10,30)



$\Rightarrow$ Imputation with $B = 100$ imputed tables with PCA, JM, CM



Estimate (analysis model): $\hat{\theta}_b, \widehat{Var}\left(\hat{\theta}_b\right)$: $\theta_1 = \mathbb{E}\left[Y\right], \theta_2 = \beta_1$

Rubin: $\hat{\theta} = \frac{1}{B}\sum_{b=1}^{B}\hat{\theta}_b \;\; T = \frac{1}{B}\sum_b \widehat{Var}\left(\hat{\theta}_b\right) + \frac{1}{B-1}\sum_b\left(\hat{\theta}_b - \hat{\theta}\right)^2$

$\Rightarrow$ Bias, CI width, coverage - 1000 simulations

Results for the expectation

	parameters				confidence interval width			coverage		
	n	p	ρ	%	*Joint*	*Cond*	*MIPCA*	*Joint*	*Cond*	*MIPCA*
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

$\Rightarrow$ Good estimates of θ and coverage ≈ 0.95: variability due to missing is taken into account

$\Rightarrow$ PCA: small - large n/p; strong - weak relation; low-high % NA

Outline

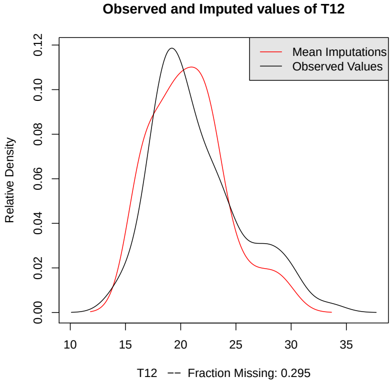
Multiple imputation in practice

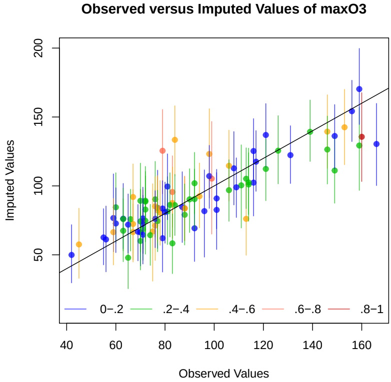
⇒ Step 1: Generate M imputed data sets

```
> library(Amelia)
> res.amelia <- amelia(don, m = 100)

> library(mice)
> res.mice <- mice(don, m = 100, defaultMethod = "norm.boot")

> library(missMDA)
>  res.MIPCA <- MIPCA(don, ncp = 2, nboot  = 100)
> res.MIPCA$res.MI
```

Multiple imputation in practice

⇒ Step 2: visualization



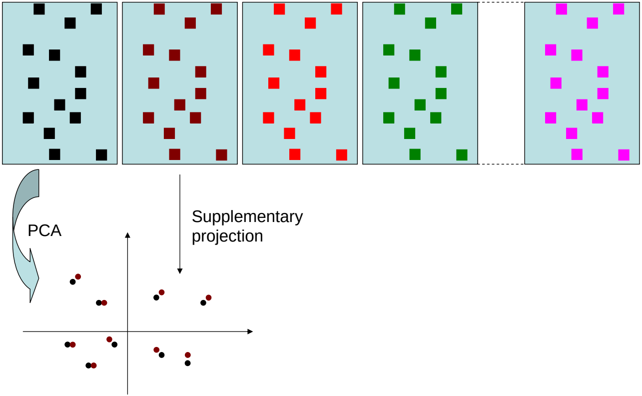



```
# library(Amelia)
> res.amelia <- amelia(don, m = 100)
> compare.density(res.amelia, var = "T12")
> overimpute(res.amelia, var = "maxO3")

# library(missMDA)
res.over<-Overimpute(res.MIPCA)
```

Multiple imputation in practice

⇒ Step 2: visualization

⇒ Individuals position (and variables) with other predictions

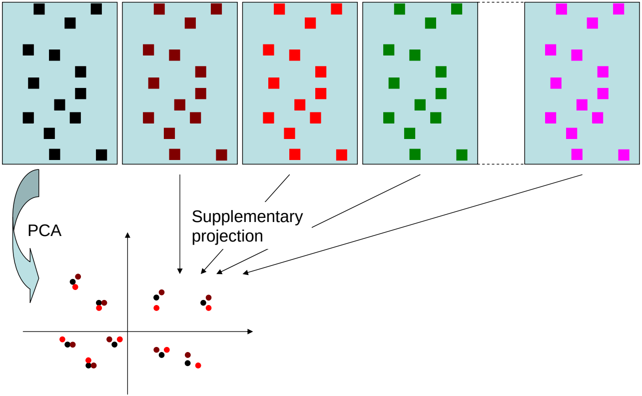


Regularized iterative PCA
⇒ reference configuration

Multiple imputation in practice

⇒ Step 2: visualization

⇒ Individuals position (and variables) with other predictions

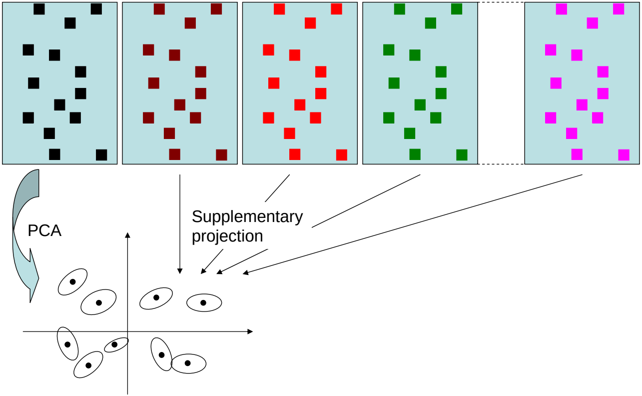


Regularized iterative PCA
⇒ reference configuration

Multiple imputation in practice

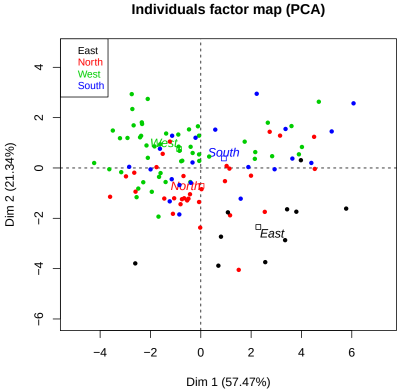
⇒ Step 2: visualization

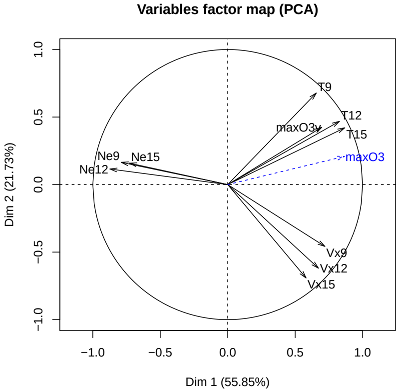
⇒ Individuals position (and variables) with other predictions



Regularized iterative PCA
⇒ reference configuration

PCA representation



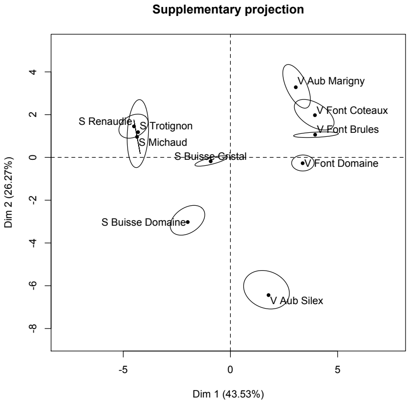


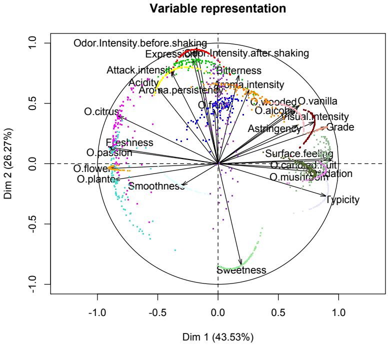

```
> imp <- cbind.data.frame(res.comp$completeObs, ozo[, 12])
> res.pca <- PCA(imp,quanti.sup = 1, quali.sup = 12)
> plot(res.pca, hab =12, lab = "quali"); plot(res.pca, choix = "var")
> res.pca$ind$coord #scores (principal components)
```

Multiple imputation in practice

⇒ Step 2: visualization

```
> res.MIPCA <- MIPCA(don, ncp = 2)
> plot(res.MIPCA, choice = "ind.supp"); plot(res.MIPCA, choice = "var")
```





Multiple imputation in practice

⇒ Step 3. Regression on each table and pool the results

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

```
> library(mice)
> res.mice <- mice(don, m = 100)
> imp.micerf <- mice(don, m = 100, defaultMethod = "rf")
> lm.mice.out <- with(res.mice, lm(maxO3 ~ T9+T12+T15+Ne9+...+Vx15+maxO3v))
> pool.mice <- pool(lm.mice.out)
> summary(pool.mice)
```

```
              est    se     t    df Pr(>|t|)  lo 95 hi 95 nmis  fmi lambda
(Intercept) 19.31 16.30  1.18 50.48     0.24 -13.43 52.05   NA 0.46   0.44
T9          -0.88  2.25 -0.39 26.43     0.70  -5.50  3.75   37 0.71   0.69
T12          3.29  2.38  1.38 27.54     0.18  -1.59  8.18   33 0.70   0.68
....
Vx15         0.23  1.33  0.17 39.00     0.87  -2.47  2.93   21 0.57   0.55
maxO3v       0.36  0.10  3.65 46.03     0.00   0.16  0.56   12 0.50   0.48
```

Outline

Categorical data

Survey data

```
 region                        sex        age          year         edu         drunk        alcohol      glasses
 Ile de France       :8120   F:29776   18_25: 6920   2005:27907   E1:12684  0     :44237   <1/m :12889   0  : 2812
 Rhone Alpes         :5421   M:23165   26_34: 9401   2010:25034   E2:23521  1-2   : 4952   0    : 6133   0-2:37867
 Provence Alpes      :4116             35_44:10899                E3:6563   10-19:  839    1-2/m: 7583   10+:  590
 Nord Pas de Calais  :3819             45_54: 9505                E4:10100  20-29:  212    1-2/w: 9526   3-4: 9401
 Pays de Loire       :3152             55_64: 9503                NA:73     3-5   : 1908   3-4/w: 6815   5-6: 1795
 Bretagne            :3038             65_+ : 6713                          30+   :  404   5-6/w: 3402   7-9:  391
 (Other)             :25275                                                 6-9   :  389   7/w  : 6593          NA: 85

 binge                   Pbsleep                   Tabac
 <2/m:10323              Never:20605               Frequent  : 9176
 0   :34345              Often: 10172              Never     :39080
 1/m : 6018              Rare :22134               Occasional: 4588
 1/w : 1800              NA:  30                   NA:  97
 7/w :  374
 NA  :   81
```

INPES `http://www.inpes.sante.fr`

Principal components method: Multiple Correpondence Analysis Single imputation based on MCA for categorical data

Multiple Correspondence Analysis (MCA)

$X_{n \times m}$ m categorical variables coded with indicator matrix A

$$X = \begin{array}{|c|c|c|} y & \dots & attack \\ y & \dots & attack \\ y & \dots & attack \\ n & \dots & suicide \\ & & \\ n & \dots & accident \\ n & \dots & suicide \end{array} \quad A = \begin{array}{|cc|c|ccc|} 1 & 0 & \dots & 1 & 0 & 0 \\ 1 & 0 & \dots & 1 & 0 & 0 \\ 1 & 0 & \dots & 1 & 0 & 0 \\ 0 & 1 & \dots & 0 & 1 & 0 \\ & & & & & \\ 0 & 1 & \dots & 0 & 0 & 1 \\ 0 & 1 & \dots & 0 & 1 & 0 \end{array} \quad D_p = \begin{pmatrix} p_1 & & 0 \\ & \ddots & \\ 0 & & p_J \end{pmatrix}$$

For a category c, the frequency of the category: $p_c = n_c/n$.

A SVD on weighted matrix: $Z = \frac{1}{\sqrt{mn}}(A - 1p^T)D_p^{-1/2} = U \Lambda V'$

The PC ($F = U\Lambda^{1/2}$) satisfies: $\arg\max_{F_s \in \mathbb{R}^n} \quad \frac{1}{m}\sum_{j=1}^{m} \eta^2(F_s, X_j)$

$$\eta^2(F, X_j) = \frac{\sum_{c=1}^{C_j} n_c (F_{.c} - F_{..})^2}{\sum_{i=1}^{n}\sum_{c=1}^{C_j}(F_{ic})^2} = \frac{\text{RSS between}}{\text{RSS tot}}$$

Benzecri, 1973 :*"In data analysis the mathematical problems reduces to computing eigenvectors; all the science (the art) is in finding the right matrix to diagonalize"*

Regularized iterative MCA *(Chavent et al., 2012)*

Iterative MCA algorithm:

	V1	V2	V3	...	V14
ind 1	a	NA	g	...	u
ind 2	NA	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	NA		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	NA	NA	1	0	...
ind 2	NA	NA	NA	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	NA	NA	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

```
library(missMDA); ?imputeMCA
```

Iterative MCA algorithm:

1. initialization: imputation of the indicator matrix (proportion)

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.41	0.59	1	0	...
ind 2	0.20	0.30	0.50	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.27	0.78	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

```
library(missMDA); ?imputeMCA
```

Regularized iterative MCA *(Chavent et al., 2012)*

Iterative MCA algorithm:

1. initialization: imputation of the indicator matrix (proportion)
2. iterate until convergence
   (a) estimation: MCA on the completed data $\rightarrow U, \Lambda, V$

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.41	0.59	1	0	...
ind 2	0.20	0.30	0.50	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.27	0.78	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

```
library(missMDA); ?imputeMCA
```

Regularized iterative MCA *(Chavent et al., 2012)*

Iterative MCA algorithm:

1. initialization: imputation of the indicator matrix (proportion)
2. iterate until convergence
   (a) estimation: MCA on the completed data $\rightarrow U, \Lambda, V$
   (b) imputation with the fitted matrix $\hat{\mu} = U_S \Lambda_S^{1/2} V_S'$

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.65	0.35	1	0	...
ind 2	0.11	0.20	0.69	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.30	0.40	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

```
library(missMDA); ?imputeMCA
```

Regularized iterative MCA *(Chavent et al., 2012)*

Iterative MCA algorithm:

1. initialization: imputation of the indicator matrix (proportion)
2. iterate until convergence
   (a) estimation: MCA on the completed data $\rightarrow U, \Lambda, V$
   (b) imputation with the fitted matrix $\hat{\mu} = U_S \Lambda_S^{1/2} V_S'$
   (c) column margins are updated

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.65	0.35	1	0	...
ind 2	0.11	0.20	0.69	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.30	0.40	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

```
library(missMDA); ?imputeMCA
```

Regularized iterative MCA *(Chavent et al., 2012)*

Iterative MCA algorithm:

1. initialization: imputation of the indicator matrix (proportion)
2. iterate until convergence
   (a) estimation: MCA on the completed data $\rightarrow U, \Lambda, V$
   (b) imputation with the fitted matrix $\hat{\mu} = U_S \Lambda_S^{1/2} V_S'$
   (c) column margins are updated

	V1	V2	V3	…	V14
ind 1	a	**NA**	g	…	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
…	…	…	…		…
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	…
ind 1	1	0	0	**0.71**	**0.29**	1	0	…
ind 2	**0.12**	**0.29**	**0.59**	0	1	1	0	…
ind 3	1	0	0	1	0	0	1	…
ind 4	1	0	0	1	0	0	1	…
ind 5	0	1	0	0	1	0	1	…
ind 6	0	0	1	0	1	0	1	…
ind 7	0	0	1	0	1	**0.37**	**0.63**	…
…	…	…	…	…	…	…	…	…
ind 1232	0	0	1	0	1	0	1	…

$\Rightarrow$ the imputed values can be seen as degree of membership

```
library(missMDA); ?imputeMCA
```

Regularized iterative MCA *(Chavent et al., 2012)*

Iterative MCA algorithm:

1. initialization: imputation of the indicator matrix (proportion)
2. iterate until convergence
   (a) estimation: MCA on the completed data $\rightarrow U, \Lambda, V$
   (b) imputation with the fitted matrix $\hat{\mu} = U_S \Lambda_S^{1/2} V_S'$
   (c) column margins are updated

	V1	V2	V3	...	V14
ind 1	a	**e**	g	...	u
ind 2	**c**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**g**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	**0.71**	**0.29**	1	0	...
ind 2	**0.12**	**0.29**	**0.59**	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	**0.37**	**0.63**	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Two ways to obtain categories: majority or draw

```
library(missMDA); ?imputeMCA
```

Multiple imputation with MCA

1. Variability of the parameters: M sets $(U_{n\times S}, \Lambda_{S\times S}, V^{\top}_{m\times S})$ using a non-parametric bootstrap

$\hat{X}_1$

1	0	...	1	0	0
1	0	...	1	0	0
1	0	...			
			0.01	0.80	0.19
			0	0	1
0.25	0.75				
0	1		0	0	1

$\hat{X}_2$

1	0	...	1	0	0
1	0	...	1	0	0
1	0	...			
			0.60	0.2	0.20
			0	0	1
0.26	0.74				
0	1		0	0	1

...

$\hat{X}_M$

1	0	...	1	0
1	0	...	1	0
1	0	...		
			0.11	0.74
			0	0
0.20	0.80			
0	1		0	0

2. Categories drawn from multinomial disribution using the values in $\left(\hat{X}_m\right)_{1\leq m\leq M}$

y	...	Attack
y	...	Attack
y	...	
		Suicide
	...	Accident
n		
n	...	S

y	...	Attack
y	...	Attack
y	...	
		Attack
	...	Accident
n		
n	...	B

...

y	...	Attack
y	...	Attack
y	...	
		Suicide
	...	Accident
n		
n	...	Suicide

```
library(missMDA); MIMCA()
```

Multiple imputation for categorical data

⇒ Joint modeling:

- Log-linear model (Schafer, 1997) (cat): pb many levels
- Latent class models (Vermunt, 2014) - nonparametric Bayesian (Si & Reiter, 2014, Murray & Reiter, 2016) (MixedDataImpute, NPBayesImpute, NestedCategBayesImpute)

⇒ Conditional model: logistic, multinomial logit, forests (mice)

⇒ MIMCA provides valid inference (ex. logistic reg with missing) applied to data of various size (many levels, rare levels)

Time (seconds)	Titanic	Galetas	Income
rows-variables-levels	(2000 - 4 - 4)	(1000 - 4 -11)	(6000 - 14 - 9)
MIMCA	2.750	8.972	**58.729**
Loglinear	0.740	4.597	NA
Nonparametric bayes	10.854	17.414	143.652
Cond logistic	4.781	38.016	881.188
Cond forests	265.771	112.987	6329.514

Categorical imputation in practice

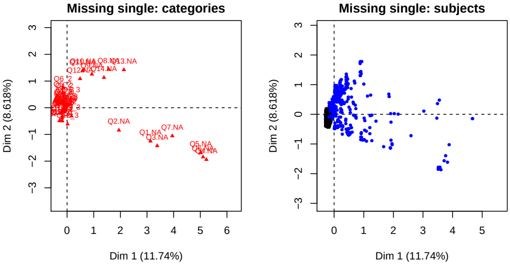
- 1232 respondents, 14 questions, 35 categories, 9% of missing values concerning 42% of respondents

In missMDA (Youtube)

```
data(vnf)
summary(vnf)
MCA(vnf)

#1) select the number of components
nb <- estim_ncpMCA(vnf, ncp.max = 5) #Time-consuming, nb = 4

#2) Impute the indicator matrix
res.impute <- imputeMCA(vnf, ncp = 4)
res.impute$tab.disj
res.impute$comp
summary(res.impute$comp)

# MCA on the incomplete data vnf
res.mca <- MCA(vnf, tab.disj = res.impute$tab.disj)
plot(res.mca, invisible=c("var"))
plot(res.mca,invisible=c("ind"),autoLab="yes", selectMod="cos2 5", cex  = 0.6)
```

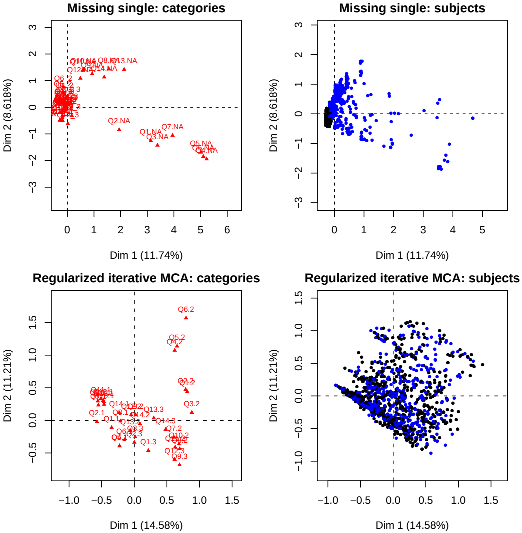
Categorical imputation in practice

- 1232 respondents, 14 questions, 35 categories, 9% of missing values concerning 42% of respondents



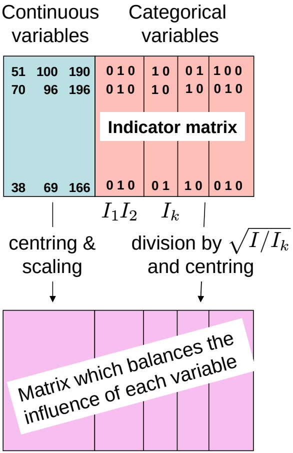
Categorical imputation in practice

- 1232 respondents, 14 questions, 35 categories, 9% of missing values concerning 42% of respondents



Principal component method for mixed data (complete)

Factorial Analysis Mixed Data FAMD (Escofier, 1979), PCAMIX (Kiers, 1991)



A PCA is performed on the weighted matrix with standard deviation for continuous variable and square root of the proportion for categorical variables

Properties of FAMD (complete)

Benzecri, 1973 :*"All in all, doing a data analysis, in good mathematics, is simply searching eigenvectors; all the science (or the art) of it is just to find the right matrix to diagonalize"*

- The distance between observations is:

$$d^2(i,l) = \sum_{j=1}^{p_{cont}} \frac{1}{\sigma_j}(x_{ij} - x_{lj})^2 + \sum_{q=1}^{Q_{cat}} \sum_{k=1}^{K_q} \frac{1}{I_{k_q}}(x_{ik_q} - x_{lk_q})^2$$

- The principal component F_s maximises:

$$\sum_{j=1}^{p_{cont}} r^2(F_s, x_{.j}) + \sum_{q=1}^{Q_{cat}} \eta^2(F_s, x_{.q})$$

Iterative FAMD algorithm

1. Initialization: imputation mean (continuous) and proportion (dummy)
2. Iterate until convergence
   (a) estimation: FAMD on the completed data $\Rightarrow U, \Lambda, V$
   (b) imputation of the missing values with the fitted matrix $\hat{X} = U_S \Lambda_S^{1/2} V_S'$
   (c) means, standard deviations and column margins are updated

age	weight	size	alcohol	sex	snore	tobacco
NA	100	190	NA	M	yes	no
70	96	186	1-2 gl/d	M	NA	<=1
NA	104	194	No	W	no	NA
62	68	165	1-2 gl/d	M	no	<=1

NA	100	190	NA	NA	NA	1	0	0	1	1	0	0
70	96	186	0	1	0	1	0	NA	NA	0	1	0
NA	104	194	1	0	0	0	1	1	0	NA	NA	NA
62	68	165	0	1	0	1	0	1	0	0	1	0

imputeAFDM

51	100	190	0.2	0.7	0.1	1	0	0	1	1	0	0
70	96	186	0	1	0	1	0	0.8	0.2	0	1	0
48	104	194	1	0	0	0	1	1	0	0.1	0.8	0.1
62	68	165	0	1	0	1	0	1	0	0	1	0

age	weight	size	alcohol	sex	snore	tobacco
51	100	190	1-2 gl/d	M	yes	no
70	96	186	1-2 gl/d	M	no	<=1
48	104	194	No	W	no	<=1
62	68	165	1-2 gl/d	M	no	<=1

$\Rightarrow$ Imputed values can be seen as degrees of membership

Simulation study

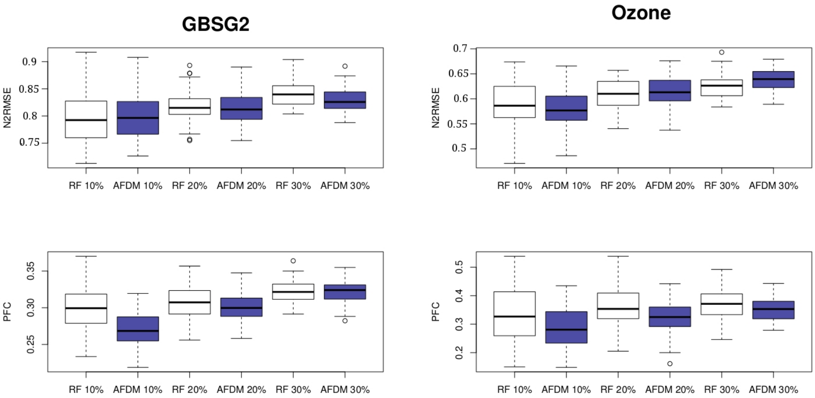
Several data sets

- Relationships between variables
- Number of categories
- percentage of missing values (10%,20%,30%)

Criteria:

- for continuous data: RMSE
- for categorical data: proportion of falsely classified entries

Imputations obtained with random forest & FAMD algorithm



Summary

Imputations with PC methods are good:

- for strong linear relationships
- for categorical variables
- especially for rare categories (weights of MCA)

⇒ Number of components S?? Cross-Validation (GCV)

Imputations with RF are good:

- for strong non-linear relationships between continuous variables
- when there are interactions

⇒ No tunning parameters?

Rq: categorical data improve the imputation on continuous data and continuous data improve the imputation on categorical data

Summary

Imputations with PC methods are good:

- for strong linear relationships
- for categorical variables
- especially for rare categories (weights of MCA)

$\Rightarrow$ Number of components S?? Cross-Validation (GCV)

Imputations with RF are good:

- for strong non-linear relationships between continuous variables (cutting continuous variables into categories)
- when there are interactions (creating interactions)

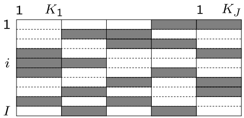
$\Rightarrow$ No tunning parameters?

Rq: categorical data improve the imputation on continuous data and continuous data improve the imputation on categorical data

Mixed imputation in practice

```
> library(missMDA)
> res.ncp <- estim_ncpFAMD(ozo)
> res.famd <-imputeFAMD(ozo, ncp = 2)
> res.famd$completeObs

> library(missForest)
> res.rf <- missForest(ozo)
> res.rf$ximp
```

Multi-blocks data set



- Sensory analysis: products described by people and by physico-chemical measurements

  (each judge can't taste more than 8 products: Planned missing products per judge, experimental design: BIB)

- Biology. DNA/RNA (samples without expression data)

Continuous / categorical / contingency sets of variables

⇒ Missing rows per subtable
⇒ Regularized iterative Multiple Factor Analysis (Husson & Josse, 2013)

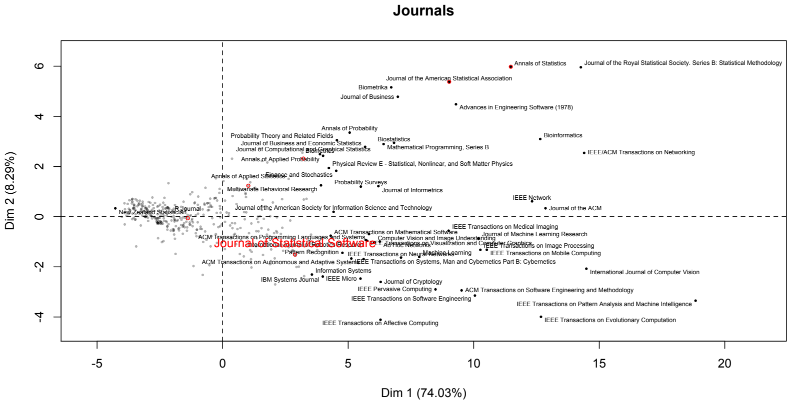
Journal impact factors

journalmetrics.com provides 27000 journals/ 15 years of metrics.

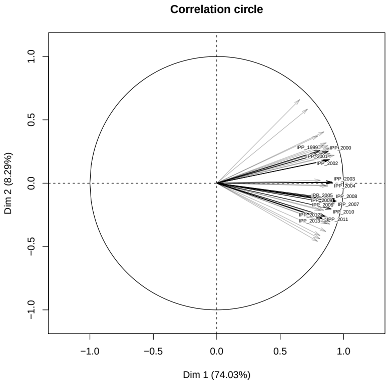
443 journals (Computer Science, Statistics, Probability and Mathematics). 45 metrics, some may be NA, 15 years by 3 types of measures:

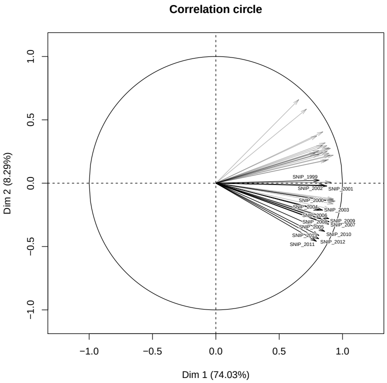
- IPP - Impact Per Publication (like the ISI impact factor but for 3 (rather than 2) years.
- SNIP - Source Normalized Impact Per Paper: Tries to weight by the number of citations per subject field to adjust for different citation cultures.
- SJR - SCImago Journal Rank: Tries to capture average prestige per publication.

MFA with missing values

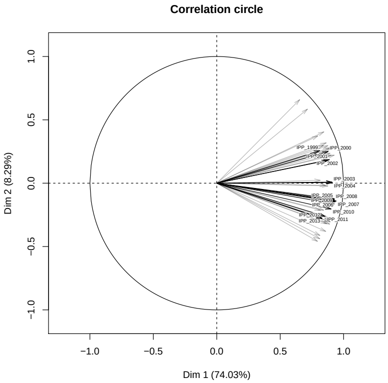


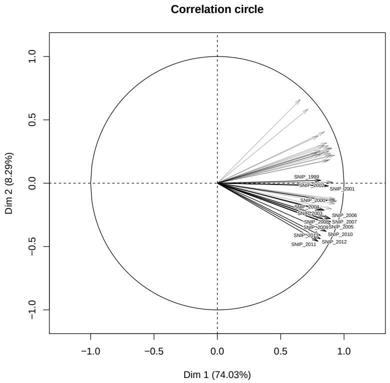
MFA with missing values





MFA with missing values

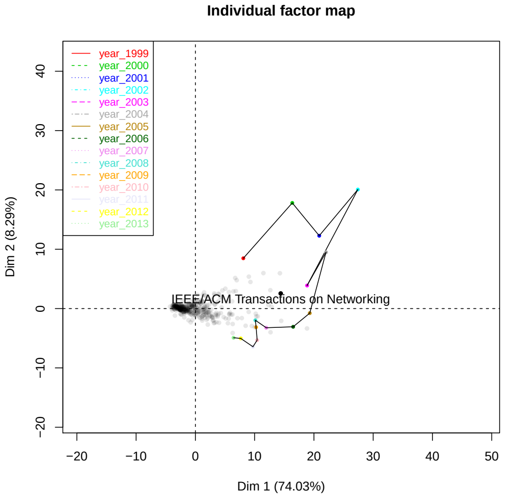




MFA with missing values

Rows: 47000 journals / Groups: 15 years of data/ Variables: 3 scores each year. Many missing...

ACM Transactions on Networking trajectory.pdf



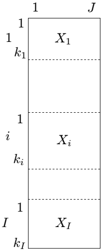
Multi-table imputation in practice

```
> library(denoiseR)
> library(missMDA)
> data(impactfactor)
> year=NULL; for (i in 1: 15) year= c(year, seq(i,45,15))
> res.imp <- imputeMFA(impactfactor,  group = rep(3, 15),  type = rep("s", 15))

##
> res.mfa  <-MFA(res.imp$completeObs, group=rep(3,15),  type=rep("s",15),
name.group=paste("year", 1999:2013,sep="_"),graph=F)

plot(res.mfa, choix = "ind", select = "contrib 15", habillage = "group", cex = 0.7)
points(res.mfa$ind$coord[c("Journal of Statistical Software",
"Journal of the American Statistical Association", "Annals of Statistics"),
1:2], col=2, cex=0.6)
text(res.mfa$ind$coord[c("Journal of Statistical Software"), 1],
res.mfa$ind$coord[c("Journal of Statistical Software"), 2],cex=1,
labels=c("Journal of Statistical Software"),pos=3, col=2)

plot.MFA(res.mfa,choix="var", cex=0.5,shadow=TRUE, autoLab = "yes")

plot(res.mfa, select="IEEE/ACM Transactions on Networking",
partial="all",
habillage="group",unselect=0.9,chrono=TRUE)
```

Multilevel component analysis

Ex: inhabitants nested within countries $X \in \mathbb{R}^{K \times J}$

- similarities between countries? level 1
- similarities between inhabitants within each country? level 2
- relationship between variables at each level



$$x_{ijk_i} = x_{.j.} + (x_{ij.} - x_{.j.}) \quad + \quad (x_{ijk_i} - x_{ij.})$$

$$\text{Between} \quad + \quad \text{Within}$$

Analysis of variance: split the sum of squares for each variable j

$$\sum_{i=1}^{I}\sum_{k=1}^{k_i}(x_{ijk_i})^2 = \sum_{i=1}^{I} k_i (x_{.j.})^2 + \sum_{i=1}^{I} k_i (x_{ij.} - x_{.j.})^2 + \sum_{i=1}^{I}\sum_{k=1}^{k_i}(x_{ijk_i} - x_{ij.})^2$$

Multilevel PCA MLPCA

$\Rightarrow$ Model for the between and within part $i = 1, ..., I$ groups, J var

$$X_{i_{(k_i \times J)}} = 1_{k_i} m' + 1_{k_i} F_i^{b'} V^{b'} + F_i^w V^{w'} + E_i$$

- F_i^b $(Q_b \times 1)$ between component scores of group i
- V^b $(J \times Q_b)$ between loading matrix
- F_i^w $(k_i \times Q_w)$ within component scores of group i
- V_w $(J \times Q_w)$ within loading matrix. Constant across groups

Fitted by solving the least squares (Timmerman, 2006)

$$\text{argmin}\, F(m, F_i^b, V^b, F_i^w, V^w) = \sum_{i=1}^{I} \left\| X_i - 1_{k_i} m' - 1_{k_i} F_i^{b'} V^{b'} - F_i^w V^{w'} \right\|^2,$$

$\sum_{i=1}^{I} k_i F_i^b = 0_{Q_b}$ and $1'_{k_i} F_i^w = 0_{Q_w}$, $\forall i$ for identifiability.

MLPCA - quantitative data

$i = 1, ..., I$ groups, J var, k_i nb obs in group i
$\Rightarrow$ Estimation: minimize the RSS

$$\text{argmin}\, F() = \sum_{i=1}^{I} \left\| X_i - 1_{k_i} m' - 1_{k_i} F_i^{b'} V^{b'} - F_i^{w} V^{w'} \right\|^2,$$

$\sum_{i=1}^{I} k_i F_i^b = 0_{Q_b}$ and $1'_{k_i} F_i^w = 0_{Q_w}$, $\forall i$ for identifiability.

$(\hat{F}^b, \hat{V}^b)$: Weigthed PCA on the between part: SVD on $D_w X_m$; X_m $(I \times J)$ the means of the variables per group, D_w $(I \times I)$ $D_{w\,ii} = \sqrt{k_i}$

$(\hat{F}^w, \hat{V}^w)$ PCA on the within part: SVD on the centered data per group X^w $(K \times J)$, $K = \sum_i k_i$

$\Rightarrow$ With missing values: Weighted Least Squares

$\Rightarrow$ Iterative imputation algorithm (imputation - estimation)

Iterative MLPCA

2. iteration ℓ: *estimation of the between structure*
   - SVD $D_w X_m^\ell = PDQ'$; Q_b eigenvectors are kept:
     $\hat{F}_i^b = [D_w^{-1} P_{Q_b}]_i$, $\hat{F}^b$ concatenation by row of $[\mathbf{1}_{k_i} \hat{F}_i^b]$
     $\hat{V}^b = Q_{Q_b} D_{Q_b}$, $(J \times Q_b)$
   - the between hat matrix is computed: $(\hat{X}^b)^\ell = \hat{F}^b \hat{V}^{b'}$

3. iteration ℓ: *imputation of the missing values with the fitted values*
   - $\hat{X}^\ell = \mathbf{1}_K \hat{m}^{(\ell-1)'} + (\hat{X}^b)^\ell + (\hat{X}^w)^{(\ell-1)}$. The newly imputed dataset is
     $X^\ell = W \odot X + (\mathbf{1}_K \times \mathbf{1}_J' - W) \odot \hat{X}^\ell$
   - $\hat{m}^\ell$ is computed on X^ℓ

4. iteration ℓ: *estimation of the within structure*
   - SVD $(X^w)^\ell = PDQ'$; Q_w eigenvectors are kept:
     $F^w = P_{Qw}$ $(K \times Q_w)$
     $V^w = Q_{Qw} D_{Qw}$ $(J \times Q_w)$
   - the within hat matrix is computed $(\hat{X}^w)^\ell = \hat{F}^w \hat{V}^{w'}$

5. iteration ℓ: *imputation of the missing values with the fitted values*
   - $X^{\ell+1} = W \odot X + (\mathbf{1}_K \times \mathbf{1}_J' - W) \odot \left(\mathbf{1}_K \hat{m}^{(\ell)'} + (\hat{X}^b)^\ell + (\hat{X}^w)^\ell\right)$
   - $\hat{m}^{\ell+1}$ is computed on $X^{\ell+1}$

Multilevel MCA

$\Rightarrow$ Start with the matrix of dummy variables A and define a between and a within part
$\Rightarrow$ Then, MCA is applied on each part

Between: Apply MCA on the matrix with the mean of A per group i (proportion of obs taking each category in group i) (proportion of some disease in a particular hospital). $\hat{A}^b = F^b V^{b'} D_p^{1/2} + 1_n p^{'}$

Within part Apply MCA on the data where the between part has been swept out (SVD is applied to $\frac{1}{np}\left(A - \hat{A}^b\right) D_p^{-1/2}$)
$\hat{A}^w = (np) F^w V^{w'} D_p^{1/2}$.

$$\hat{A} = \hat{A}^b + \hat{A}^w$$

Regularized iterative Multilevel MCA

	V1	V2	V3	...	V14
ind 1	a	NA	g	...	u
ind 2	NA	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	NA		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	NA	NA	1	0	...
ind 2	NA	NA	NA	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	NA	NA	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)

	V1	V2	V3	...	V14
ind 1	a	NA	g	...	u
ind 2	NA	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	NA		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.41	0.59	1	0	...
ind 2	0.20	0.30	0.50	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.27	0.78	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)
- Iterate until convergence
  1. estimation: Multilevel MCA on the completed data → $\hat{F}^b, \hat{V}^b, \hat{F}^w, \hat{V}^w$

	V1	V2	V3	...	V14
ind 1	a	NA	g	...	u
ind 2	NA	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	NA		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.41	0.59	1	0	...
ind 2	0.20	0.30	0.50	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.27	0.78	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)
- Iterate until convergence
  1. estimation: Multilevel MCA on the completed data $\rightarrow$ $\hat{F}^b, \hat{V}^b, \hat{F}^w, \hat{V}^w$
  2. imputation with the fitted matrix $\hat{A} = \hat{A}^b + \hat{A}^w$

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.65	0.35	1	0	...
ind 2	0.11	0.20	0.69	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.30	0.40	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)
- Iterate until convergence
  1. estimation: Multilevel MCA on the completed data $\rightarrow$ $\hat{F}^b, \hat{V}^b, \hat{F}^w, \hat{V}^w$
  2. imputation with the fitted matrix $\hat{A} = \hat{A}^b + \hat{A}^w$
  3. column margins are updated

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	0.65	0.35	1	0	...
ind 2	0.11	0.20	0.69	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	0.30	0.40	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)
- Iterate until convergence
  1. estimation: Multilevel MCA on the completed data $\rightarrow$ $\hat{F}^b, \hat{V}^b, \hat{F}^w, \hat{V}^w$
  2. imputation with the fitted matrix $\hat{A} = \hat{A}^b + \hat{A}^w$
  3. column margins are updated

	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	**0.71**	**0.29**	1	0	...
ind 2	**0.12**	**0.29**	**0.59**	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	**0.37**	**0.63**	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

$\Rightarrow$ the imputed values can be seen as degree of membership

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)
- Iterate until convergence
  1. estimation: Multilevel MCA on the completed data $\rightarrow$ $\hat{F}^b, \hat{V}^b, \hat{F}^w, \hat{V}^w$
  2. imputation with the fitted matrix $\hat{A} = \hat{A}^b + \hat{A}^w$
  3. column margins are updated

	V1	V2	V3	...	V14
ind 1	a	**e**	g	...	u
ind 2	**c**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**g**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	**0.71**	**0.29**	1	0	...
ind 2	**0.12**	**0.29**	**0.59**	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	**0.37**	**0.63**	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Two ways to impute categories: majority or draw

Regularized iterative Multilevel MCA

- Initialization: imputation of the indicator matrix (proportions)
- Iterate until convergence
  1. estimation: Multilevel MCA on the completed data $\rightarrow$ $\hat{F}^b, \hat{V}^b, \hat{F}^w, \hat{V}^w$
  2. imputation with the fitted matrix $\hat{A} = \hat{A}^b + \hat{A}^w$
  3. column margins are updated

	V1	V2	V3	...	V14
ind 1	a	**e**	g	...	u
ind 2	**c**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**g**		v
...	...	...	...		...
ind 1232	c	f	h		v

	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	**0.71**	**0.29**	1	0	...
ind 2	**0.12**	**0.29**	**0.59**	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	**0.37**	**0.63**	...
...	...	...	...	...	...	...	...	...
ind 1232	0	0	1	0	1	0	1	...

Two ways to impute categories: majority or draw

Public Assistance - Paris Hospitals

Traumabase: 15000 patients/ 250 variables/ 8 hospitals

```
             Center       Accident Age Sex Weight Height   BMI  BP SBP
1           Beaujon         Fall    54   m     85     NR    NR 180 110
2             Lille         Other   33   m     80    1.8 24.69 130  62
3  Pitie Salpetriere        Gun     26   m     NR     NR    NR 131  62
4           Beaujon      AVP moto   63   m     80    1.8 24.69 145  89
6  Pitie Salpetriere   AVP bicycle  33   m     75     NR    NR 104  86
7  Pitie Salpetriere AVP pedestrian 30   w     NR     NR    NR 107  66
9              HEGP   White weapon  16   m     98   1.92 26.58 118  54
10           Toulon  White weapon  20   m     NR     NR    NR 124  73
11          Bicetre         Fall    61   m     84    1.7 29.07 144 105
...................

   SpO2 Temperature Lactates   Hb  Glasgow Transfusion ...........
1    97        35.6     <NA> 12.7       12         yes
2   100        36.5      4.8 11.1       15         no
3   100          36      3.9 11.4        3         no
4   100        36.7     1.66   13       15         yes
6   100          36       NM 14.4       15         no
7   100        36.6       NM 14.3       15         yes
9   100        37.5       13 15.9       15         yes
10  100        36.9       NM 13.7       15         no
11  100        36.6      1.2 14.2       14         no
...........
```

Imputed Paris Hospitals data

Traumabase: 15000 patients/ 250 variables/ 8 hospitals

```
              Center       Accident   Age Sex Weight Height   BMI BP SBP
1            Beaujon       Fall         54    m 85.00   1.84 27.04  83  13
2              Lille       Other        33    m 80.00   1.80 24.69  33  98
3  Pitie Salpetriere       Gun          26    m 81.78   1.85 24.33  34  98
4            Beaujon       AVP moto     63    m 80.00   1.80 24.69  48 125
6  Pitie Salpetriere       AVP bicycle  33    m 75.00   1.83 24.53   6 122
7  Pitie Salpetriere       AVP pedestri 30    m 81.89   1.82 25.24   9 102
9               HEGP       White weapon 16    m 98.00   1.92 26.58  21  90
10            Toulon       White weapon 20    m 81.68   1.82 25.05  27 109
11           Bicetre       Fall         61    m 84.00   1.70 29.07  47   8
   SpO2 Temperature Lactates Hb Glasgow...........
1    46          61   289.07 33      14
2     2          72   464.00 16      14
3     2          65   416.00 19       7
4     2          74   130.00 36       6
6     2          65   285.91 50       6
7     2          73   244.99 49       6
9     2          83   196.00 65       6
10    2          76   262.44 43       6
11    2          73    84.00 48       5
```

Design

The simulated data:

- $X_{i_{(k_i \times J)}} = 1_{k_i} m' + 1_{k_i} F_i^{b'} V^{b'} + F_i^{w} V^{w'} + E_i$, with $E_{ijk_i} \sim \mathcal{N}(0, \sigma)$
- 5 groups, 10 variables, $Q_b = 2$, $Q_w = 2$

Many scenarios are considered:

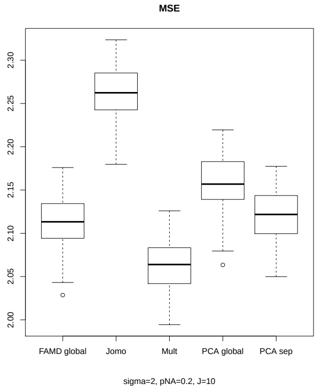
- number of individuals per group: 10-20, 70-100
- level of noise: low, strong
- percentage of missing values: 10%, 25%, 40%
- missing values mechanism: MCAR, MAR

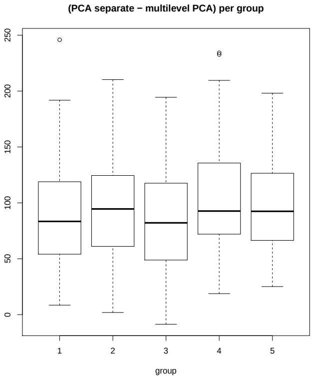
$\Rightarrow$ Prediction error: $\frac{1}{KJ} \sum (x_{ijk_i} - \hat{x_{ijk_i}})^2$

Results

Competitors:

- Conditional model with random effect regression (mice)
- Random forests (bühlmann, 2012) (not designed)
- Global PCA - Separate PCA
- Global mixed PCA (with hospital)





Results

	$J = 10$	$J = 30$	$5cat$	$5cat$
Global PCA	0.09	0.3		
mice	11	282		
Multilevel SVD	1.5	1.2	2	7
Global mixed PCA	0.4	0.7	1	4
Random forest	59	200	27	246

Table 1: Time in seconds for a dataset with 20% NA, $I = 5$ $k_i = 200$

- PCA mixed as Random Forest
- mice (random effect model): difficulties with large dimensions
- Separate PCA: pb with many missing values
- Multilevel SVD = global SVD when no group effect
- Imputation properties depends on the method (linear)
- Other methods do not handle categorical variables

Distribution

Combining data from different institutional databases promises many advantages in personalizing medical care (large n, more chance for finding patients like me)

$\Rightarrow$ NIH requires sharing of data from funded projects

Distribution

Combining data from different institutional databases promises many advantages in personalizing medical care (large n, more chance for finding patients like me)

$\Rightarrow$ NIH requires sharing of data from funded projects

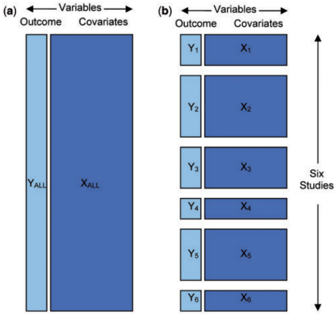
$\Rightarrow$ The problem: high barriers to aggregation of medical data

- lack of standardization of ontologies
- privacy concerns
- proprietary attitude towards data, reluctance to cede control
- complexity/size of aggregated data, updates problems

Solution: distributed computation

⇒ Data aggregation is not always necessary

⇒ NIH splits the storage of aggregated data across several centers

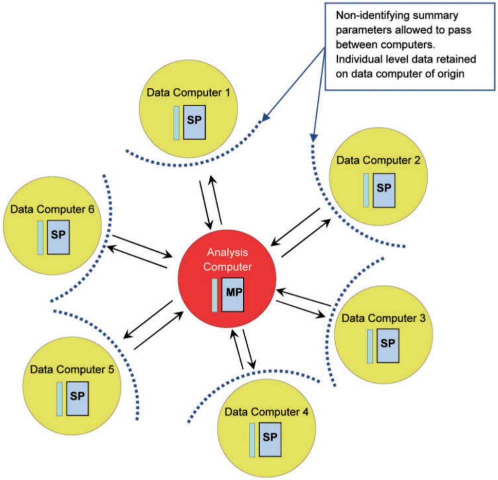


⇒ Data can stay at site

⇒ Computations can be distributed (share burden)

⇒ Hospitals only share intermediate results instead of the raw data

Topology: master-workers (Wolfson, et. al (2010))



⇒ Ex: Each site share the sum of age $\tilde{X}_i$ and the number of patients n_i. The master computes $\bar{X} = \sum n_i \tilde{X}_i / \sum n_i$

Solution: distributed computation

$\Rightarrow$ Many models fitting can be implemented:

- Maximizing a likelihood. Intermediate computations break up into sums of quantities computed on local data at sites. Log-likelihood, score function and Fisher information can partition into sums. (OK for logistic regression)
- Singular Value Decomposition. Iterative algorithms available for SVD using quantities computed on local data at sites.
- And more.

Implemented in the R package discomp (Narasimhan et. al., 2017)

Singular value decomposition

SVD: $X_{n\times p} : U_{n\times k} D_{k\times k} V'_{p\times k}$

Power method to get the first direction:

Data: $\boldsymbol{X} \in \mathcal{R}^{n\times p}$
Result: $u \in \mathcal{R}^n,\ v \in \mathcal{R}^p$, and $d > 0$
$u \leftarrow (\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}, \dots, \frac{1}{\sqrt{n}})$;
repeat
  $v \leftarrow \boldsymbol{X}^\top u$;
  $v \leftarrow v/\|v\|$;
  $u \leftarrow \boldsymbol{X} v$;
  $d \leftarrow \|u\|$;
  $u \leftarrow u/\|u\|$;
until *convergence*;

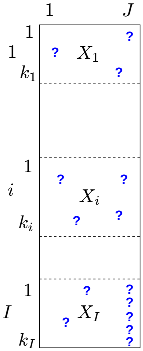
Other dims: "deflation", same procedure in the residuals $(X - udv')$

$\Rightarrow$ Involves inner products and sums: distributed

Privacy preserving rank k SVD

Data: each worker has private data $\boldsymbol{X}_i \in \mathcal{R}^{n_i \times p}$
Result: $V \in \mathcal{R}^{p \times k}$, and $d_1 \geq \ldots d_k \geq 0$
$V \leftarrow 0,\ d \leftarrow 0$ **foreach** *worker site* j **do**
    $U^{[j]} = 0$;
    transmit n_j to master;
end
for $i \leftarrow 1$ **to** k **do**
    foreach *worker site* j **do** $u^{[j]} \leftarrow (1, 1, \ldots, 1)$ of length n_j;
    $\|u\| \leftarrow \sqrt{\sum_j n_j}$;
    transmit $\|u\|, V$, and D to workers;
    repeat
        foreach *worker site* j **do**
            $u^{[j]} \leftarrow u^{[j]}/\|u\|$;
            calculate $v^{[j]} \leftarrow (\boldsymbol{X}^{[j]} - U^{[j]} D V^\top)^\top u^{[j]}$;
            transmit $v^{[j]}$ to master;
        end
        $v \leftarrow \sum_j v^{[j]}$;
        $v \leftarrow v/\|v\|$;
        transmit v to workers;
        foreach *worker site* j **do**
            calculate $u^{[j]} \leftarrow \boldsymbol{X}^{[j]} v$;
            transmit $\|u^{[j]}\|$ to master;
        end
        $\|u\| \leftarrow \sum_j \|u^{[j]}\|$;
        transmit $\|u\|$ to workers;
        $d_i \leftarrow \|u\|$;
    until *convergence*;
    $V \leftarrow$ `cbind`(V, v);
    foreach *worker site* j **do** $U^{[j]} \leftarrow$ `cbind`$(U^{[j]}, u^{[j]})$;
end

Multilevel imputation



⇒ Impute multilevel data with Multilevel SVD

⇒ Distributed multilevel imputation

⇒ Impute the data of one hospital using the data of the others

⇒ Incentive to encourage the hospitals to participate in the project

⇒ Apply other statistical methods on the imputed data (logistic regression)

Take home message - On going work

Multilevel PCA powerful for single imputation of continuous & categorical multilevel data: reduce the dimensionality - capture the similarities between rows and relationship between variables at both levels.

Method without missing values

⇒ Computationaly fast - distributed - Implemented R package missMDA

- Numbers of components Q_b and Q_w ?

- Inference after imputation. Underestimation of the variance with single imputation

Take home message - On going work

Multilevel PCA powerful for single imputation of continuous & categorical multilevel data: reduce the dimensionality - capture the similarities between rows and relationship between variables at both levels.

Method without missing values

⇒ Computationaly fast - distributed - Implemented R package missMDA

- Numbers of components Q_b and Q_w ?

cross-validation?

- Inference after imputation. Underestimation of the variance with single imputation

Take home message - On going work

Multilevel PCA powerful for single imputation of continuous & categorical multilevel data: reduce the dimensionality - capture the similarities between rows and relationship between variables at both levels.

Method without missing values

⇒ Computationaly fast - distributed - Implemented R package missMDA

• Numbers of components Q_b and Q_w ?
cross-validation?

• Inference after imputation. Underestimation of the variance with single imputation
Multiple imputation: bootstrap + drawn from the predictive distribution
$\mathcal{N}\left(\mathbf{1}_K \hat{m}' + \hat{F}^b \hat{B}^{b'} + \hat{F}^w \hat{B}^{w'}, \hat{\sigma}^2\right)$

more...

Low-rank model with covariates for count data with missing values (2019, Journal of Multivariate Analysis) Geneviève Robin, Julie Josse, Éric Moulines and Sylvain Sardy

https://genevieverobin.files.wordpress.com/2019/08/presentation_grobin.pdf

Main effects and interactions in mixed and incomplete data frames (2019, Journal of American Statistical Association) Geneviève Robin, Olga Klopp, Julie Josse, Éric Moulines and Robert Tibshirani

Outline

Ignorable

X has a density, parametrized by θ that we want to estimate
$f(X, M|\theta, \phi)$ the joint distribution
ML estimate:

$$\begin{aligned} f(X_{\text{obs}}, M; \theta, \phi) &= \int f(X_{\text{obs}}, X_{\text{mis}}, M; \theta, \phi) dX_{\text{mis}} \\ &= \int f(X_{\text{obs}}, X_{\text{mis}}; \theta) f(M|X_{\text{obs}}, X_{\text{miss}}; \phi) dX_{\text{mis}}. \end{aligned}$$

When MAR

$$\begin{aligned} f(X_{\text{obs}}, M; \theta, \phi) &= \int f(X_{\text{obs}}, X_{\text{mis}}; \theta) f(M|X_{\text{obs}}; \phi) dX_{\text{mis}}, \\ &= f(M|X_{\text{obs}}; \phi) \int f(X_{\text{obs}}, X_{\text{miss}}; \theta) dX_{\text{miss}}, \end{aligned}$$

$$f(X_{\text{obs}}, M; \theta, \phi) = f(M|X_{\text{obs}}; \phi) f(X_{\text{obs}}; \theta).$$

Expectation - Maximization (Dempster *et al.*, 1977)

Rationale to get ML estimates: max L_{obs} through max of L_{comp} of $X = (X_{obs}, X_{miss})$. Augment the data to simplify the problem.

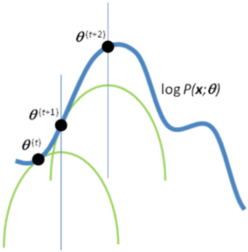
E step (conditional expectation):

$$Q(\theta, \theta^{\ell}) = \int \ln(f(X;\theta)) f(X_{miss}|X_{obs};\theta^{\ell}) dX_{miss}$$

M step (maximization):

$$\theta^{\ell+1} = \text{argmax}_{\theta} Q(\theta, \theta^{\ell})$$

Result: when $\theta^{\ell+1}$ max $Q(\theta, \theta^{\ell})$ then $L(X_{obs}, \theta^{\ell+1}) \geq L(X_{obs}, \theta^{\ell})$.



Maximum likelihood approach

Ex: Hypothesis $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$

$\Rightarrow$ Point estimates with EM:

```
> library(norm)
> pre <- prelim.norm(as.matrix(don))
> thetahat <- em.norm(pre)
> getparam.norm(pre,thetahat)
```

Exercice: EM with bivariate data

Maximum likelihood approach

Ex: Hypothesis $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$

$\Rightarrow$ Point estimates with EM:

```
> library(norm)
> pre <- prelim.norm(as.matrix(don))
> thetahat <- em.norm(pre)
> getparam.norm(pre,thetahat)
```

Exercice: EM with bivariate data

$\Rightarrow$ Variances:

- Supplemented EM (Meng, 1991), Louis formulae
- Bootstrap approach:
  - Bootstrap rows: X^1, ... , X^B
  - EM algorithm: $(\hat{\mu}^1, \hat{\Sigma}^1)$, ... , $(\hat{\mu}^B, \hat{\Sigma}^B)$

Maximum likelihood approach

Ex: Hypothesis $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$

$\Rightarrow$ Point estimates with EM:

```
> library(norm)
> pre <- prelim.norm(as.matrix(don))
> thetahat <- em.norm(pre)
> getparam.norm(pre,thetahat)
```

Exercice: EM with bivariate data

$\Rightarrow$ Variances:

- Supplemented EM (Meng, 1991), Louis formulae
- Bootstrap approach:
  - Bootstrap rows: X^1, ... , X^B
  - EM algorithm: $(\hat{\mu}^1, \hat{\Sigma}^1)$, ... , $(\hat{\mu}^B, \hat{\Sigma}^B)$

Other models: SAEM (SAEM for logistic regression)

Logistic regression with missing covariates: Parameter estimation, model selection and prediction (Jiang, J., Lavielle, Gauss, Hamada, 2018)

$x = (x_{ij})$ a $n \times d$ matrix of quantitative covariates
$y = (y_i)$ an n-vector of binary responses $\{0, 1\}$

Logistic regression model: $\mathbb{P}(y_i = 1 | x_i; \beta) = \frac{\exp(\beta_0 + \sum_{j=1}^{d} \beta_j x_{ij})}{1 + \exp(\beta_0 + \sum_{j=1}^{d} \beta_j x_{ij})}$

Covariables: $x_i \underset{\text{i.i.d.}}{\sim} \mathcal{N}_d(\mu, \Sigma)$

Log-likelihood with $\theta = (\mu, \Sigma, \beta)$:

$\mathcal{LL}(\theta; x, y) = \sum_{i=1}^{n} \Big(\log(\mathrm{p}(y_i | x_i; \beta)) + \log(\mathrm{p}(x_i; \mu, \Sigma)) \Big).$

Logistic regression with missing covariates: Parameter estimation, model selection and prediction (Jiang, J., Lavielle, Gauss, Hamada, 2018)

$x = (x_{ij})$ a $n \times d$ matrix of quantitative covariates
$y = (y_i)$ an n-vector of binary responses $\{0, 1\}$

Logistic regression model: $\mathbb{P}(y_i = 1 | x_i; \beta) = \frac{\exp(\beta_0 + \sum_{j=1}^{d} \beta_j x_{ij})}{1 + \exp(\beta_0 + \sum_{j=1}^{d} \beta_j x_{ij})}$

Covariables: $x_i \underset{\text{i.i.d.}}{\sim} \mathcal{N}_d(\mu, \Sigma)$

Log-likelihood with $\theta = (\mu, \Sigma, \beta)$:

$\mathcal{LL}(\theta; x, y) = \sum_{i=1}^{n} \Big(\log(\mathrm{p}(y_i | x_i; \beta)) + \log(\mathrm{p}(x_i; \mu, \Sigma)) \Big).$

X_1	X_2	X_3	...	Y
NA	20	10	...	shock
-6	45	NA	...	shock
0	NA	30	...	no shock
NA	32	35	...	shock
1	63	40	...	shock
-2	NA	12	...	no shock

Likelihood inference with Missing At Random values

$$\mathcal{LL}(\theta; x, y) = \sum_{i=1}^{n} \Big(\log(\mathrm{p}(y_i|x_i;\beta)) + \log(\mathrm{p}(x_i;\mu,\Sigma)) \Big)$$

X_1	X_2	X_3	...	M_1	M_2	M_3	...	Y
NA	20	10	...	1	0	0	...	shock
-6	45	NA	...	0	0	1	...	shock
0	NA	30	...	0	1	0	...	no shock
NA	32	35	...	1	0	0	...	shock

$m = (m_{ij})$ a $n \times d$ matrix $m_{ij} = 0$ if x_{ij} is observed and 1 otherwise

$(y_i, x_i, m_i) \underset{\text{i.i.d.}}{\sim} \{p_\theta(x, y) f_\phi(m \,|\, x, y)\}$ data & missing values mechanism

Likelihood inference with Missing At Random values

$$\mathcal{LL}(\theta; x, y) = \sum_{i=1}^{n} \Big(\log(\mathrm{p}(y_i|x_i;\beta)) + \log(\mathrm{p}(x_i;\mu,\Sigma)) \Big)$$

X_1	X_2	X_3	...	M_1	M_2	M_3	...	Y
NA	20	10	...	1	0	0	...	shock
-6	45	NA	...	0	0	1	...	shock
0	NA	30	...	0	1	0	...	no shock
NA	32	35	...	1	0	0	...	shock

$m = (m_{ij})$ a $n \times d$ matrix $m_{ij} = 0$ if x_{ij} is observed and 1 otherwise

$(y_i, x_i, m_i) \underset{\text{i.i.d.}}{\sim} \{p_\theta(x, y) f_\phi(m \,|\, x, y)\}$ data & missing values mechanism

Ex: Income & Age with missing values on income

MAR: depends only on observed values, i.e. on age (not income)

Ignorable mechanism $\mathcal{L}_{obs}(\theta) \triangleq \prod_{i=1}^{n} \int p_\theta(x_i, y_i) dx_{i,mis}$

Stochastic Approximation EM - package misaem

argmax $\mathcal{LL}(\theta; x_{\text{obs}}, y) = \int \mathcal{LL}(\theta; x, y) dx_{\text{mis}}$

- **E-step:** Evaluate the quantity

$$Q_k(\theta) = \mathbb{E}[\mathcal{LL}(\theta; x, y)|x_{\text{obs}}, y; \theta_{k-1}]$$
$$= \int \mathcal{LL}(\theta; x, y)\mathrm{p}(x_{\text{mis}}|x_{\text{obs}}, y; \theta_{k-1}) dx_{\text{mis}}$$

- **M-step:** $\theta_k = \text{argmax}_\theta \, Q_k(\theta)$

$\Rightarrow$ *Unfeasible computation of expectation*

MCEM (Wei & Tanner, 1990): Generate samples of missing data from $\mathrm{p}(x_{\text{mis}}|x_{\text{obs}}, y; \theta_{k-1})$ and replace the expectation by an empirical mean

$\Rightarrow$ *Require a huge number of samples*

SAEM (Lavielle, 2014) almost sure convergence to MLE

Unbiased estimates: $\hat{\beta}_1, \ldots, \hat{\beta}_d$ - $\hat{V}(\hat{\beta}_1), \ldots, \hat{V}(\hat{\beta}_d)$ - good coverage

Stochastic Approximation EM

(book, Lavielle 2014) Starting from an initial guess θ_0, the kth iteration consists of three steps:

- **Simulation:** For $i = 1, 2, \cdots, n$, draw one sample $x_{i,\mathrm{mis}}^{(k)}$ from

$$\mathrm{p}(x_{i,\mathrm{mis}} | x_{i,\mathrm{obs}}, y_i; \theta_{k-1}).$$

- **Stochastic approximation:** Update the function Q

$$Q_k(\theta) = Q_{k-1}(\theta) + \gamma_k \left(\mathcal{LL}(\theta; x_{\mathrm{obs}}, x_{\mathrm{mis}}^{(k)}, y) - Q_{k-1}(\theta) \right),$$

  where (γ_k) is a decreasing sequence of positive numbers.

- **Maximization:** $\theta_k = \mathrm{argmax}_\theta \, Q_k(\theta)$.

Stochastic Approximation EM

(book, Lavielle 2014) Starting from an initial guess θ_0, the kth iteration consists of three steps:

- **Simulation:** For $i = 1, 2, \cdots, n$, draw one sample $x_{i,\mathrm{mis}}^{(k)}$ from

$$\mathrm{p}(x_{i,\mathrm{mis}} | x_{i,\mathrm{obs}}, y_i; \theta_{k-1}).$$

- **Stochastic approximation:** Update the function Q

$$Q_k(\theta) = Q_{k-1}(\theta) + \gamma_k \left(\mathcal{LL}(\theta; x_{\mathrm{obs}}, x_{\mathrm{mis}}^{(k)}, y) - Q_{k-1}(\theta) \right),$$

  where (γ_k) is a decreasing sequence of positive numbers.

- **Maximization:** $\theta_k = \mathrm{argmax}_\theta \, Q_k(\theta)$.

Convergence: (Allassonniere et al. 2010)
The choice of the sequence (γ_k) is important for ensuring the almost sure convergence of SAEM to a MLE.

Metropolis-Hastings algorithm

Target distribution

$$f_i(x_{i,\mathrm{mis}}) = \mathrm{p}(x_{i,\mathrm{mis}}|x_{i,\mathrm{obs}}, y_i; \theta)$$
$$\propto \mathrm{p}(y_i|x_i; \beta)\, \mathrm{p}(x_{i,\mathrm{mis}}|x_{i,\mathrm{obs}}; \mu, \Sigma).$$

Metropolis-Hastings algorithm

Target distribution

$$f_i(x_{i,\mathrm{mis}}) = \mathrm{p}(x_{i,\mathrm{mis}} | x_{i,\mathrm{obs}}, y_i; \theta)$$
$$\propto \mathrm{p}(y_i | x_i; \beta)\, \mathrm{p}(x_{i,\mathrm{mis}} | x_{i,\mathrm{obs}}; \mu, \Sigma).$$

Proposal distribution $g_i(x_{i,\mathrm{mis}}) = \mathrm{p}(x_{i,\mathrm{mis}} | x_{i,\mathrm{obs}}; \mu, \Sigma)$

$$x_{i,\mathrm{mis}} | x_{i,\mathrm{obs}} \sim \mathcal{N}_p(\mu_i, \Sigma_i)$$

$$\mu_i = \mu_{i,\mathrm{mis}} + \Sigma_{i,\mathrm{mis},\mathrm{obs}} \Sigma^{-1}_{i,\mathrm{obs},\mathrm{obs}} (x_{i,\mathrm{obs}} - \mu_{i,\mathrm{obs}}),$$
$$\Sigma_i = \Sigma_{i,\mathrm{mis},\mathrm{mis}} - \Sigma_{i,\mathrm{mis},\mathrm{obs}} \Sigma^{-1}_{i,\mathrm{obs},\mathrm{obs}} \Sigma_{i,\mathrm{obs},\mathrm{mis}},$$

Metropolis-Hastings algorithm

Target distribution

$$\begin{aligned} f_i(x_{i,\mathrm{mis}}) &= \mathrm{p}(x_{i,\mathrm{mis}}|x_{i,\mathrm{obs}}, y_i; \theta) \\ &\propto \mathrm{p}(y_i|x_i; \beta)\, \mathrm{p}(x_{i,\mathrm{mis}}|x_{i,\mathrm{obs}}; \mu, \Sigma). \end{aligned}$$

Proposal distribution $g_i(x_{i,\mathrm{mis}}) = \mathrm{p}(x_{i,\mathrm{mis}}|x_{i,\mathrm{obs}}; \mu, \Sigma)$

$$x_{i,\mathrm{mis}}|x_{i,\mathrm{obs}} \sim \mathcal{N}_p(\mu_i, \Sigma_i)$$

$$\begin{aligned} \mu_i &= \mu_{i,\mathrm{mis}} + \Sigma_{i,\mathrm{mis},\mathrm{obs}}\Sigma^{-1}_{i,\mathrm{obs},\mathrm{obs}}(x_{i,\mathrm{obs}} - \mu_{i,\mathrm{obs}}), \\ \Sigma_i &= \Sigma_{i,\mathrm{mis},\mathrm{mis}} - \Sigma_{i,\mathrm{mis},\mathrm{obs}}\Sigma^{-1}_{i,\mathrm{obs},\mathrm{obs}}\Sigma_{i,\mathrm{obs},\mathrm{mis}}, \end{aligned}$$

Metropolis

- $z_{im}^{(k)} \sim g_i(x_{i,mis}),\ u \sim \mathcal{U}[0,1]$
- $r = \frac{f_i(z_{im}^{(k)})/g_i(z_{im}^{(k)})}{f_i(z_{i,m-1}^{(k)})/g_i(z_{i,m-1}^{(k)})}$
- If $u < r$, accept $z_{im}^{(k)}$

Only need a few steps of Markov chains in each iteration of SAEM!

Variance estimation

Observed Fisher information matrix (FIM) *wrt* β

$$\mathcal{I}(\theta) = -\frac{\partial^2 \mathcal{LL}(\theta; x_{\text{obs}}, y)}{\partial\theta\partial\theta^T}.$$

Variance estimation

Observed Fisher information matrix (FIM) *wrt* β

$$\mathcal{I}(\theta) = -\frac{\partial^2 \mathcal{LL}(\theta; x_{\text{obs}}, y)}{\partial\theta\partial\theta^T}.$$

Louis formula

$$\begin{aligned}
\mathcal{I}(\theta) = & -\mathbb{E}\left(\frac{\partial^2 \mathcal{LL}(\theta; x, y)}{\partial\theta\partial\theta^T}\Big| x_{\text{obs}}, y; \theta\right) \\
& -\mathbb{E}\left(\frac{\partial \mathcal{LL}(\theta; x, y)}{\partial\theta}\frac{\partial \mathcal{LL}(\theta; x, y)^T}{\partial\theta}\Big| x_{\text{obs}}, y; \theta\right) \\
& +\mathbb{E}\left(\frac{\partial \mathcal{LL}(\theta; x, y)}{\partial\theta}\Big| x_{\text{obs}}, y; \theta\right)\mathbb{E}\left(\frac{\partial \mathcal{LL}(\theta; x, y)}{\partial\theta}\Big| x_{\text{obs}}, y; \theta\right)^T.
\end{aligned}$$

Given the MH samples of unobserved data $(x_{i,\text{mis}}^{(m)}, 1 \leq i \leq n, 1 \leq m \leq M)$, and the SAEM estimate $\hat{\theta}$

$\Rightarrow$ Estimate FIM by empirical means.

Model selection : criterion BIC

With $\tilde{p}_\theta$ the number of estimated parameters in a given model $\mathcal{M}$, model selection criterion (*penalized likelihood*) :

$$\mathrm{BIC}(\mathcal{M}) = -2\mathcal{L}\mathcal{L}(\hat{\theta}_{\mathcal{M}}; x_{\mathrm{obs}}, y) + \log(n) d(\mathcal{M}),$$

How to estimate *observed likelihood* ?

Model selection : criterion BIC

With $\tilde{p}_\theta$ the number of estimated parameters in a given model $\mathcal{M}$, model selection criterion (*penalized likelihood*) :

$$\mathrm{BIC}(\mathcal{M}) = -2\mathcal{LL}(\hat{\theta}_{\mathcal{M}}; x_{\mathrm{obs}}, y) + \log(n) d(\mathcal{M}),$$

How to estimate *observed likelihood* ?

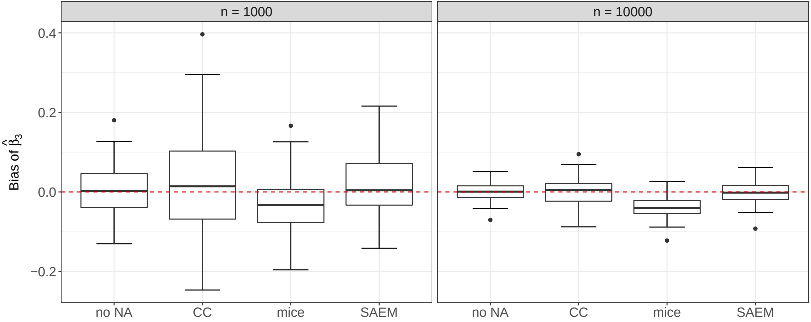
$$\begin{aligned}
\mathrm{p}(y_i, x_{i,\mathrm{obs}}; \theta) &= \int \mathrm{p}(y_i, x_{i,\mathrm{obs}} | x_{i,\mathrm{mis}}; \theta) \mathrm{p}(x_{i,\mathrm{mis}}; \theta) dx_{i,\mathrm{mis}} \\
&= \int \mathrm{p}(y_i, x_{i,\mathrm{obs}} | x_{i,\mathrm{mis}}; \theta) \frac{\mathrm{p}(x_{i,\mathrm{mis}}; \theta)}{g_i(x_{i,\mathrm{mis}})} g_i(x_{i,\mathrm{mis}}) dx_{i,\mathrm{mis}} \\
&= \mathbb{E}_{g_i} \left(\mathrm{p}(y_i, x_{i,\mathrm{obs}} | x_{i,\mathrm{mis}}; \theta) \frac{\mathrm{p}(x_{i,\mathrm{mis}}; \theta)}{g_i(x_{i,\mathrm{mis}})} \right).
\end{aligned}$$

Sample from g_i (the proposal distribution in SAEM)

$\Rightarrow$ Empirical mean.

Comparison with competitors: estimates

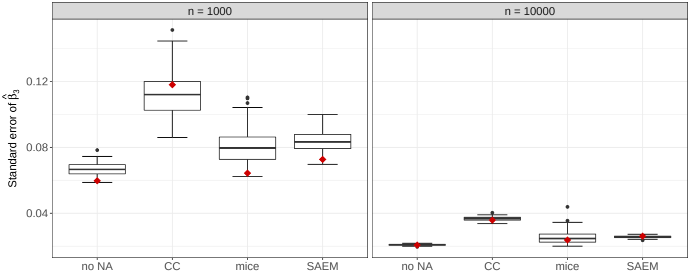
x: $p = 5$, $n = 1000$ / $n = 10\,000 \Rightarrow y \in \{0, 1\}$
percentage of missingness $= 10\%$.
Repeat 1000 times for each setting.



Comparison with competitors : coverage

Table 2: Coverage (%) for $n = 10\,000$, calculated over 1000 simulations.

parameter	no NA	CC	mice	SAEM
β_0	95.2	94.4	95.2	94.9
β_1	96.0	94.7	93.9	95.1
β_2	95.5	94.6	94.0	94.3
β_3	94.9	94.3	*86.5*	94.7
β_4	94.6	94.2	96.2	95.4
β_5	95.9	94.4	*89.6*	94.7



Comparison with competitors : execution time

Table 3: Comparison of execution time between no NA, MCEM, mice, and SAEM with $n = 1000$ calculated over 1000 simulations.

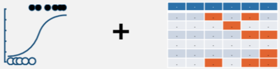
Execution time (seconds)	no NA	MCEM	mice	SAEM
min	2.87×10^{-3}	492	0.64	9.96
mean	4.65×10^{-3}	773	0.70	13.50
max	43.50×10^{-3}	1077	0.76	16.79

Application on TraumaBase

- Age
- Weight
- Height
- BMI
- Glasgow
- Systolic BP
- Diastolic BP
- Heart Rate
- Hb Hemocue
- SpO_2
- Volume Expander
- Pulse Pressure

- 14 continuous variables
- Gaussian distribution assumption

Logistic regression with missing values



Hemorrhagic shock

$P(y = 1 \mid X; \hat{\beta})$?

Exploration of dataset

Data preprocessing ⇒ *6384 patients* in the dataset.

Clinical experience ⇒ *14 influential quantitative measurements*

The percentage of missingness of some variables varies form 0 to *60%*, which indicates the importance of analysis of missing data.

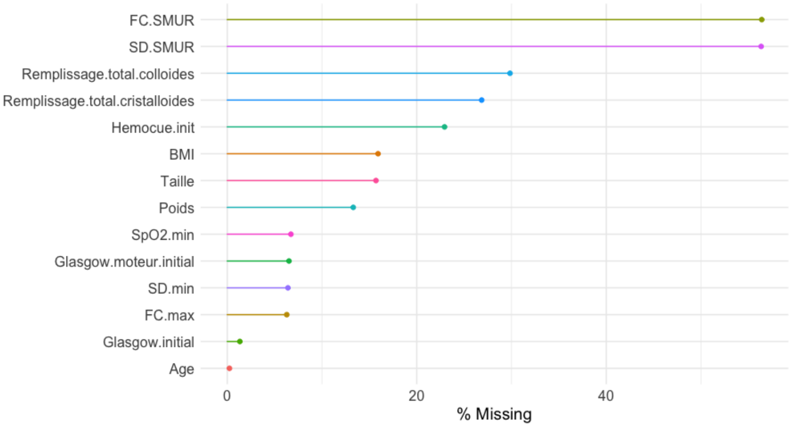


Figure 3: Percentage of missing information in each variable.

Exploration of dataset

Based on *penalized observed log-likelihood*

⇒ Observations resulting in a very small value of the log-likelihood.

⇒ wrong records

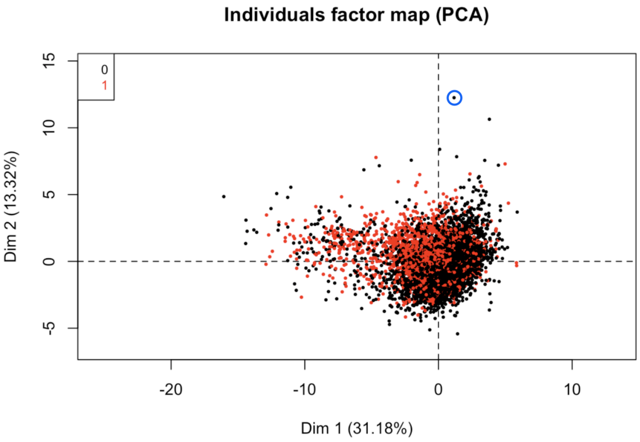


Figure 4: Individual factor map of PCA; Blue circle remarks the outlier; Red

Estimation and interpretation

Estimation and model selection:

Variable	Effect	Estimate (std error)
Intercept		-0.52 (0.59)
Age	+	0.011 (0.0033)
Glasgow.moteur	-	-0.16 (0.036)
FC.max	+	0.026 (0.0025)
Hemocue.init	-	-0.23 (0.031)
RT.cristalloides	+	0.00090 (0.00010)
RT.colloides	+	0.0019 (0.00021)
SD.min	-	-0.025 (0.0050)
SD.SMUR	-	-0.021 (0.0056)

Estimation and interpretation

Estimation and model selection:

Variable	Effect	Estimate (std error)
Intercept		-0.52 (0.59)
Age	+	0.011 (0.0033)
Glasgow.moteur	-	-0.16 (0.036)
FC.max	+	0.026 (0.0025)
Hemocue.init	-	-0.23 (0.031)
RT.cristalloides	+	0.00090 (0.00010)
RT.colloides	+	0.0019 (0.00021)
SD.min	-	-0.025 (0.0050)
SD.SMUR	-	-0.021 (0.0056)

- Older people tend to have a larger possibility to suffer from hemorrhagic shock.
- A low Glasgow score means one makes no motor response, often in the case of hemorrhagic shock.

Outline

On the consistency of supervised learning with missing values. (2019). J., Prost, Scornet & Varoquaux

- A feature matrix $\mathbf{X}$ and a response vector Y
- Find a prediction function that minimizes the expected risk

  Bayes rule: $f^\star \in \underset{f:\, \mathcal{X} \rightarrow \mathcal{Y}}{\operatorname{argmin}} \mathbb{E}\left[\ell(f(\mathbf{X}), Y)\right]; \quad f^\star(\mathbf{X}) = \mathbb{E}[Y|\mathbf{X}]$
- Empirical risk: $\hat{f}_{\mathcal{D}_{n,\mathrm{train}}} \in \underset{f:\mathcal{X} \rightarrow \mathcal{Y}}{\operatorname{argmin}} \left(\frac{1}{n}\sum_{i=1}^{n} \ell\left(f(\mathbf{X}_i), Y_i\right)\right)$

  A new data $\mathcal{D}_{n,\mathrm{test}}$ to estimate the generalization error rate
- Bayes consistent: $\mathbb{E}[\ell(\hat{f}_n(\mathbf{X}), Y)] \xrightarrow[n \to \infty]{} \mathbb{E}[\ell(f^\star(\mathbf{X}), Y)]$

On the consistency of supervised learning with missing values. (2019). J., Prost, Scornet & Varoquaux

- A feature matrix $\mathbf{X}$ and a response vector Y
- Find a prediction function that minimizes the expected risk

  Bayes rule: $f^\star \in \underset{f:\mathcal{X}\rightarrow\mathcal{Y}}{\operatorname{argmin}} \mathbb{E}\left[\ell(f(\mathbf{X}), Y)\right]; \quad f^\star(\mathbf{X}) = \mathbb{E}[Y|\mathbf{X}]$

- Empirical risk: $\hat{f}_{\mathcal{D}_{n,\text{train}}} \in \underset{f:\mathcal{X}\rightarrow\mathcal{Y}}{\operatorname{argmin}} \left(\frac{1}{n}\sum_{i=1}^{n} \ell\left(f(\mathbf{X}_i), Y_i\right)\right)$

  A new data $\mathcal{D}_{n,\text{test}}$ to estimate the generalization error rate

- Bayes consistent: $\mathbb{E}[\ell(\hat{f}_n(\mathbf{X}), Y)] \xrightarrow[n\to\infty]{} \mathbb{E}[\ell(f^\star(\mathbf{X}), Y)]$

Differences with classical litterature

- explicitely consider the response variable Y - Aim: Prediction
- two data sets (out of sample) with missing values: Train & test sets

$\Rightarrow$ Is it possible to use previous approaches (EM - impute), consistent?

$\Rightarrow$ Do we need to design new ones?

EM and out-of sample prediction - package misaem

$\mathbb{P}(y_i = 1 | x_i; \beta) = \frac{\exp^{(X\beta)}}{1+\exp^{(X\beta)}}$ After EM: $\hat{\theta}_n = (\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_d, \hat{\mu}, \hat{\Sigma})$

New obs: $x_{n+1} = (x_{(n+1)1}, \mathrm{NA}, \mathrm{NA}, x_{(n+1)4}, \ldots, x_{(n+1)d})$

Predict Y on **a test set with missing entries** $x_{\text{test}} = (x_{obs}, x_{miss})$

EM and out-of sample prediction - package misaem

$\mathbb{P}(y_i = 1 | x_i; \beta) = \frac{\exp^{(X\beta)}}{1+\exp^{(X\beta)}}$ After EM: $\hat{\theta}_n = (\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_d, \hat{\mu}, \hat{\Sigma})$

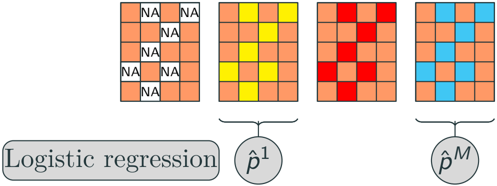
New obs: $x_{n+1} = (x_{(n+1)1}, \mathrm{NA}, \mathrm{NA}, x_{(n+1)4}, \ldots, x_{(n+1)d})$

Predict Y on **a test set with missing entries** $x_{\text{test}} = (x_{obs}, x_{miss})$

$$\hat{y} = \text{argmax}_y \, \mathrm{p}_{\hat{\theta}}(y | x_{\mathrm{obs}}) = \text{argmax}_y \int \mathrm{p}_{\hat{\theta}}(y|x) \mathrm{p}_{\hat{\theta}}(x_{\mathrm{mis}} | x_{\mathrm{obs}}) dx_{\mathrm{mis}}$$

$$= \text{argmax}_y \, \mathbb{E}_{\mathrm{p}_{\mathbf{X}_m | X_o = \mathbf{x}_o}} \mathrm{p}_{\hat{\theta}_n}(y | X_m, \mathbf{x}_o) \approx \text{argmax}_y \sum_{m=1}^{M} \mathrm{p}_{\hat{\theta}_n}\left(y | x_{\mathrm{obs}}, x_{\mathrm{mis}}^{(m)}\right)$$

$\approx$ Multiple imputation: Draw M values from $X_{miss} | X_{obs}$

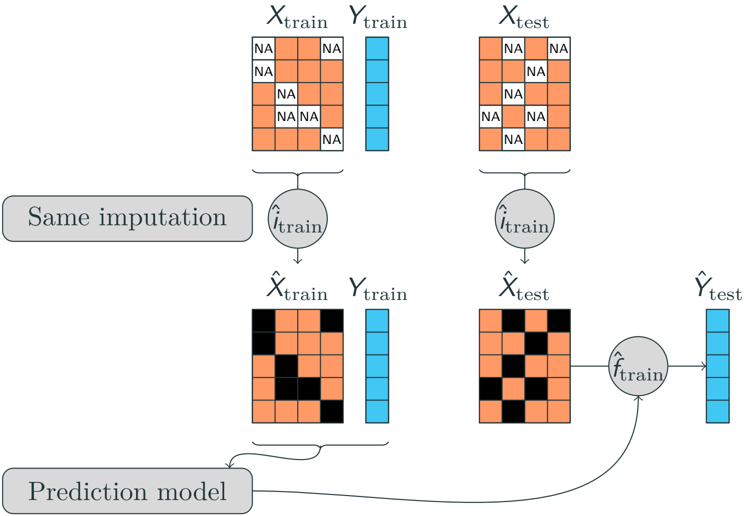


$\hat{p} = \frac{1}{M} \sum_{m=1}^{M} \hat{p}^m$

Imputation prior to learning

Impute the train with $\hat{i}_{train}$ learn a model $\hat{f}_{train}$ with $\hat{X}_{\text{train}}, Y_{\text{train}}$

Impute the test with the same imputation $\hat{i}_{train}$ - predict $\hat{X}_{\text{test}}$ with $\hat{f}_{train}$



Imputation prior to learning

Imputation with the same model

Easy to implement for univariate imputation: The means $(\hat{\mu}_1, ..., \hat{\mu}_d)$ of each colum of the train. Also OK for Gaussian imputation.

Issue: Many methods are "black-boxes" and take as an input the incomplete data and output the completed data (`mice`, `missForest`)

Separate imputation

Impute train and test separately (with a different model)

Issue: Depends on the size of the test set? one observation?

Group imputation/ semi-supervised

Impute train and test simultaneously but the predictive model is learned only on the training imputed data set

Issue: Sometimes no training set at test time

Imputation with the same model: Mean imputation consistent

Learn on the mean-imputed training data, impute the test set with the **same means** and predict is optimal if the missing data are MAR and the **learning algorithm is universally consistent**

Framework - assumptions

- $Y = f(\overline{X}) + \varepsilon$
- $\overline{X} = (X_1, \ldots, X_d)$ has a continuous density $g > 0$ on $[0,1]^d$
- $\|f\|_\infty < \infty$
- Missing data MAR on X_1 with $M_1 \perp\!\!\!\perp X_1 | X_2, \ldots, X_d$.
- $(x_2, \ldots, x_d) \mapsto \mathbb{P}[M_1 = 1 | X_2 = x_2, \ldots, X_d = x_d]$ is continuous
- ε is a centered noise independent of $(\overline{X}, M_1)$

(remains valid when missing values occur for variables $X_1, \ldots, X_j$)

Imputation with the same model: Mean imputation consistent

Learn on the mean-imputed training data, impute the test set with the **same means** and predict is optimal if the missing data are MAR and the **learning algorithm is universally consistent**

Mean imputed entry $\mathbf{x}' = (x_1', x_2, \dots, x_d)$: $x_1' = x_1 \mathbb{1}_{M_1=0} + \mathbb{E}[X_1] \mathbb{1}_{M_1=1}$

Note the data: $\widetilde{\mathbf{X}} = \mathbf{X} \odot (\mathbf{1} - \mathbf{M}) + \mathtt{NA} \odot \mathbf{M}$ (takes value in $\mathbb{R} \cup \{\mathtt{NA}\}$)

Theorem

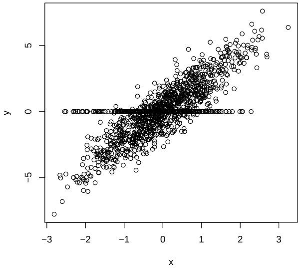
Prediction with mean is equal to the Bayes function almost everywhere

$$f^\star_{impute}(x') = \widetilde{f}^\star(\widetilde{\mathbf{X}}) = \mathbb{E}[Y | \widetilde{\mathbf{X}} = \widetilde{\mathbf{x}}]$$

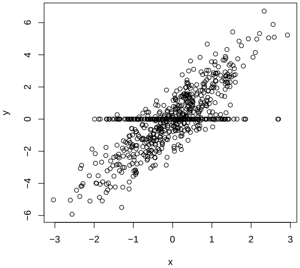
Other values than the mean are OK but use the same value for the train and test sets, otherwise the algorithm may fail as the distributions differ

Consistency of supervised learning with NA: Rationale

- Specific value, systematic like a code for missing
- The learner detects the code and recognizes it at the test time
- With categorical data, just code "Missing"
- With continuous data, any constant:
- Need a lot of data (asymptotic result) and a super powerful learner



Train

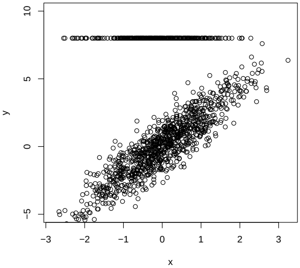


Test

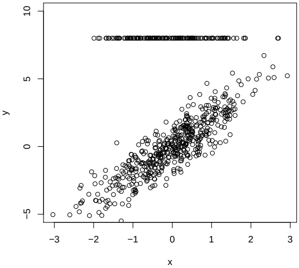
Mean imputation not bad for prediction; it is consistent; despite its drawbacks for estimation - Useful in practice!

Consistency of supervised learning with NA: Rationale

- Specific value, systematic like a code for missing
- The learner detects the code and recognizes it at the test time
- With categorical data, just code "Missing"
- With continuous data, any constant: out of range
- Need a lot of data (asymptotic result) and a super powerful learner



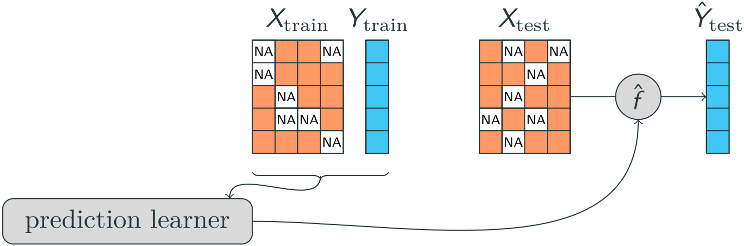
Train



Test

Mean imputation not bad for prediction; it is consistent; despite its drawbacks for estimation - Useful in practice!

End-to-end learning with missing values

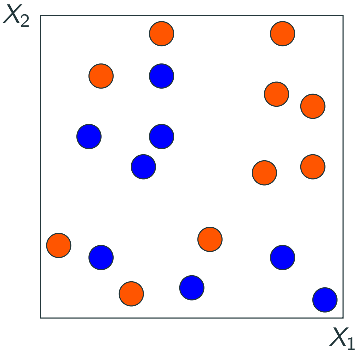


- Trees well suited for empirical risk minimization with missing values: Handle half discrete data $\tilde{\mathbf{X}}$ that takes values in $\mathbb{R} \cup \{\mathtt{NA}\}$
- Random forests powerful learner

CART (Breiman, 1984)

Built recursively by splitting the current cell into two children: Find the feature $j^\star$, the threshold $z^\star$ which minimises the (quadratic) loss

$$(j^\star, z^\star) \in \underset{(j,z)\in\mathcal{S}}{\operatorname{argmin}} \ \mathbb{E}\Big[\big(Y - \mathbb{E}[Y|X_j \leq z]\big)^2 \cdot \mathbb{1}_{X_j \leq z} + \big(Y - \mathbb{E}[Y|X_j > z]\big)^2 \cdot \mathbb{1}_{X_j > z}\Big].$$

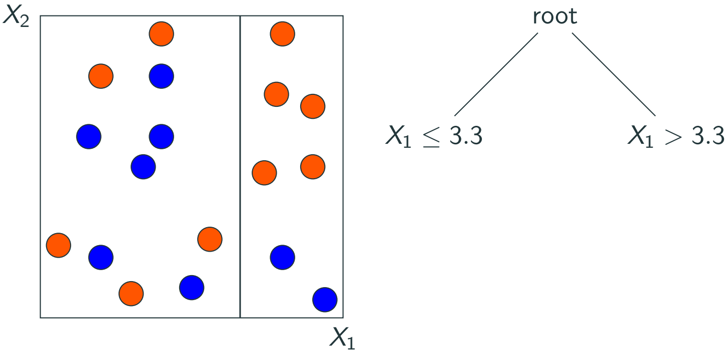


root

CART (Breiman, 1984)

Built recursively by splitting the current cell into two children: Find the feature $j^\star$, the threshold $z^\star$ which minimises the (quadratic) loss

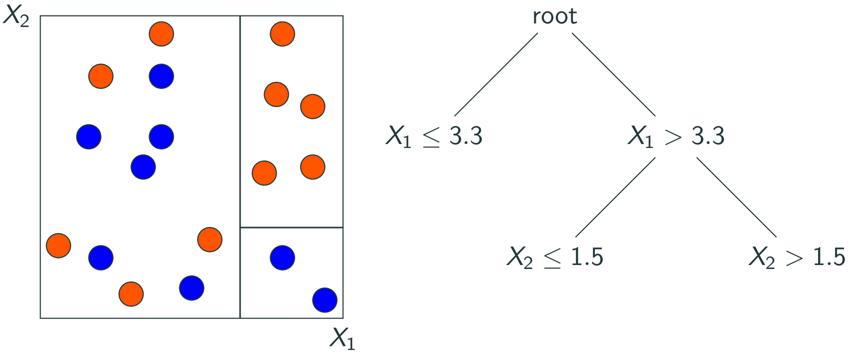
$$(j^\star, z^\star) \in \underset{(j,z)\in\mathcal{S}}{\text{argmin}}\ \mathbb{E}\Big[\big(Y - \mathbb{E}[Y|X_j \leq z]\big)^2 \cdot \mathbb{1}_{X_j \leq z} + \big(Y - \mathbb{E}[Y|X_j > z]\big)^2 \cdot \mathbb{1}_{X_j > z}\Big].$$



CART (Breiman, 1984)

Built recursively by splitting the current cell into two children: Find the feature $j^\star$, the threshold $z^\star$ which minimises the (quadratic) loss

$$(j^\star, z^\star) \in \underset{(j,z)\in\mathcal{S}}{\operatorname{argmin}}\ \mathbb{E}\Big[\big(Y - \mathbb{E}[Y|X_j \leq z]\big)^2 \cdot \mathbb{1}_{X_j \leq z} + \big(Y - \mathbb{E}[Y|X_j > z]\big)^2 \cdot \mathbb{1}_{X_j > z}\Big].$$



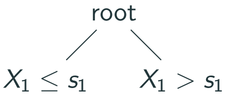
CART with missing values

root

	X_1	X_2	Y
1			
2	NA		
3	NA		
4			

4

CART with missing values



	X_1	X_2	Y
1			
~~2~~	~~NA~~		
~~3~~	~~NA~~		
4			

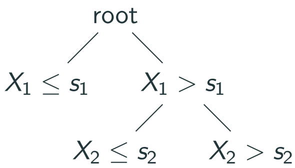
1) Select variable and threshold on observed data [4]

$$\mathbb{E}\left[\left(Y - \mathbb{E}[Y|X_j \leq z, M_j = 0]\right)^2 \cdot \mathbb{1}_{X_j \leq z, M_j = 0} + \left(Y - \mathbb{E}[Y|X_j > z, M_j = 0]\right)^2 \cdot \mathbb{1}_{X_j > z, M_j = 0}\right].$$

[4] Variable selection bias (not a problem to predict): `ctree function, partykit package`, Hothorn, Hornik & Zeileis.

CART with missing values

	X_1	X_2	Y
1			
2	~~NA~~		
3	~~NA~~		
4			



1) Select variable and threshold on observed data [4]

$$\mathbb{E}\Big[\Big(Y - \mathbb{E}[Y|X_j \leq z, M_j = 0]\Big)^2 \cdot \mathbb{1}_{X_j \leq z, M_j = 0} + \Big(Y - \mathbb{E}[Y|X_j > z, M_j = 0]\Big)^2 \cdot \mathbb{1}_{X_j > z, M_j = 0}\Big].$$

2) Propagate observations (2 & 3) with missing values?

- Probabilistic split: $\mathcal{B}ernoulli(\frac{\#L}{\#L+\#R})$ (`Rweeka`)
- Block: Send all to a side by minimizing the error (`xgboost`, `lightgbm`)
- Surrogate split: Search another variable that gives a close partition (rpart)

[4] Variable selection bias (not a problem to predict): `ctree function, partykit package`, Hothorn, Hornik & Zeileis.

Missing incorporated in attribute (Twala *et al.* 2008)

One step: Select the variable, the threshold and propagate missing values

$$f^\star \in \underset{f \in \mathcal{P}_{c,miss}}{\text{argmin}} \ \mathbb{E}\Big[\big(Y - f(\widetilde{\mathbf{X}})\big)^2\Big],$$

where $\mathcal{P}_{c,miss} = \mathcal{P}_{c,miss,L} \cup \mathcal{P}_{c,miss,R} \cup \mathcal{P}_{c,miss,sep}$ with

1. $\mathcal{P}_{c,miss,L} \ \rightarrow \ \{\{\widetilde{X}_j \leq z \vee \widetilde{X}_j = \texttt{NA}\}, \{\widetilde{X}_j > z\}\}$
2. $\mathcal{P}_{c,miss,R} \ \rightarrow \ \{\{\widetilde{X}_j \leq z\}, \{\widetilde{X}_j > z \vee \widetilde{X}_j = \texttt{NA}\}\}$
3. $\mathcal{P}_{c,miss,sep} \ \rightarrow \ \{\{\widetilde{X}_j \neq \texttt{NA}\}, \{\widetilde{X}_j = \texttt{NA}\}\}.$

- Missing values treated like a category (well to handle $\mathbb{R} \cup \texttt{NA}$)
- Good for informative pattern ($\mathbf{M}$ explains Y)
- Implementation: Duplicate the incomplete columns, and replace the missing entries once by $+\infty$ and once by $-\infty$ (J. Tibshirani)
  Implemented for conditional trees and forests partykit package

$\Rightarrow$ Target one model per pattern (2^d):

$$\mathbb{E}\left[Y \middle| \widetilde{\mathbf{X}}\right] = \sum_{\mathbf{m} \in \{0,1\}^d} \mathbb{E}\left[Y | o(\mathbf{X}, \mathbf{m}), \mathbf{M} = \mathbf{m}\right] \mathbb{1}_{\mathbf{M}=\mathbf{m}}$$

Simulations: 20% missing values

Quadratic: $Y = X_1^2 + \varepsilon$, $x_{i.} \in \mathcal{N}(\mu, \Sigma_{4\times 4})$, $\rho = 0.5$, $n = 1000$

$$\widetilde{d}_n = \begin{bmatrix} 2 & 3 & \texttt{NA} & 0 & 15 \\ 1 & \texttt{NA} & 3 & 5 & 13 \\ 9 & 4 & 2 & \texttt{NA} & 18 \\ 7 & 6 & \texttt{NA} & \texttt{NA} & 10 \end{bmatrix}$$

$$\widetilde{d}_n + \text{mask} = \begin{bmatrix} 2 & 3 & \texttt{NA} & 0 & 0 & 0 & 1 & 0 & 15 \\ 1 & \texttt{NA} & 3 & 5 & 0 & 1 & 0 & 0 & 13 \\ 9 & 4 & 2 & \texttt{NA} & 0 & 0 & 0 & 1 & 18 \\ 7 & 6 & \texttt{NA} & \texttt{NA} & 0 & 0 & 1 & 1 & 10 \end{bmatrix}$$

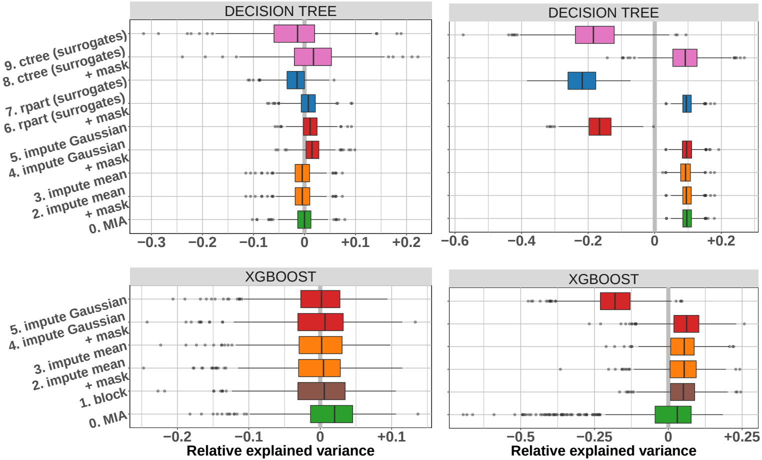
Imputation (mean, Gaussian) + prediction with trees
Imputation (mean, Gaussian) + mask + prediction with trees
Trees MIA

Simulations: 20% missing values

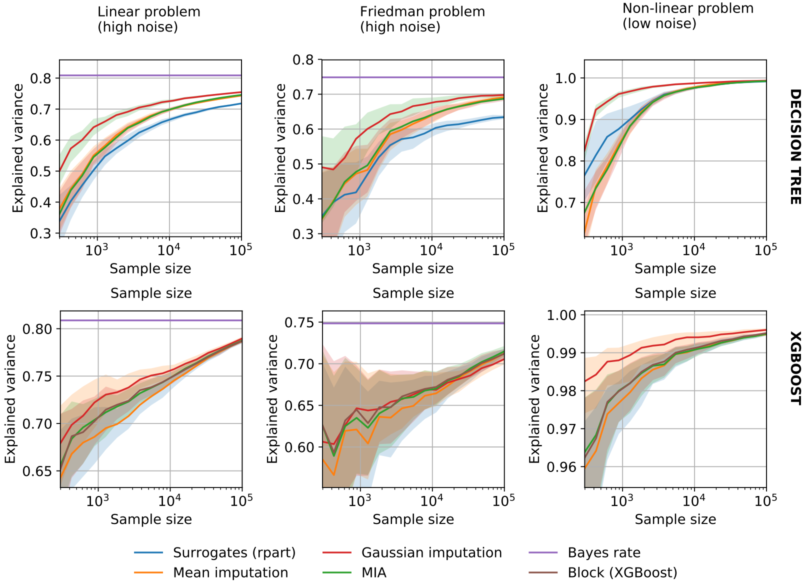
Quadratic: $Y = X_1^2 + \varepsilon,\ x_{i.} \in \mathcal{N}(\mu, \Sigma_{4\times 4}),\ \rho = 0.5,\ n = 1000$

MCAR (MAR) — $M_{i,1} \sim \mathcal{B}(p)$

MNAR - Predictive — $M_{i,1} = \mathbb{1}_{X_{i,1} > [X_1]_{(1-p)n}}$ - $Y = X_1^2 + 3M_1 + \varepsilon$



Consistency: 40% missing values MCAR



Outline

Take home message EM/imputation

- Few implementation of EM strategies

"***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 imputed data have substantial biases." (Dempster & Rubin, 1983)*

- Single imputation aims at completing a dataset as best as possible

- **Multiple imputation** aims at estimating the parameters and their variability taking into account the uncertainty of the missing values

- Single imputation can be appropriate for point estimates

- Both % of NA & structure matter (5% of NA can be an issue)

To conclude

Take home message:

- Principal component methods powerful for single & multiple imputation of quanti & categorical data (rare categories): dimensionality reduction and capture similarities between obs and variables. (be careful some implementations do not handle well categorical data)

  ⇒ Correct inferences for analysis model based on relationships between pairs of variables

  ⇒ SVD can be distributed! Master - Slave, privacy preserving

  ⇒ Requires to choose the number of dimensions S

- Handling missing values in PCA, MCA, FAMD, Multiple Factor Analysis (MFA), Correspondence analysis for contingency tables
- Preprocessing before clustering
- Package R `missMDA` (youtube, website, blog)

Take-home message supervised learning

- Incomplete train and test $\rightarrow$ **same imputation model**
- **Single mean imputation is consistent given a powerful learner**
- Empirically, good imputation methods reduce the number of samples required to reach good prediction

Tree-based models :

- **Missing Incorporated in Attribute** optimizes not only the split but also the handling of the missing values
- Informative missing data: **Adding the mask** helps imputation - MIA

To be done

- Nonasymptotic results
- Uncertainty associated with the prediction
- Distributional shift: No missing values in the test set?
- Prove the usefulness of methods in MNAR

Still an active area of research! Join this exciting field!

Current works

- Variable selection in high dimension Adaptive bayesian SLOPE with missing values. 2019. Jiang, Bogdan, J., Miasojedow, Rockova & TraumaBase
- **MNAR missing values**
  - Contribution of causality for missing data

  - Graphical Models for Processing Missing Data. 2019. Mohan, Pearl.

  - Estimation and imputation in Probabilistic Principal Component Analysis with Missing Not At Random data. 2019. Sportisse, Boyer, J.

  - Contribution of neural nets J., Prost, Scornet, Varoquaux

Other challenges

- MI theory: Good theory for regression parameters but others? Theory with other asymptotic small n, large p ?, imputation model as complex as the analysis one
- Practical imputation issues: Imputation not in agreement (X & X^2), imputation out of range? problems of logical bounds (> 0), MI with large

Resources implementation

Package `missMDA`:
http://factominer.free.fr/missMDA/index.html

Youtube: https://www.youtube.com/watch?v=OOM8_FH6_8o&list=PLnZgp6epRBbQzxFnQrcxg09kRt-PA66T_playlist

Article JSS: https://www.jstatsoft.org/article/view/v070i01

MOOC Exploratory Multivariate Data Analysis

FactoShiny

Resources

R-miss-tastic https://rmisstastic.netlify.com/R-miss-tastic

J., I. Mayer, N. Tierney & N. Vialaneix

Project funded by the R consortium (Infrastructure Steering Committee)[5]

Aim: a reference platform on the theme of missing data management

- list existing packages
- available literature
- tutorials
- analysis workflows on data
- main actors

$\Rightarrow$ Federate the community

$\Rightarrow$ Contribute!

[5] https://www.r-consortium.org/projects/call-for-proposals

Examples:

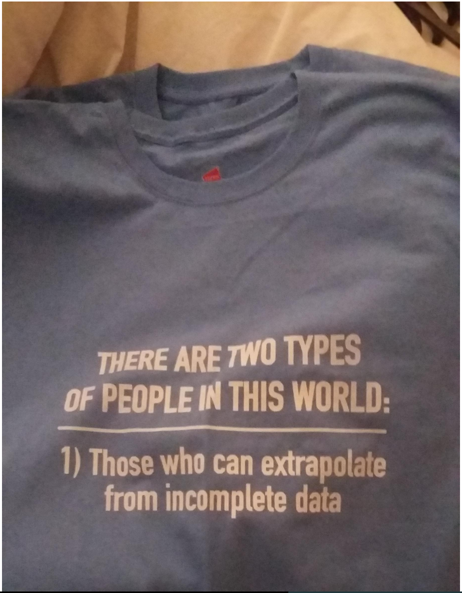
- Lecture [6] - General tutorial : Statistical Methods for Analysis with Missing Data (Mauricio Sadinle)
- Lecture - Multiple Imputation: `mice` by Nicole Erler [7]
- Longitudinal data, Time Series Imputation (Steffen Moritz - very active contributor of r-miss-tastic), Principal Component Methods[8]

[6] https://rmisstastic.netlify.com/lectures/

[7] https://rmisstastic.netlify.com/tutorials/erler_course_multipleimputation_2018/erler_practical_mice_2018

[8] https://rmisstastic.netlify.com/tutorials/Josse_slides_imputation_PCA_2018.pdf

Thank you


THERE ARE TWO TYPES
OF PEOPLE IN THIS WORLD:
1) Those who can extrapolate
from incomplete data