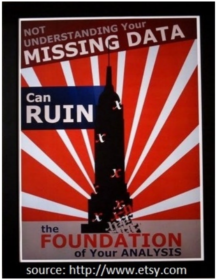
Missing Values Imputation - special focus on principal components methods

Julie Josse & Nick Tierney
Ecole Polytechnique, INRIA & Monash University
useR!2018, Brisbane



July 10, 2018

Presentation J.J

- Dimensionality reduction methods to visualize complex data (PCA based): multi-sources data, textual data, arrays
- Missing values - matrix completion
- Low rank estimation, selection of regularization parameters
- 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:

  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

Presentation N.T

- Methods and tools for exploring and assessing missing data
- Improving analysis workflow
- R Community: rOpenSci member, organiser of rOpenSci Ozunconf, R Consortium group for missing data (with Julie!)
- R packages:
  - naniar: For exploring and analysing missing data
  - visdat: To visualise whole dataframes
  - maxcovr: An interface to the maximal covering location problem
  - mmcc: Rearrange and tidy up MCMC output
  - ukpolice: Pull crime data from the UK police API

Outline

⇒ 9am-10.30am:

- overview of missing values problems
- single imputation with SVD methods

⇒ 11am-12.30am:

- multiple imputation
- categorical, mixed data

⇒ Lecture notes + code available

Overview

1. Missing values
2. Single imputation with PCA
3. Multiple imputation with PCA
4. Categorical data
5. Conclusion

Outline

1. Missing values
2. Single imputation with PCA
3. Multiple imputation with PCA
4. Categorical data
5. Conclusion

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" Gertrude Mary Cox.

⇒ Still an issue in the "big data" area

Data integration: data from different sources

Public Assistance - Paris Hospitals

Traumabase: 15000 patients/ 250 variables

	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

....

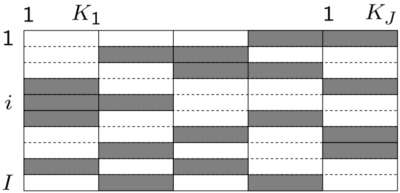
	SpO2	Temperature	Lactates	Hb	Glasgow	Transfusion	....
1	97	35.6	<NA>	12.7	12	yes	
2	100	36.5	4.8	11.1	15	no	
3	100	36	3.9	11.4	3	no	
4	100	36.7	1.66	13	15	yes	
6	100	36	NM	14.4	15	no	
7	100	36.6	NM	14.3	15	yes	
9	100	37.5	13	15.9	15	yes	
10	100	36.9	NM	13.7	15	no	
11	100	36.6	1.2	14.2	14	no	

....

⇒ Predict the Glasgow score, whether to start a blood transfusion, to administer fresh frozen plasma, etc...

⇒ Logistic regressions/Random Forests with missing categorical/continuous values

Multi-blocks data set



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

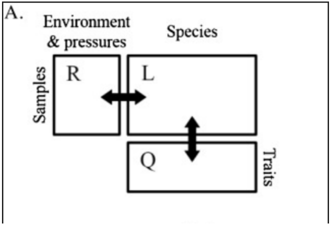
- Self-assessment questionnaire: life style, skin and hair characteristics, care and consumer habits
- Clinical assessments by a dermatologist: facial skin complexion, wrinkles, scalp dryness, greasiness
- Hair assessments by a hair dresser: abundance, volume, breakage, curliness
- Skin and hair photographs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

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 infp: meteorological, geographical (altitude, long)

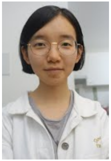


⇒ Aims: Assess the effect of time on species abundances
Monitor the population and assess wetlands conservation policies.

⇒ 70% of missing values in contingency tables

On going works J.J

- François Husson (Agrocampus)
- Genevieve Robin (PhD student), B. Narasimhan (Stanford): distributed matrix completion for multilevel medical data
- Genevieve Robin (PhD student), R. Tibshirani (Stanford): imputation of contingency tables with side information
- Wei Jiang (PhD student): glm with missing values and variable selection
- Erwan Scornet (Polytechnique), N. Prost (PhD student), S. Wager, G. Varoquaux (INRIA): random forest with missing values and causal inference

Ozone data set

	maxO3	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	maxO3v
0601	NA	15.6	18.5	18.4	4	4	8	NA	-1.7101	-0.6946	84
0602	82	17	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

http://www.airbreizh.asso.fr/

Aim: regression with missing values

Missing values problematic

A very simple way: deletion (default `lm` function in R)

Dealing with missing values depends on:

- the pattern of missing values
- the mechanism leading to missing values

Missing values problematic

A very simple way: deletion (default `lm` function in R)
Dealing with missing values depends on:

- the pattern of missing values
- the mechanism leading to missing values

$X = (X_{miss}, X_{obs})$. Let M with $M_{ik} = 1$ if X_{ik} is observed and 0 otherwise. M and X have distributions.

- MCAR: probability does not depend on any values $f(M|X_{obs}, X_{miss}; \phi) = f(M; |phi)$ each entry has the same probability to be observed
- MAR: probability may depend on values on other variables $f(M|X_{obs}, X_{miss}; \phi) = f(M|X_{obs}; \phi)$
- MNAR: probability depends on the value itself $f(M|X_{obs}, X_{miss}; \phi) = f(M|X_{miss}; \phi)$
  $\Rightarrow$ Ex, Age Income.

Missing values problematic

A very simple way: deletion (default `lm` function in R)

Dealing with missing values depends on:

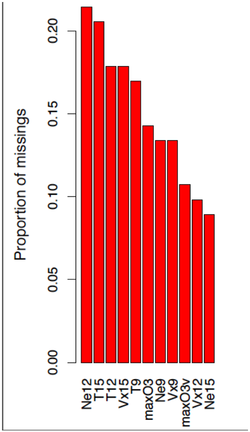
- the pattern of missing values
- the mechanism leading to missing values

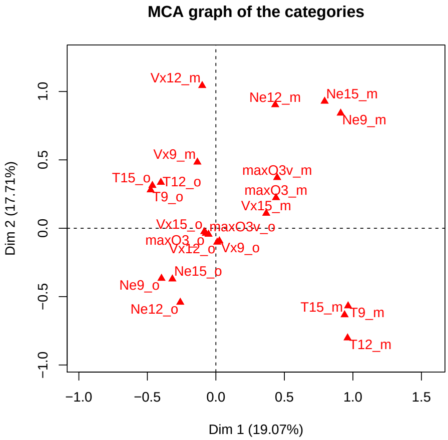
$X = (X_{miss}, X_{obs})$. Let M with $M_{ik} = 1$ if X_{ik} is observed and 0 otherwise. M and X have distributions.

- MCAR: probability does not depend on any values
  $f(M|X_{obs}, X_{miss}; \phi) = f(M; |phi)$ each entry has the same probability to be observed
- MAR: probability may depend on values on other variables
  $f(M|X_{obs}, X_{miss}; \phi) = f(M|X_{obs}; \phi)$
- MNAR: probability depends on the value itself
  $f(M|X_{obs}, X_{miss}; \phi) = f(M|X_{miss}; \phi)$
  $\Rightarrow$ Ex, Age Income.

$\Rightarrow$ Assume MAR: ignore $f(M|X_{obs}, X_{miss}; \phi)$ when doing inference.

Visualization - Multiple Correspondence Analysis





Implemented in VIM, naniar (Matthias Templ, Nick Tierney) - FactoMineR (YouTube): visu pattern, mechanism

Hypothesis: no Missing Not At Random (proba to have missing values depend on the underlying values)

Recommended methods

$\Rightarrow$ Modify the estimation process to deal with missing values.
Maximum likelihood: EM algorithm to obtain point estimates + Supplemented EM (Meng & Rubin, 1991) or Louis for their variability

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

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

Ex: Logistic regression with missing values SAEM algorithm

```
library(devtools)
install_github("wjiang94/misaem")
```

One specific algorithm for each statistical method...

Recommended methods

⇒ Modify the estimation process to deal with missing values.
Maximum likelihood: EM algorithm to obtain point estimates + Supplemented EM (Meng & Rubin, 1991) or Louis for their variability

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

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

Ex: Logistic regression with missing values SAEM algorithm

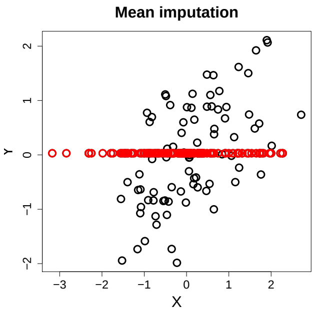
```
library(devtools)
install_github("wjiang94/misaem")
```

One specific algorithm for each statistical method...

⇒ Imputation (multiple) to get a completed data set on which you can perform any statistical method (Rubin, 1976)

Dealing with missing values

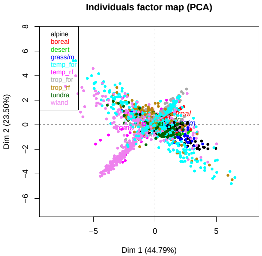
$\Rightarrow$ Imputation to get a completed data set

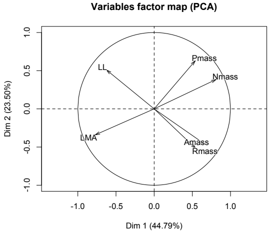


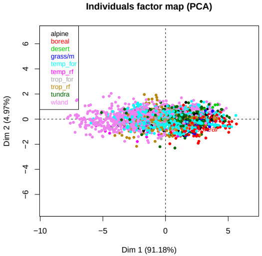
$\mu_y = 0$
$\sigma_y = 1$
$\rho = 0.6$

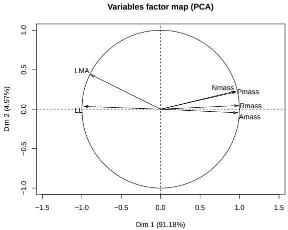
$\hat{\mu}_y = 0.01$
$\hat{\sigma}_y = 0.5$
$\hat{\rho} = 0.30$

Dealing with missing values



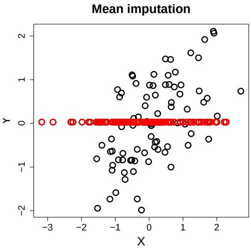






Wright IJ, et al. (2004). The worldwide leaf economics spectrum. *Nature*,
69 000 species - LMA (leaf mass per area), LL (leaf lifespan), Amass (photosynthetic assimilation), Nmass (leaf nitrogen), Pmass (leaf phosphorus), Rmass (dark respiration rate)

Imputation methods

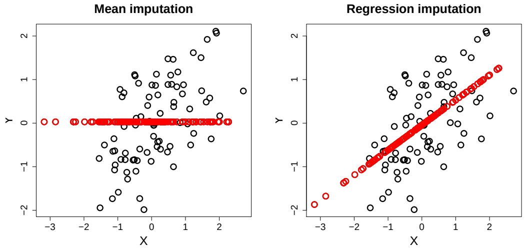


$\mu_y = 0$
$\sigma_y = 1$
$\rho = 0.6$

0.01
0.5
0.30

Imputation methods

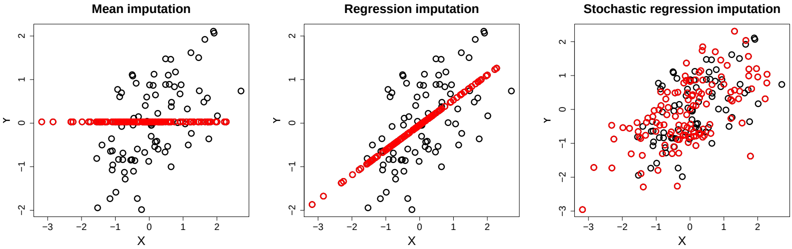
- Impute by regression take into account the relationship: estimate β - impute $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \Rightarrow$ variance underestimated and correlation overestimated.



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

Imputation methods

- Impute by regression take into account the relationship: estimate β - impute $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \Rightarrow$ variance underestimated and correlation overestimated.
- Impute 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 distribution



$\mu_y = 0$	0.01	0.01	0.01
$\sigma_y = 1$	0.5	0.72	0.99
$\rho = 0.6$	0.30	0.78	0.59

Other single imputation methods

- based on Gaussian assumption: $x_{i.} \sim \mathcal{N}(\mu, \Sigma)$
  - Bivariate with missing on $x_{.1}$ (stochastic reg): estimate β and σ - impute from the predictive $x_{i1} \sim \mathcal{N}\left(x_{i2}\hat{\beta}, \hat{\sigma}^2\right)$
  - Extension to multivariate case: estimate μ and Σ from an incomplete data with EM - impute by drawing from $\mathcal{N}\left(\hat{\mu}, \hat{\Sigma}\right)$

  packages `Amelia`, `mice` (conditional)

- k-nearest neighbor (package `VIM`, `yaImpute`, `impute`, etc)
- random forest (package `missForest`)

⇒ Stef van Buuren webpage (mice)

⇒ R miss-tatic N. T. & J.J Task View, Nathalie Villa Vialaneix

⇒ Statistical Science issue (2018) - Imbert & Vialaneix (2018).

Outline

1. Missing values
2. Single imputation with PCA
3. Multiple imputation with PCA
4. Categorical data
5. Conclusion

PCA (complete)

Find the subspace that best represents the data

Figure: 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

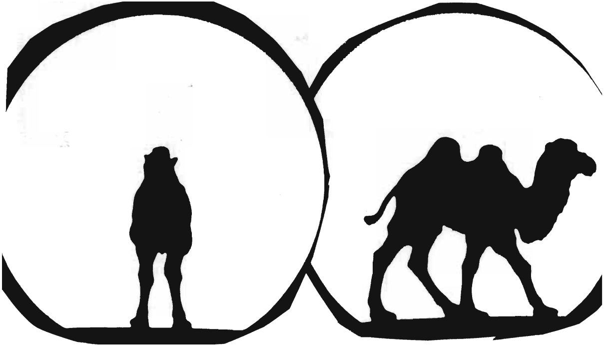


Figure: Camel or dromedary? source J.P. Fénelon

⇒ Best approximation with projection

⇒ Best representation of the variability ⇒ Do not distort the distances between individuals

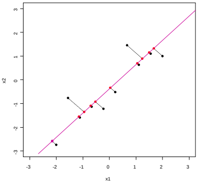
PCA reconstruction

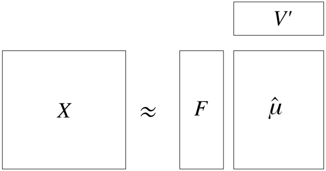
X

-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





$\Rightarrow$ Minimizes distance between observations and their projection

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

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

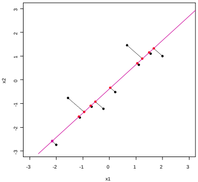
PCA reconstruction

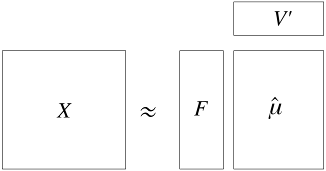
X

-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





$\Rightarrow$ Minimizes distance between observations and their projection

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

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

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

$= F_{n\times S} V'_{p\times S}$ 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

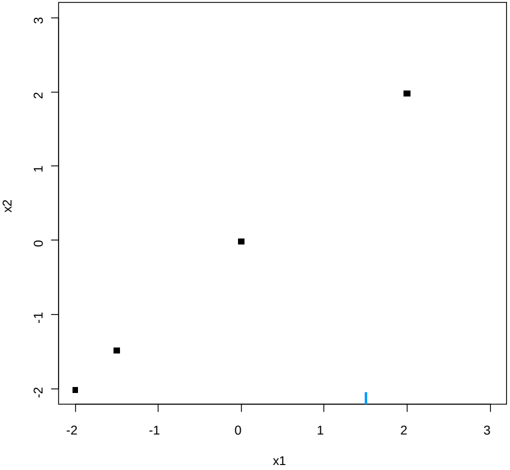
$$\text{argmin}_{\mu} \left\{ \|W_{n\times p} * (X - \mu)\|_2^2 : \text{rank}\,(\mu) \leq S \right\}$$

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

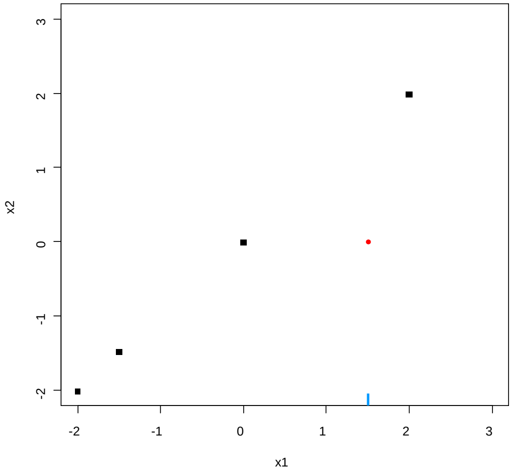
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98



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



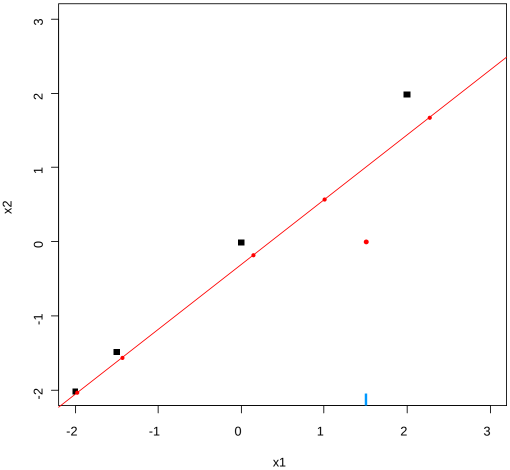
Initialization $\ell = 0$: X^0 (mean imputation)

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



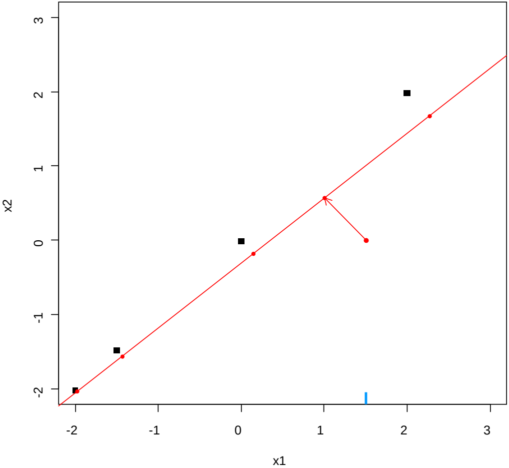
PCA on the completed data set $\rightarrow (U^{\ell}, \Lambda^{\ell}, V^{\ell})$;

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



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

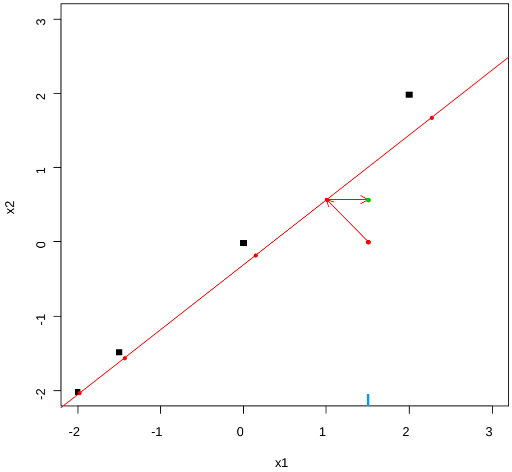
Iterative PCA

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

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



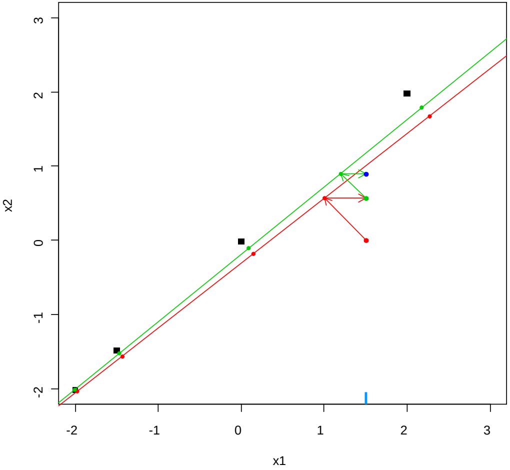
The new imputed dataset is $\hat{X}^{\ell} = W * X + (\mathbf{1} - W) * \hat{\mu}^{\ell}$

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

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



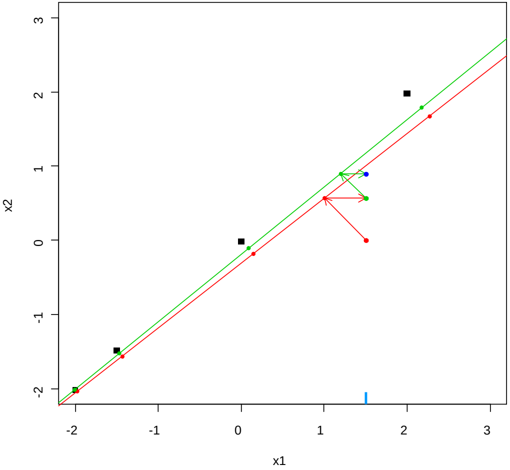
Iterative PCA

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

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

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



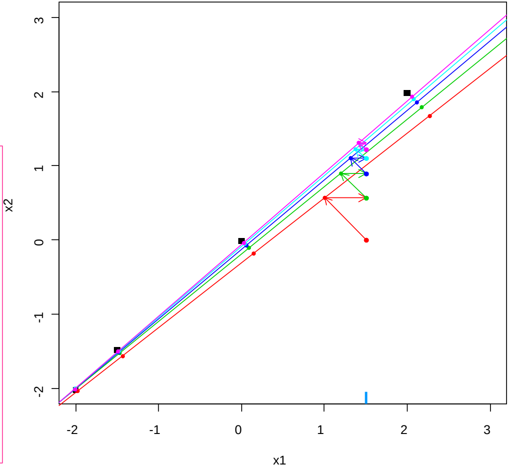
Iterative PCA

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

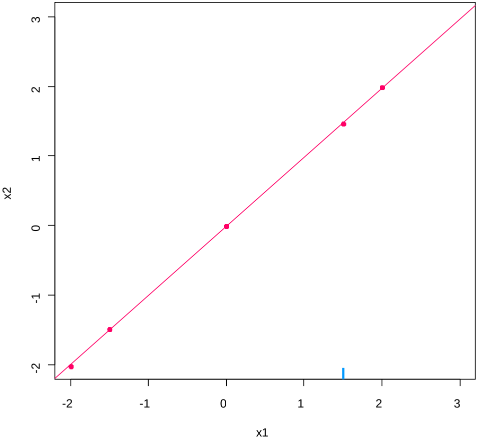
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98



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 $X = \mu + \varepsilon$, $\varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$
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 $X = \mu + \varepsilon$, $\varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$
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 (Josse & Husson, 2012)

Soft thresholding iterative SVD

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

$\Rightarrow$ 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 (Josse *et al.*, 2009)

$\Rightarrow$ 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 $\rightarrow$ regularized PCA $\approx$ PCA

σ^2 large $\rightarrow$ 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

$\Rightarrow$ 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

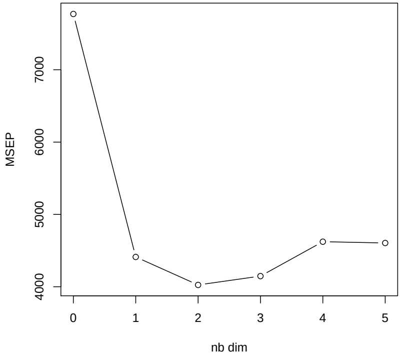
- low noise: iterative PCA (tuning S: cross-validation, GCV)
- moderate noise: iterative regularized PCA (non-linear transformation, tuning σ, S)
- high noise (SNR low, S large): soft thresholding (tuning λ, σ)

  Implemented in R packages `denoiseR` (Josse, Wager, Sardy)

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
```

Incomplete ozone

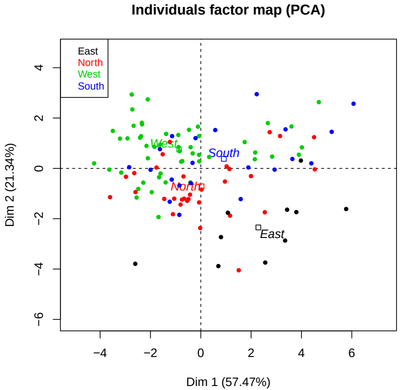
	O3	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	O3v
0601	87	15.6	18.5	18.4	4	4	8	NA	-1.7101	-0.6946	84
0602	82	NA	18.4	17.7	5	5	7	NA	NA	NA	87
0603	92	NA	17.6	19.5	2	5	4	2.9544	1.8794	0.5209	82
0604	114	16.2	NA	NA	1	1	0	NA	NA	NA	92
0605	94	17.4	20.5	NA	8	8	7	-0.5	NA	-4.3301	114
0606	80	17.7	NA	18.3	NA	NA	NA	-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

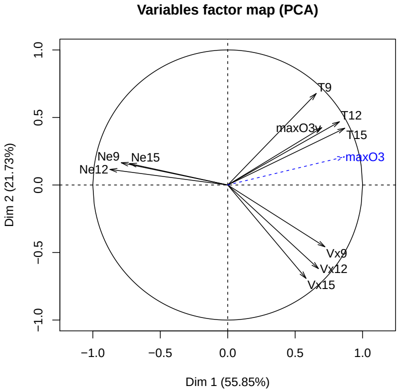
Complete ozone

```
         max03    T9    T12    T15   Ne9  Ne12  Ne15    Vx9   Vx12   Vx15  max03v
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
```

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)
```

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	1.0	1.00	1	1
C2	1	1.0	1.00	1	1
C3	2	2.0	2.00	2	2
C4	2	2.0	2.00	2	2
C5	3	3.0	3.00	3	3
C6	3	3.0	3.00	3	3
C7	4	4.0	4.00	4	4
C8	4	4.0	4.00	4	4
C9	5	5.0	5.00	5	5
C10	5	5.0	5.00	5	5
C11	6	6.0	6.00	6	6
C12	6	6.0	6.00	6	6
C13	7	7.0	7.00	7	7
C14	7	7.0	7.00	7	7
Igor	8	6.87	6.87	8	8
Frank	8	6.87	6.87	8	8
Bertrand	9	6.87	6.87	9	9
Alex	9	6.87	6.87	9	9
Yohann	10	6.87	6.87	10	10
Jean	10	6.87	6.87	10	10

	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	8	8	8	8
Frank	8	8	8	8	8
Bertrand	9	9	9	9	9
Alex	9	9	9	9	9
Yohann	10	10	10	10	10
Jean	10	10	10	10	10

$\Rightarrow$ with Random Forests $\Rightarrow$ with PCA

(Stekhoven, Buhlmann, 2011 - Bartlett, Carpenter, 2014)

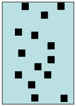
$\Rightarrow$ Non linear relationship well handled by forests

Outline

1. Missing values
2. Single imputation with PCA
3. Multiple imputation with PCA
4. Categorical data
5. Conclusion

Multiple imputation (Rubin, 1987)

Iterative (reg) PCA: single impute with $\hat{\mu}^{\text{shrink}} = \sum_l \psi(\lambda_l)\, U_l V_l^\top$

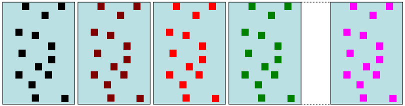


Regression $\hat{\beta}$. Underestimation of std errors: $\widehat{Var}\left(\hat{\beta}\right)$ too small.

$\Rightarrow$ A unique value can't reflect the uncertainty of prediction.

Multiple imputation (Rubin, 1987)

Iterative (reg) PCA: single impute with $\hat{\mu}^{\text{shrink}} = \sum_l \psi(\lambda_l) U_l V_l^\top$



Regression $\hat{\beta}$. Underestimation of std errors: $\widehat{Var}\left(\hat{\beta}\right)$ too small.

$\Rightarrow$ A unique value can't reflect the uncertainty of prediction.

$\Rightarrow$ Multiple imputation: (Ex, one cell: pred 4.88 empirical interval [3.98 ; 5.89])

- B plausibles values for each missing entry.
- Perform the analysis on each imputed data $\hat{\beta}_b, \widehat{Var}\left(\hat{\beta}_b\right)$
- Combine: $\hat{\beta} = \frac{1}{B}\sum_{b=1}^{B}\hat{\beta}_b$ $T = \frac{1}{B}\sum_b \widehat{Var}\left(\hat{\beta}_b\right) + \frac{1}{B-1}\sum_b\left(\hat{\beta}_b - \hat{\beta}\right)^2$

Multiple imputation bivariate case (proper)

Stochastic regression single impute with model $x_{i1} = x_{i2}\beta + \varepsilon_i$
First idea: impute by drawing several times: $b = 1, ..., B$ x_{i1}^b from $\mathcal{N}(x_{i2}\hat{\beta}, (\hat{\sigma}^2))$

Multiple imputation bivariate case (proper)

Stochastic regression single impute with model $x_{i1} = x_{i2}\beta + \varepsilon_i$
First idea: impute by drawing several times: $b = 1, ..., B$ x_{i1}^b from $\mathcal{N}(x_{i2}\hat{\beta}, (\hat{\sigma}^2))$

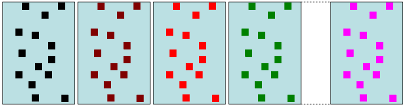
$\Rightarrow$ Variance of prediction: variance of estimation + noise

1) Variability of param: $(\hat{\beta})^1, ..., (\hat{\beta})^B$
$\Rightarrow$ nonparametric bootstrap.
2) Noise: $b = 1, ..., B$ missing x_{i1}^b drawn from $\mathcal{N}(x_{i2}\hat{\beta}^b, (\hat{\sigma}^2)^b)$

PCA multiple imputation

Iterative (reg) PCA: single impute with $\hat{\mu}^{\text{shrink}} = \sum_l \psi(\lambda_l)\, U_l V_l^\top$
Model $X = \mu + \varepsilon \; \varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma^2\right)$.

Noise: $b = 1, ..., B$ missing X_{ij}^b drawn from $\mathcal{N}(\hat{\mu}_{ij}, \hat{\sigma}^2)$



Implemented in missMDA (website François Husson)

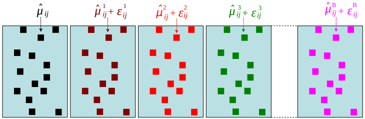
PCA multiple imputation

Iterative (reg) PCA: single impute with $\hat{\mu}^{\text{shrink}} = \sum_l \psi(\lambda_l)\, U_l V_l^\top$
Model $X = \mu + \varepsilon$ $\varepsilon_{ij} \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$.

1) Variability of param: $\hat{\mu}^1 = U^1 {\Lambda^{1/2}}^1 V'^1, .., \hat{\mu}^B = U^B {\Lambda^{1/2}}^B V^{B'}$
$\Rightarrow$ nonparametric bootstrap:

- B samples (rows with replacement) with missing: $X^1, ..., X^B$
- Iterative reg PCA on each bootstrap sample: $\hat{\mu}^1, .., \hat{\mu}^B$

2) Noise: $b = 1, ..., B$ missing X_{ij}^b drawn from $\mathcal{N}((\hat{\mu}_{ij})^b, (\hat{\sigma}^2)^b)$



Implemented in missMDA (website François Husson)

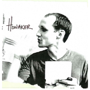
Joint Modeling

$\Rightarrow$ Joint model : $X_{i.} \sim \mathcal{N}(\mu, \Sigma)$

1. Variability of parameters. Bootstrap rows: X^1, ... , X^B
   EM algorithm: $(\hat{\mu}^1, \hat{\Sigma}^1)$, ... , $(\hat{\mu}^B, \hat{\Sigma}^B)$
2. Noise. Imputation: X_{ij}^b drawn from $\mathcal{N}\left(\hat{\mu}^b, \hat{\Sigma}^b\right)$

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

Amelia Earhart



James Honaker

Gary King

Matt Blackwell

Fully conditional modeling

$\Rightarrow$ One model/variable

1. For a variable j

    2.2 Fit a regression model X_j^{obs} on X_{-j} and impute X_j^{miss} with stochastic regression $\mathcal{N}\left(X_{-j}\hat{\beta}_{-j}, \hat{\sigma}^2_{-j}\right)$

2. Cycling through variables

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

$\Rightarrow$ One model/variable

1. For a variable j
   2.1 $(\beta_{-j}, \sigma_{-j})$ drawn from a bootstrap (or a posterior distribution)
   2.2 Fit a regression model X_j^{obs} on X_{-j} and impute X_j^{miss} with stochastic regression $\mathcal{N}\left(X_{-j}\hat{\beta}_{-j}, \hat{\sigma}^2_{-j}\right)$
2. Cycling through variables - Repeat B times

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

$\Rightarrow$ One model/variable

1. For a variable j
   2.1 $(\beta_{-j}, \sigma_{-j})$ drawn from a bootstrap (or a posterior distribution)
   2.2 Fit a regression model X_j^{obs} on X_{-j} and impute X_j^{miss} with stochastic regression $\mathcal{N}\left(X_{-j}\hat{\beta}_{-j}, \hat{\sigma}^2_{-j}\right)$
2. Cycling through variables - Repeat B times

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

Implemented in `mice` (website) and Python

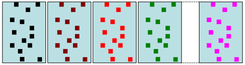
"*There is no clear-cut method for determining whether the MICE algorithm has converged*"

Stef van Buuren

More flexible? Tedious? (incompatible)

Properties

- Imputation model: B multiple imputed sets with PCA, JM, CM



- Analysis model: $\hat{\theta}_b, \widehat{Var}\left(\hat{\theta}_b\right)$ - combine Rubin's rules: $\hat{\theta}$, T

A regression on each imputed data set

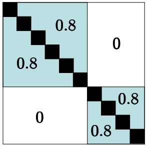
$\Rightarrow$ Aim: inference with missing values

$\Rightarrow$ Good estimates of θ (bias) and coverage ≈ 0.95 (CI width OK):
variability due to missing values is taken into account
reflects well the distribution of the data

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

Simulations

- 1000 simulations
  - data set drawn from $\mathcal{N}_p(\mu, \Sigma)$ with a two-block structure, varying n (30 or 200), p (6 or 60) and ρ (0.3 or 0.9)



  - 10% or 30% of missing values using a MCAR mechanism
  - multiple imputation using $M = 20$ imputed data
- Quantities of interest: $\theta_1 = \mathbb{E}[Y], \theta_2 = \beta_1, \theta_3 = \rho$
- Criteria
  - bias
  - CI width, coverage

Results for the expectation

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

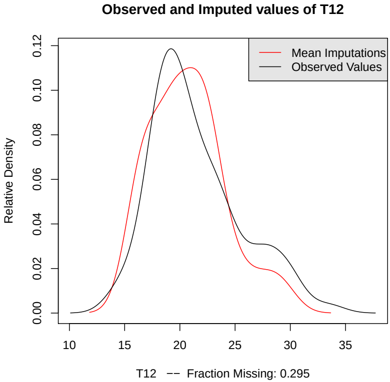
Multiple imputation in practice

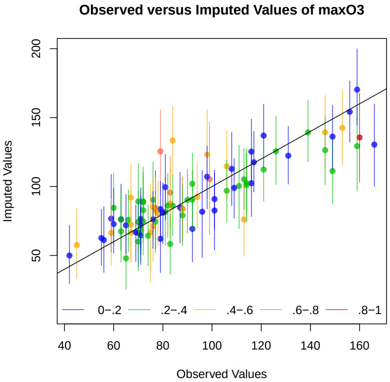
⇒ Step 1: Generate M imputed data sets

```
> 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



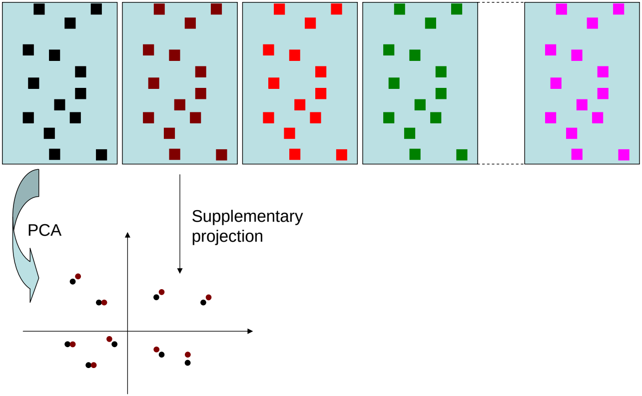



```
# library(Amelia)
> res.amelia <- amelia(don, m = 100)
> compare.density(res.amelia, var = "T12")
> overimpute(res.amelia, var = "maxO3")

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

Multiple imputation in practice

⇒ Step 2: visualization

⇒ Individuals position (and variables) with other predictions

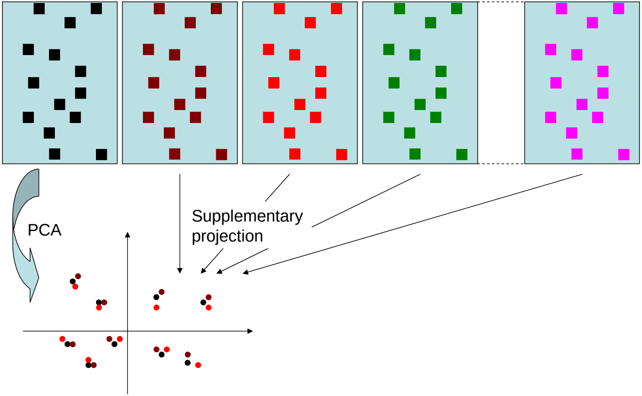


Regularized iterative PCA
⇒ reference configuration

Multiple imputation in practice

⇒ Step 2: visualization

⇒ Individuals position (and variables) with other predictions



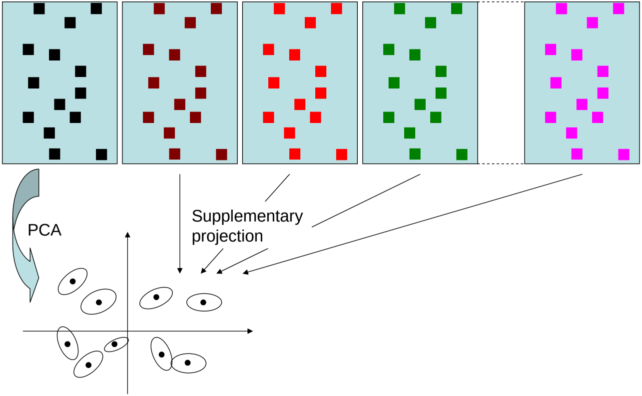
Regularized iterative PCA

⇒ reference configuration

Multiple imputation in practice

⇒ Step 2: visualization

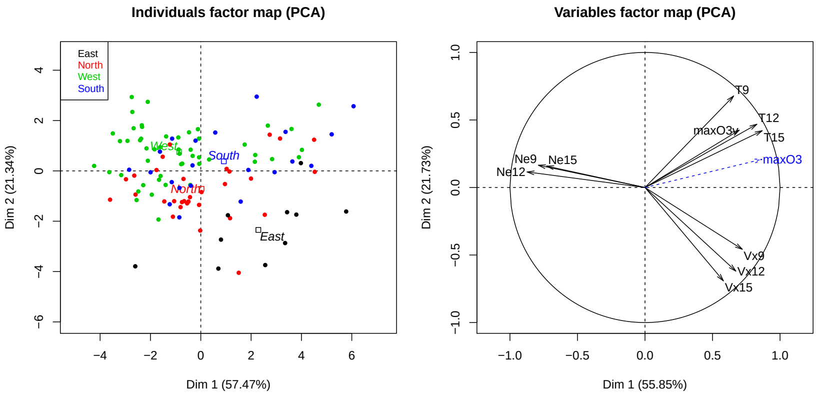
⇒ Individuals position (and variables) with other predictions



Regularized iterative PCA

⇒ reference configuration

PCA representation

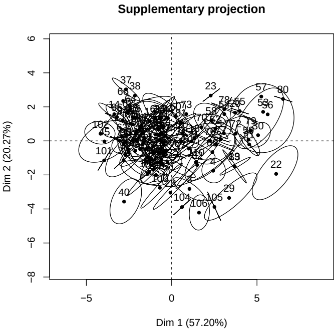


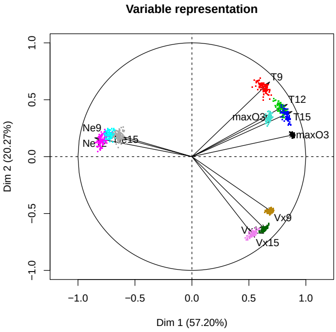

```
> imp <- cbind.data.frame(res.comp$completeObs, ozo[, 12])
> res.pca <- PCA(imp,quanti.sup = 1, quali.sup = 12)
> plot(res.pca, hab =12, lab = "quali"); plot(res.pca, choix = "var")
> res.pca$ind$coord #scores (principal components)
```

Multiple imputation in practice

⇒ Step 2: visualization

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





⇒ Percentage of NA?

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

1. Missing values
2. Single imputation with PCA
3. Multiple imputation with PCA
4. Categorical data
5. Conclusion

Categorical data

Survey data

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

 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

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

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
```

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

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	0.7
			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

Outline

1. Missing values
2. Single imputation with PCA
3. Multiple imputation with PCA
4. Categorical data
5. Conclusion

To conclude

Take home message:

- "***The idea of imputation is both seductive and dangerous****. It is seductive because it can lull the user into the pleasurable state of believing that the data are complete after all, and it is dangerous because it lumps together situations where the problem is sufficiently minor that it can be legitimately handled in this way and situations where standard estimators applied to the real and imputed data have substantial biases.*" (Dempster and Rubin, 1983)
- Single imputation aims to complete a dataset as best as possible (prediction)
- Multiple imputation aims to perform other statistical methods after and to estimate parameters and their variability taking into account the missing values uncertainty
- Single imputation can be appropriate for point estimates

To conclude

Take home message:

- Principal component methods powerful for single & multiple imputation of quanti & categorical data: dimensionality reduction and capture similarities between obs and variables.

  $\Rightarrow$ Correct inferences for analysis model based on relationships between pairs of variables

  $\Rightarrow$ SVD can be distributed! Master - Slave, privacy preserving

  $\Rightarrow$ 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)

Challenges

⇒ MI theory:

- Imputation model as complex as the analysis one (interaction)
- Good theory for regression parameters: others?
- MI theory with new asymptotic small n, large p ?
  ⇒ Still an active area of research
  ⇒ Imputation/Multiple imputation for prediction.
  ⇒ Variable selection

⇒ Some practical issues:

- Imputation not in agreement (X and X^2): missing passive, Imputation out of range?, Problems of logical bounds (> 0)
- Multiple imputation is appealing .... but ... with large data?