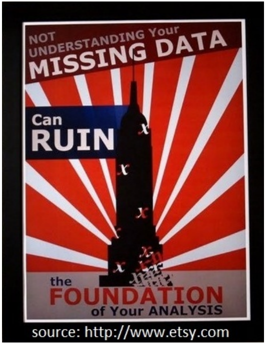
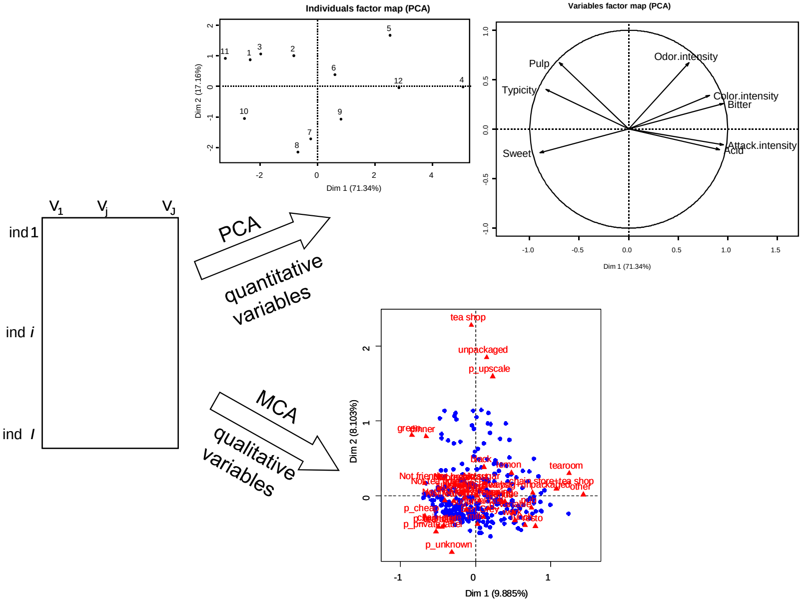
Handling missing values with a special focus on the use of principal components methods

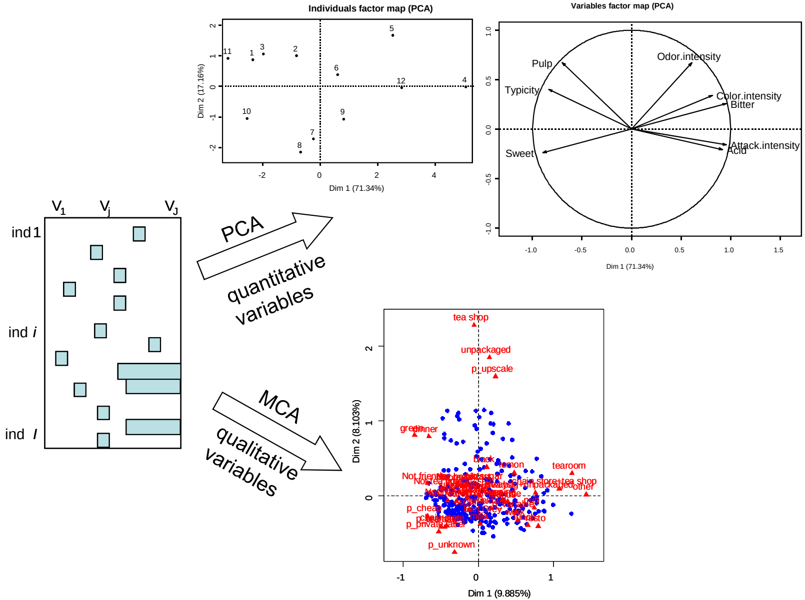
François Husson & Julie Josse
Applied mathematics department, Agrocampus Ouest, Rennes, France



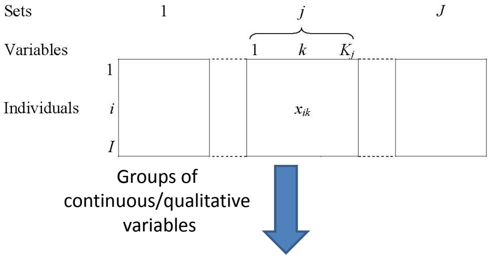
Research activities



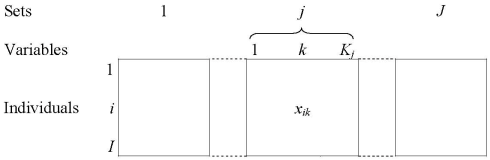
Research activities



Research activities

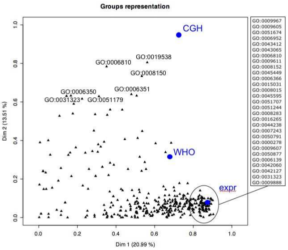


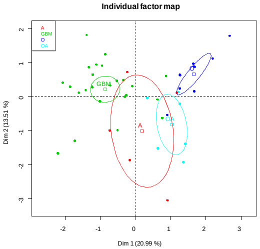
Research activities

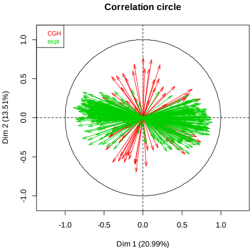


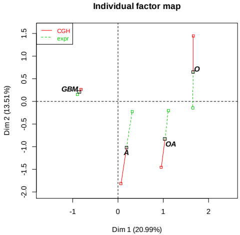
Groups of continuous/qualitative variables

Multiple Factor Analysis









Research activities

- Exploratory multivariate data analysis (principal components methods to visualize data)
- Missing values

- Fields of application: Bio-sciences; sensory analysis

- Books (*Exploratory multivariate analysis with R*, *R for Statistics* and 3 books in French)
- R packages (`FactoMineR - missMDA - SensoMineR`)
- A MOOC on exploratory multivariate data analysis

Outline

Missing values

Gertrude Mary Cox

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

Missing values are ubiquitous:

- no answer in a questionnaire
- data that are lost or destroyed
- machines that fail
- plants damaged
- ...

Missing values

Gertrude Mary Cox

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

Missing values are ubiquitous:

- no answer in a questionnaire
- data that are lost or destroyed
- machines that fail
- plants damaged
- ...

Still an issue in the big data area

A real dataset

	O3	T9	T12	T15	Ne9	Ne12	Ne15	Vx9	Vx12	Vx15	O3v
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

Some references

Schafer (1997),

Joseph L. Schafer

Little & Rubin (1987, 2002)

Roderick Little

Donald Rubin

Suggested reading: chap 25 of Gelman & Hill (2006)

Andrew Gelman

Jennifer L. Hill

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
  - MCAR: probability does not depend on any values
  - MAR: probability may depend on values on other variables
  - MNAR: probability depends on the value itself

(Ex: Income - Age)

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
    - MCAR: probability does not depend on any values
    - MAR: probability may depend on values on other variables
    - MNAR: probability depends on the value itself

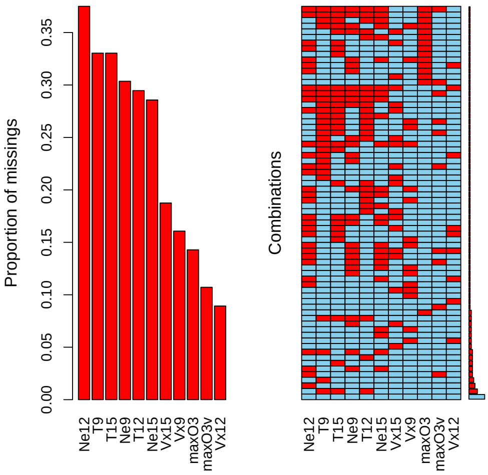
  (Ex: Income - Age)

$\Rightarrow$ Visualization of missing data

Count missing values

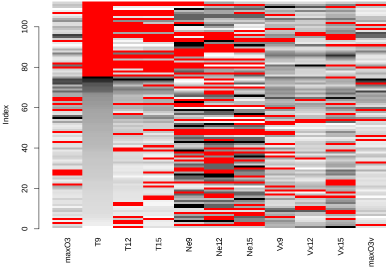
```
> library(VIM)
> res<-summary(aggr(don,prop=TRUE,combined=TRUE))$combinations
> res[rev(order(res[,2])),]

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
                               .....................     .  ...
```

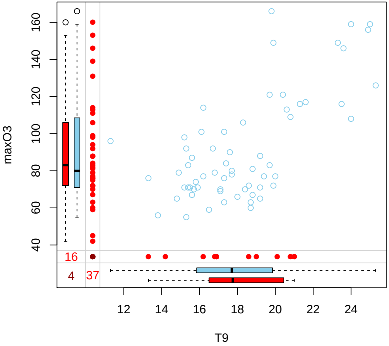
Pattern visualization




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

Visualization





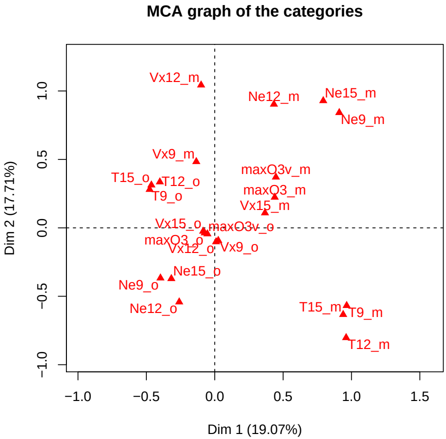

```
> library(VIM)
> matrixplot(don,sortby=2)
> marginplot(don[,c("T9","maxO3")])
```

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")
```

Recommended approaches

$\Rightarrow$ Modify the method, the estimation process to deal with missing values

$\Rightarrow$ Imputation (multiple imputation) to get a completed data set on which you can perform any statistical method

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

Need the modification of the estimation process (not always easy!)

Rationale to get ML estimates on the observed values 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

Hypothesis $\mathbf{x}_{i.} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

$\Rightarrow$ Point estimates with EM:

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

Maximum likelihood approach

Hypothesis $\mathbf{x}_{i.} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

$\Rightarrow$ Point estimates with EM:

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

$\Rightarrow$ Variances:

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

Maximum likelihood approach

Hypothesis $\mathbf{x}_{i.} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

$\Rightarrow$ Point estimates with EM:

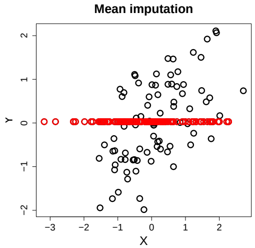
```
> library(norm)
> pre <- prelim.norm(as.matrix(don))
> thetahat <- em.norm(pre)
> getparam.norm(pre,thetahat)
```

$\Rightarrow$ Variances:

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

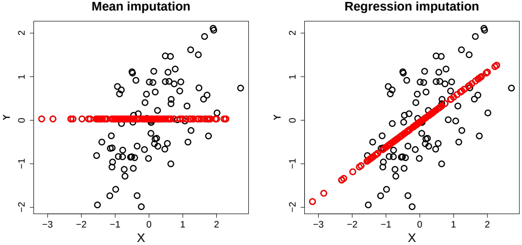
Issue: develop a specific method for each statistical method

Single imputation methods



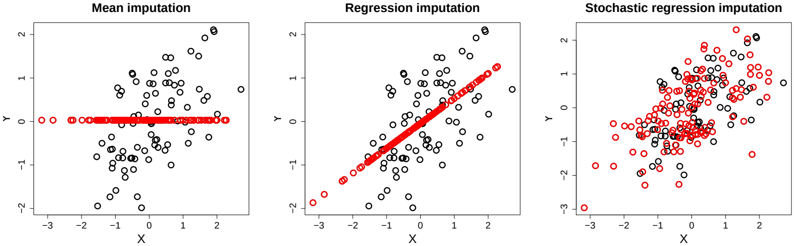
$\mu_y = 0$	0.01
$\sigma_y = 1$	0.5
$\rho = 0.6$	0.30
$CI\mu_y 95\%$	39.4

Single imputation methods



	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
$CI \mu_y 95\%$	39.4	61.6

Single imputation methods

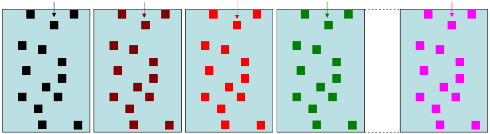


	Mean imputation	Regression imputation	Stochastic regression imputation
$\mu_y = 0$	0.01	0.01	0.01
$\sigma_y = 1$	0.5	0.72	0.99
$\rho = 0.6$	0.30	0.78	0.59
$CI \mu_y 95\%$	39.4	61.6	70.8

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

Multiple imputation (Rubin, 1987)

- Generate M plausible values for each missing value



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

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

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

A multiple imputation procedure requires a single imputation method

1. Single imputation based on normal distribution
2. Single imputation with PCA

3. Multiple imputation based on normal distribution
4. Multiple imputation with Bayesian PCA

Outline

Joint modeling

$\Rightarrow$ Hypothesis $\mathbf{x}_{i.} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

Bivariate case with missing values on Y (stochastic regression):

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

Extension to the multivariate case:

- Estimate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ from an incomplete dataset with EM
- Draw from $\mathcal{N}\left(\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}\right)$

```
> pre <- prelim.norm(as.matrix(don))
> thetahat <- em.norm(pre)
> rngseed(123)
> imp <- imp.norm(pre,thetahat,don)
```

Conditional modeling

$\Rightarrow$ A model per variable

Example with regression:

1. Initial imputation: mean imputation
2. Fit a stochastic regression $\mathbf{X}_j^{obs}$ on the other variables $\mathbf{X}_{-j}^{obs}$
   Predict $\mathbf{X}_j^{miss}$ using the trained regression on $\mathbf{X}_{-j}^{miss}$
3. Cycling through variables

```
> library(mice)
> res.cm <- mice(don, m=1)
```

Conditional modeling

⇒ A model per variable

Example with regression:

1. Initial imputation: mean imputation
2. Fit a stochastic regression $\mathbf{X}_j^{obs}$ on the other variables $\mathbf{X}_{-j}^{obs}$
   Predict $\mathbf{X}_j^{miss}$ using the trained regression on $\mathbf{X}_{-j}^{miss}$
3. Cycling through variables

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

⇒ Flexibility: different models for each variable

```
> library(mice)
> res.cm <- mice(don, m=1)
```

Other single imputation methods

- k-nearest neighbor (`class`, `FNN`)
- random forest (`missForest`, Stekhoven & Bühlmann, 2011)
- ...

⇒ van Buuren: http://www.stefvanbuuren.nl/mi/Software.html
⇒ R task View: Official Statistics & Survey Methodology

⇒ Imputation based on PCA became famous with the Netflix challenge!

PCA (complete)

Find the subspace that best represents the data

Figure: What's this?

⇒ Best approximation with projection
⇒ Best representation of the variability

PCA (complete)

Find the subspace that best represents the data

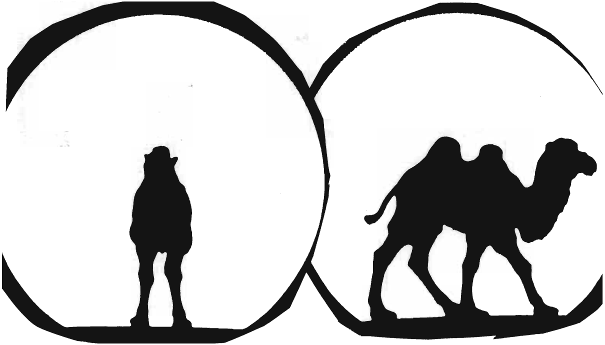


Figure: Camel or dromedary? source J.P. Fénelon

⇒ Best approximation with projection

⇒ Best representation of the variability

PCA

$\Rightarrow$ Geometrical point of view: minimize the reconstruction error

Approximation of $\mathbf{X}$ of low rank ($S < p$):

$\|\mathbf{X}_{n\times p} - \hat{\mathbf{X}}_{n\times p}\|^2$ SVD: $\hat{\mathbf{X}}^{\text{PCA}} = \mathbf{U}_{n\times S}\mathbf{\Lambda}^{\frac{1}{2}}_{S\times S}\mathbf{V}'_{p\times S} = \mathbf{F}_{n\times S}\mathbf{V}'_{p\times S}$

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

PCA

$\Rightarrow$ Geometrical point of view: minimize the reconstruction error

Approximation of $\mathbf{X}$ of low rank ($S < p$):

$\|\mathbf{X}_{n\times p} - \hat{\mathbf{X}}_{n\times p}\|^2$ SVD: $\hat{\mathbf{X}}^{\text{PCA}} = \mathbf{U}_{n\times S}\mathbf{\Lambda}^{\frac{1}{2}}_{S\times S}\mathbf{V}'_{p\times S} = \mathbf{F}_{n\times S}\mathbf{V}'_{p\times S}$

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

$\Rightarrow$ Model point of view: fixed effect model (Caussinus, 1986)

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

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

Maximum likelihood estimates: least squares estimates

Imputation with PCA

⇒ PCA: least squares

$$\|\mathbf{X}_{n\times p} - \mathbf{U}_{n\times S}\mathbf{\Lambda}^{\frac{1}{2}}_{S\times S}\mathbf{V}'_{p\times S}\|^2$$

⇒ PCA with missing values: weighted least squares

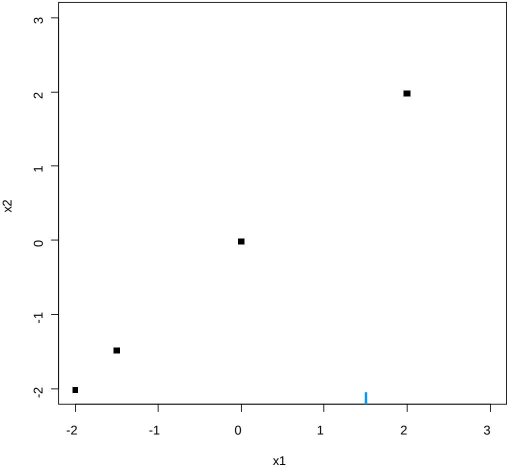
$$\|\mathbf{W}_{n\times p} * (\mathbf{X}_{n\times p} - \mathbf{U}_{n\times S}\mathbf{\Lambda}^{\frac{1}{2}}_{S\times S}\mathbf{V}'_{p\times S})\|^2$$

with $w_{ij} = 0$ if x_{ij} is missing, $w_{ij} = 1$ otherwise

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

Iterative PCA

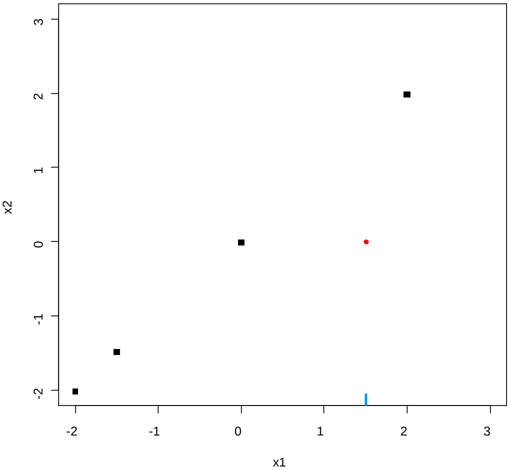
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	NA
2.0	1.98



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



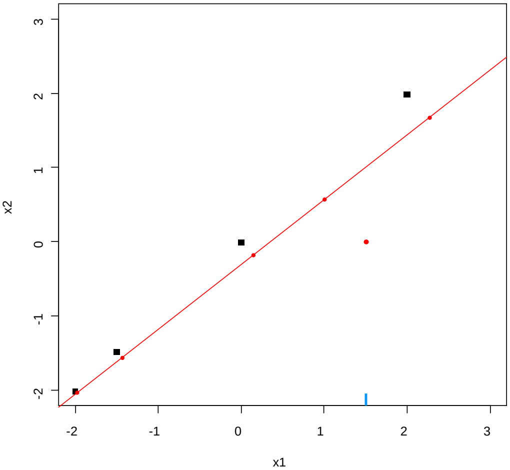
Initialization $\ell = 0$: $\mathbf{X}^0$ (mean imputation)

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



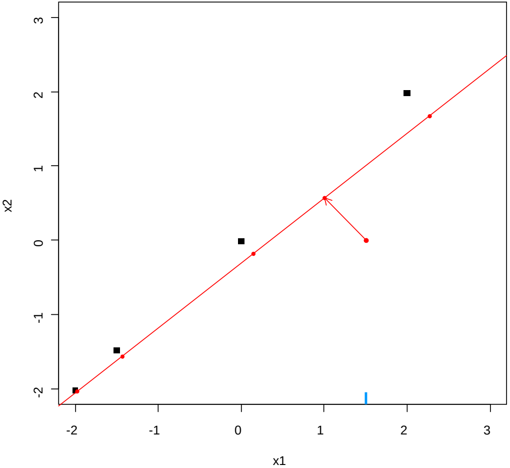
PCA on the completed data set $\rightarrow (\mathbf{U}^\ell, \mathbf{\Lambda}^\ell, \mathbf{V}^\ell)$;

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

$\hat{x1}$	$\hat{x2}$
-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 model matrix $\hat{\mathbf{X}}^{\ell} = \mathbf{U}^{\ell}\mathbf{\Lambda}^{1/2^{\ell}}\mathbf{V}^{\ell\prime}$

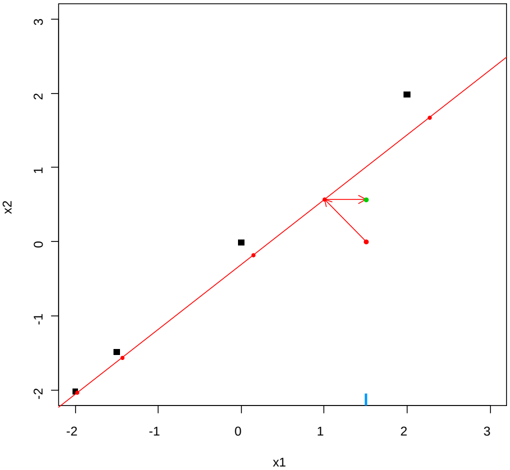
Iterative PCA

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



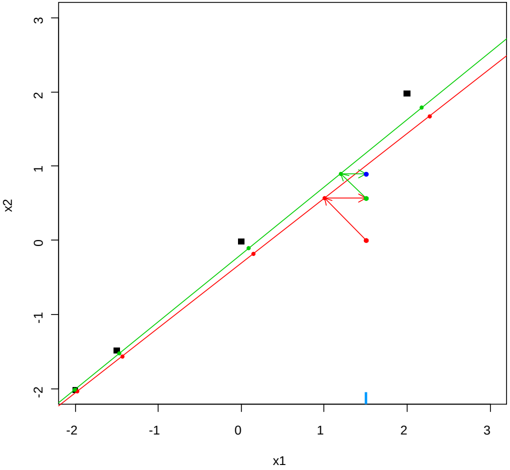
The new imputed dataset is $\mathbf{X}^{\ell} = \mathbf{W} * \mathbf{X} + (1 - \mathbf{W}) * \hat{\mathbf{X}}^{\ell}$

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



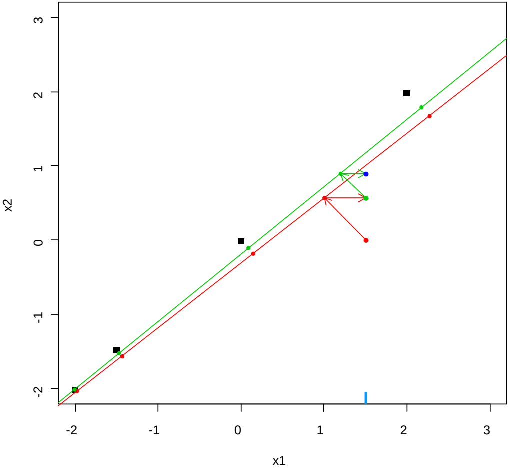
Iterative PCA

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



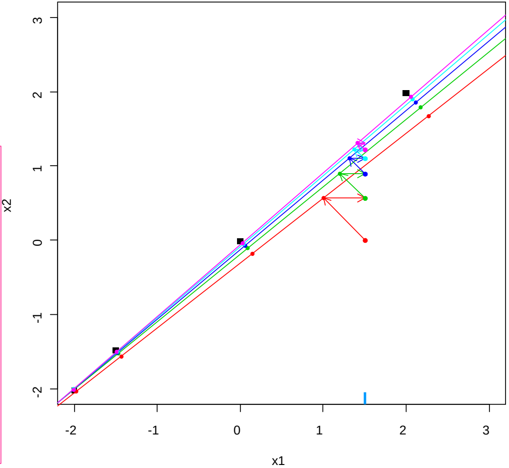
Iterative PCA

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

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

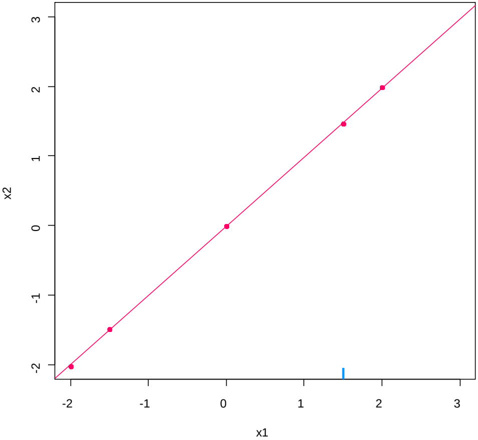
x1	x2
-2.0	-2.01
-1.5	-1.48
0.0	-0.01
1.5	**0.57**
2.0	1.98



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 $\rightarrow (\mathbf{U}^\ell, \mathbf{\Lambda}^\ell, \mathbf{V}^\ell)$

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

Iterative PCA

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

Iterative PCA

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

Iterative PCA

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

Iterative PCA

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

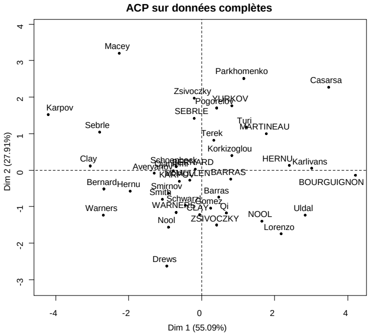
$\Rightarrow$ EM algorithm of the fixed effect model

$\Rightarrow$ Imputation (matrix completion framework, Netflix)

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

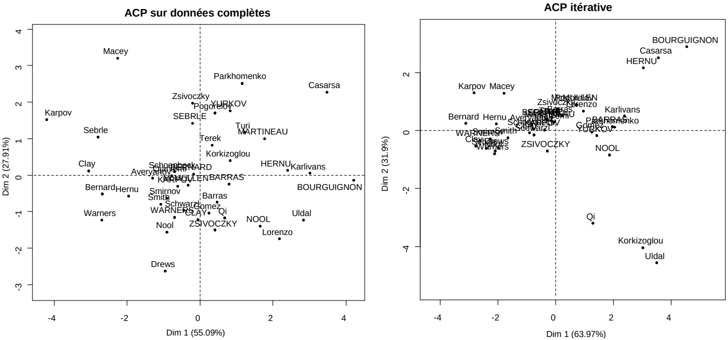
Overfitting

$$\mathbf{X}_{41\times 6} = \mathbf{F}_{41\times 2}\mathbf{V}'_{2\times 6} + \mathcal{N}(0, 0.5)$$



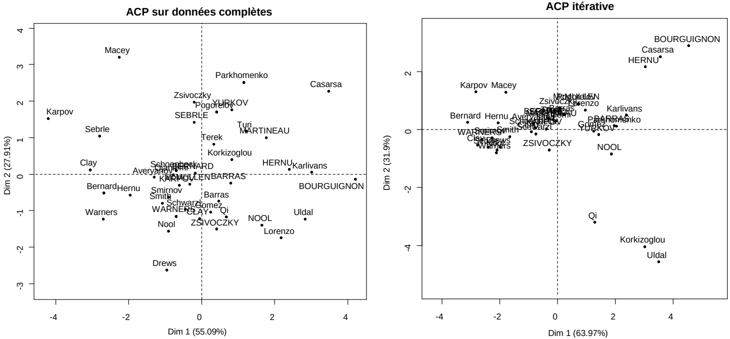
Overfitting

$$\mathbf{X}_{41\times 6} = \mathbf{F}_{41\times 2}\mathbf{V}'_{2\times 6} + \mathcal{N}(0, 0.5) \qquad \Rightarrow 50\% \text{ of NA}$$



Overfitting

$$\mathbf{X}_{41\times 6} = \mathbf{F}_{41\times 2}\mathbf{V}'_{2\times 6} + \mathcal{N}(0, 0.5) \qquad \Rightarrow 50\% \text{ of NA}$$



$\Rightarrow$ fitting error is low: $||\mathbf{W} * (\mathbf{X} - \hat{\mathbf{X}})||^2 = 0.48$

$\Rightarrow$ prediction error is high: $||(1 - \mathbf{W}) * (\mathbf{X} - \hat{\mathbf{X}})||^2 = 5.58$

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

⇒ Trust too much the relationship between variables

Remarks:

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

Solution:
⇒ Shrinkage methods

Regularized iterative PCA (Josse *et al.*, 2009)

$\Rightarrow$ Initialization - estimation step - imputation step

The imputation step:

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

is replaced by a "shrunk" imputation step:

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

Regularized iterative PCA (Josse *et al.*, 2009)

⇒ Initialization - estimation step - imputation step

The imputation step:

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

is replaced by a "shrunk" imputation step:

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

$$\hat{\sigma}^2 = \frac{RSS}{\text{ddl}} = \frac{n \sum_{s=S+1}^{q} \lambda_s}{np - p - nS - pS + S^2 + S} \qquad (\mathbf{X}_{n \times p}; \mathbf{U}_{n \times S}; \mathbf{V}_{p \times S})$$

Regularized iterative PCA (Josse *et al.*, 2009)

$\Rightarrow$ Initialization - estimation step - imputation step

The imputation step:

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

is replaced by a "shrunk" imputation step:

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

$$\hat{\sigma}^2 = \frac{RSS}{\text{ddl}} = \frac{n \sum_{s=S+1}^{q} \lambda_s}{np - p - nS - pS + S^2 + S} \qquad (\mathbf{X}_{n \times p}; \mathbf{U}_{n \times S}; \mathbf{V}_{p \times S})$$

Between hard/soft thresholding (Mazumder, Hastie & Tibshirani, 2010)

σ^2 small $\rightarrow$ regularized PCA $\approx$ PCA

σ^2 large $\rightarrow$ mean imputation

Properties of the imputation

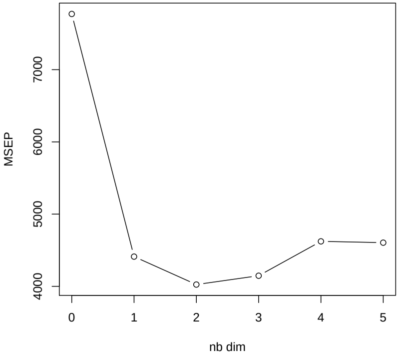
- Good imputation quality when the structure is strong (imputation using similarities between individuals and relationship between variables)

- Competitive with random forests

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")
```



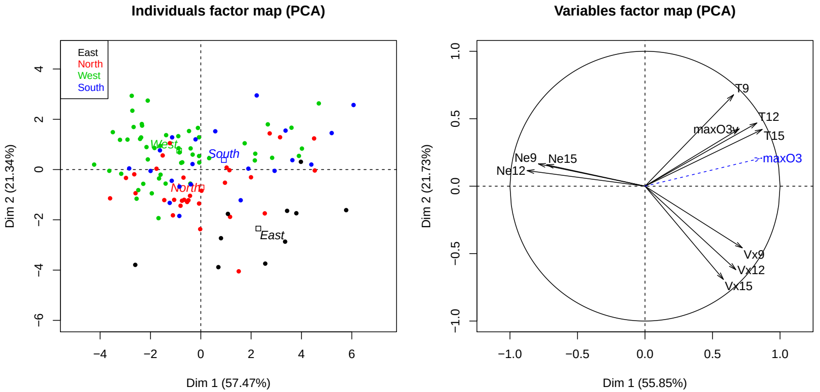
Imputation with PCA in practice

⇒ 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!

⇒ visualization of the incomplete data: a crucial step




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

An ecological data set

Glopnet data: 2494 species described by 6 quantitative variables

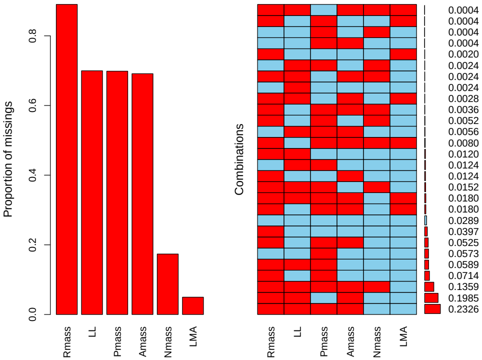
- LMA (leaf mass per area)
- LL (leaf lifespan)
- Amass (photosynthetic assimilation)
- Nmass (leaf nitrogen),
- Pmass (leaf phosphorus)
- Rmass (dark respiration rate)

and 1 categorical variable: the biome

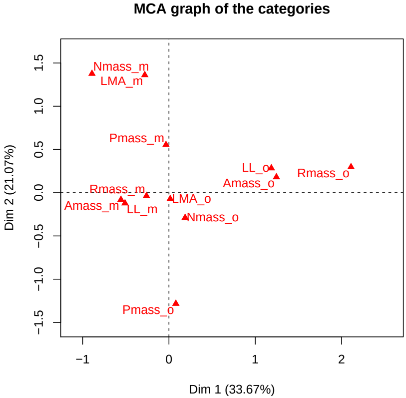
Wright IJ, et al. (2004). The worldwide leaf economics spectrum. *Nature*, 428:821.
www.nature.com/nature/journal/v428/n6985/extref/nature02403-s2.xls

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")
```

Percentage of inertia if the variables are independent

	Number of variables												
nbind	4	5	6	7	8	9	10	11	12	13	14	15	16
5	96.5	93.1	90.2	87.6	85.5	83.4	81.9	80.7	79.4	78.1	77.4	76.6	75.5
6	93.3	88.6	84.8	81.5	79.1	76.9	75.1	73.2	72.2	70.8	69.8	68.7	68.0
7	90.5	84.9	80.9	77.4	74.4	72.0	70.1	68.3	67.0	65.3	64.3	63.2	62.2
8	88.1	82.3	77.2	73.8	70.7	68.2	66.1	64.0	62.8	61.2	60.0	59.0	58.0
9	86.1	79.5	74.8	70.7	67.4	65.1	62.9	61.1	59.4	57.9	56.5	55.4	54.3
10	84.5	77.5	72.3	68.2	65.0	62.4	60.1	58.3	56.5	55.1	53.7	52.5	51.5
11	82.8	75.7	70.3	66.3	62.9	60.1	58.0	56.0	54.4	52.7	51.3	50.1	49.2
12	81.5	74.0	68.6	64.4	61.2	58.3	55.8	54.0	52.4	50.9	49.3	48.2	47.2
13	80.0	72.5	67.2	62.9	59.4	56.7	54.4	52.2	50.5	48.9	47.7	46.6	45.4
14	79.0	71.5	65.7	61.5	58.1	55.1	52.8	50.8	49.0	47.5	46.2	45.0	44.0
15	78.1	70.3	64.6	60.3	57.0	53.9	51.5	49.4	47.8	46.1	44.9	43.6	42.5
16	77.3	69.4	63.5	59.2	55.6	52.9	50.3	48.3	46.6	45.2	43.6	42.4	41.4
17	76.5	68.4	62.6	58.2	54.7	51.8	49.3	47.1	45.5	44.0	42.6	41.4	40.3
18	75.5	67.6	61.8	57.1	53.7	50.8	48.4	46.3	44.6	43.0	41.6	40.4	39.3
19	75.1	67.0	60.9	56.5	52.8	49.9	47.4	45.5	43.7	42.1	40.7	39.6	38.4
20	74.1	66.1	60.1	55.6	52.1	49.1	46.6	44.7	42.9	41.3	39.8	38.7	37.5
25	72.0	63.3	57.1	52.5	48.9	46.0	43.4	41.4	39.6	38.1	36.7	35.5	34.5
30	69.8	61.1	55.1	50.3	46.7	43.6	41.1	39.1	37.3	35.7	34.4	33.2	32.1
35	68.5	59.6	53.3	48.6	44.9	41.9	39.5	37.4	35.6	34.0	32.7	31.6	30.4
40	67.5	58.3	52.0	47.3	43.4	40.5	38.0	36.0	34.1	32.7	31.3	30.1	29.1
45	66.4	57.1	50.8	46.1	42.4	39.3	36.9	34.8	33.1	31.5	30.2	29.0	27.9
50	65.6	56.3	49.9	45.2	41.4	38.4	35.9	33.9	32.1	30.5	29.2	28.1	27.0
100	60.9	51.4	44.9	40.0	36.3	33.3	31.0	28.9	27.2	25.8	24.5	23.3	22.3
2500			35.6										

Table: 95th percentile of the percentage of inertia explained by the first component of 10,000 MCAs performed on tables made up of independent variables with 2 categories.

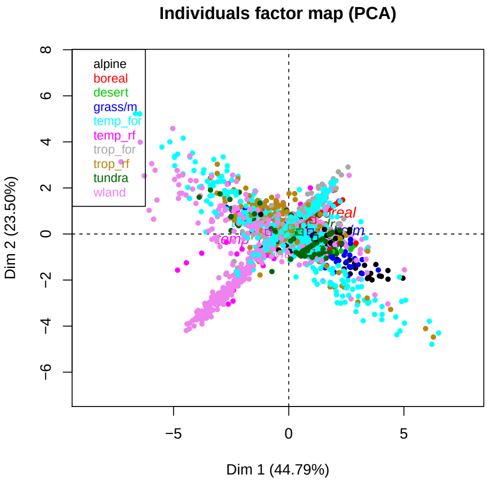
Percentage of inertia if the variables are independent

	Number of variables												
nbind	17	18	19	20	25	30	35	40	50	75	100	150	200
5	74.9	74.2	73.5	72.8	70.7	68.8	67.4	66.4	64.7	62.0	60.5	58.5	57.4
6	67.0	66.3	65.6	64.9	62.3	60.4	58.9	57.6	55.8	52.9	51.0	49.0	47.8
7	61.3	60.7	59.7	59.1	56.4	54.3	52.6	51.4	49.5	46.4	44.6	42.4	41.2
8	57.0	56.2	55.4	54.5	51.8	49.7	47.8	46.7	44.6	41.6	39.8	37.6	36.4
9	53.6	52.5	51.8	51.2	48.1	45.9	44.4	42.9	41.0	38.0	36.1	34.0	32.7
10	50.6	49.8	49.0	48.3	45.2	42.9	41.4	40.1	38.0	35.0	33.2	31.0	29.8
11	48.1	47.2	46.5	45.8	42.8	40.6	39.0	37.7	35.6	32.6	30.8	28.7	27.5
12	46.2	45.2	44.4	43.8	40.7	38.5	36.9	35.5	33.5	30.5	28.8	26.7	25.5
13	44.4	43.4	42.8	41.9	39.0	36.8	35.1	33.9	31.8	28.8	27.1	25.0	23.9
14	42.9	42.0	41.3	40.4	37.4	35.2	33.6	32.3	30.4	27.4	25.7	23.6	22.4
15	41.6	40.7	39.8	39.1	36.2	34.0	32.4	31.1	29.0	26.0	24.3	22.4	21.2
16	40.4	39.5	38.7	37.9	35.0	32.8	31.1	29.8	27.9	24.9	23.2	21.2	20.1
17	39.4	38.5	37.6	36.9	33.8	31.7	30.1	28.8	26.8	23.9	22.2	20.3	19.2
18	38.3	37.4	36.7	35.8	32.9	30.7	29.1	27.8	25.9	22.9	21.3	19.4	18.3
19	37.4	36.5	35.8	34.9	32.0	29.9	28.3	27.0	25.1	22.2	20.5	18.6	17.5
20	36.7	35.8	34.9	34.2	31.3	29.1	27.5	26.2	24.3	21.4	19.8	18.0	16.9
25	33.5	32.5	31.8	31.1	28.1	26.0	24.5	23.3	21.4	18.6	17.0	15.2	14.2
30	31.2	30.3	29.5	28.8	26.0	23.9	22.3	21.1	19.3	16.6	15.1	13.4	12.5
35	29.5	28.6	27.9	27.1	24.3	22.2	20.7	19.6	17.8	15.2	13.7	12.1	11.1
40	28.1	27.3	26.5	25.8	23.0	21.0	19.5	18.4	16.6	14.1	12.7	11.1	10.2
45	27.0	26.1	25.4	24.7	21.9	20.0	18.5	17.4	15.7	13.2	11.8	10.3	9.4
50	26.1	25.3	24.6	23.8	21.1	19.1	17.7	16.6	14.9	12.5	11.1	9.6	8.7
100	21.5	20.7	19.9	19.3	16.7	14.9	13.6	12.5	11.0	8.9	7.7	6.4	5.7

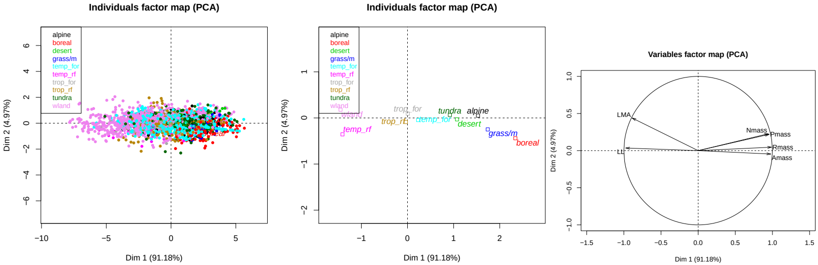
Table: 95th percentile of the percentage of inertia explained by the first component of 10,000 MCAs performed on tables made up of independent variables with 2 categories.

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)
```

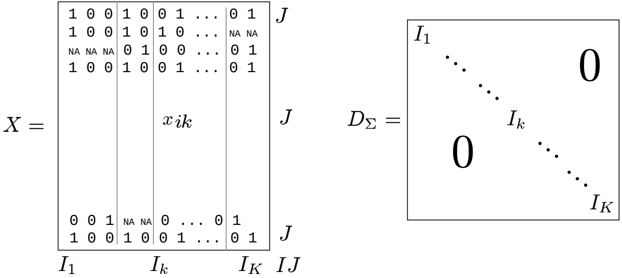
Outline

1. Introduction
2. Single imputation for continuous variables
3. Single imputation for categorical variables
4. Single imputation for mixed variables
5. Multiple imputation

Single imputation based on MCA for categorical data

Survey data

PCA on an indicator matrix **X** with specific weights $\mathbf{D}_\Sigma$



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

- Initialization: imputation of the indicator matrix (proportion)
- Iterate until convergence
  1. Estimation of $\mathbf{F}^{\ell}, \mathbf{V}^{\ell}$: MCA on the completed indicator matrix
  2. Imputation of the missing values with the model matrix
  3. Column margins are updated

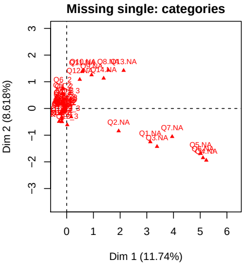
	V1	V2	V3	...	V14
ind 1	a	**NA**	g	...	u
ind 2	**NA**	f	g		u
ind 3	a	e	h		v
ind 4	a	e	h		v
ind 5	b	f	h		u
ind 6	c	f	h		u
ind 7	c	f	**NA**		v
...	...	...	...		...
ind 1232	c	f	h		v

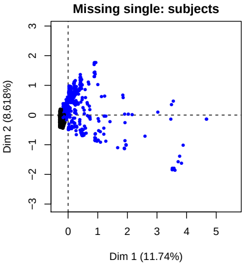
	V1_a	V1_b	V1_c	V2_e	V2_f	V3_g	V3_h	...
ind 1	1	0	0	**0.71**	**0.29**	1	0	...
ind 2	**0.12**	**0.29**	**0.59**	0	1	1	0	...
ind 3	1	0	0	1	0	0	1	...
ind 4	1	0	0	1	0	0	1	...
ind 5	0	1	0	0	1	0	1	...
ind 6	0	0	1	0	1	0	1	...
ind 7	0	0	1	0	1	**0.37**	**0.63**	...
...	...	...	...	...	...	...	...	...
ind 1232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

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

A real example

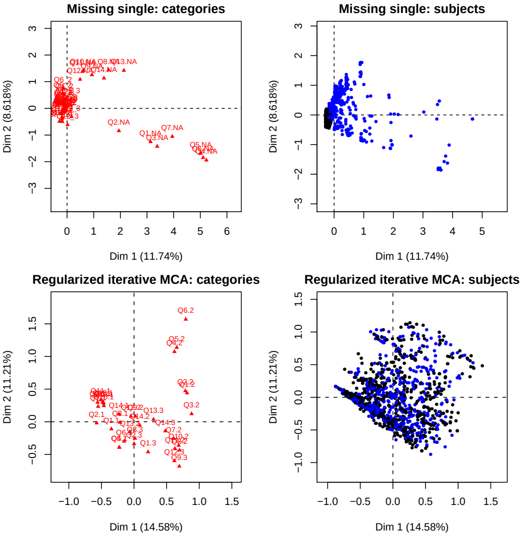
- 1232 respondents, 14 questions, 35 categories, 9% of missing values concerning 42% of respondents





A real example

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



Outline

Mixed variables

⇒ Joint modeling:

- General location model (Schafer, 1997) $\Longrightarrow$ pb when many categories
- Transform the categorical variables into dummy variables and deal as continuous variables (`Amelia`)
- Latent class models (Vermunt) – nonparametric Bayesian models (work in progress, Dunson, Reiter, Duke University)

⇒ Conditional modeling: linear, logistic, multinomial logit models (`mice`)

Mixed variables

⇒ Joint modeling:

- General location model (Schafer, 1997) $\Longrightarrow$ pb when many categories
- Transform the categorical variables into dummy variables and deal as continuous variables (`Amelia`)
- Latent class models (Vermunt) – nonparametric Bayesian models (work in progress, Dunson, Reiter, Duke University)

⇒ Conditional modeling: linear, logistic, multinomial logit models (`mice`)

⇒ Random forests (Stekhoven & Bühlmann, 2012, `missForest`)

⇒ Principal components method (Audigier, Husson & Josse, 2014, `missMDA`)

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 $\mathbf{X}_j^{obs}$ on variables $\mathbf{X}_{-j}^{obs}$ and then predict $\mathbf{X}_j^{miss}$
3. Cycling through variables until a stopping criterion is met

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 $\mathbf{X}_j^{obs}$ on variables $\mathbf{X}_{-j}^{obs}$ and then predict $\mathbf{X}_j^{miss}$
3. Cycling through variables until a stopping criterion is met

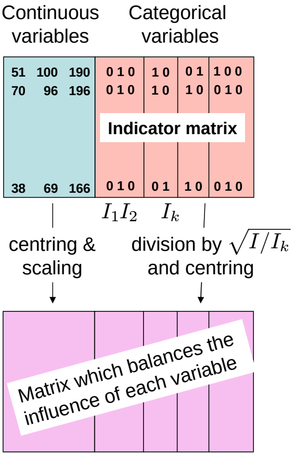
⇒ Properties:

- Non-linear relations, complex interactions
- $n << p$
- out-of-bag error rates: approximation of the imputation error

⇒ Outperforms k-nn and `mice`

Principal component method for mixed data (complete)

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



A PCA is performed on the weighted matrix

Properties of the method

- The distance between individuals is:

$$d^2(i,l) = \sum_{k=1}^{K_{cont}} (x_{ik} - x_{lk})^2 + \sum_{q=1}^{Q} \sum_{k=1}^{K_q} \frac{1}{I_{k_q}} (x_{iq} - x_{lq})^2$$

- The principal component $\mathbf{F}_s$ maximises:

$$\sum_{k=1}^{K_{cont}} r^2(\mathbf{F}_s, v_k) + \sum_{q=1}^{Q_{cat}} \eta^2(\mathbf{F}_s, v_q)$$

Iterative FAMD algorithm

1. Initialization: imputation mean (continuous) and proportion (dummy)
2. Iterate until convergence
   (a) estimation: FAMD on the completed data $\Rightarrow \mathbf{U}, \mathbf{\Lambda}, \mathbf{V}$
   (b) imputation of the missing values with the model matrix
   (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

Iterative FAMD

$\Rightarrow$ Properties:

- Imputation based on scores and loadings $\Rightarrow$ similarities between individuals and relationships between continuous and categorical variables
- Linear relationships
- Compared to a PCA on the (unweighted) indicator matrix, small categories are better imputed
- The number of dimensions is a tuning parameter
- Good performances compared to the method based on random forests, especially for categorical variables

Simulations

- Simulation pattern
  - 2 independent variables are drawn from a normal distribution
  - 1 variable is replicated 4 times, the other 8 $\Rightarrow$ 2 dimensions
  - Random noise is added
  - Half of the variables in each dimension are split in 3 clusters
  - 10%, 20% or 30% of missing values are chosen at random

$\Rightarrow$ Data are constructed (expected) to be in 4 dimensions

- Criterion
  - for continuous data:

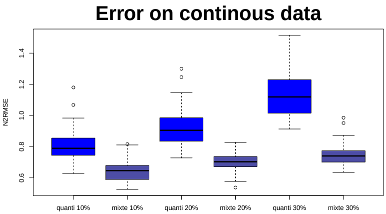
$$N2RMSE = \sqrt{\sum_{i \in \text{missing}} \frac{mean\left(\left(X_i^{true} - X_i^{imp}\right)^2\right)}{var\left(X_i^{true}\right)}}$$

  - for categorical data: proportion of falsely classified entries

Simulations

Imputation using continuous data only

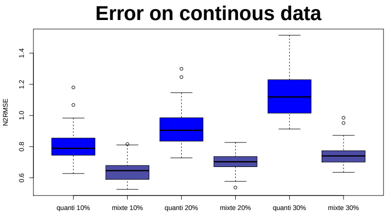
Imputation using both continuous and categorical data



Simulations

Imputation using continuous data only

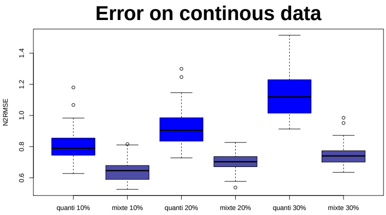
Imputation using both continuous and categorical data

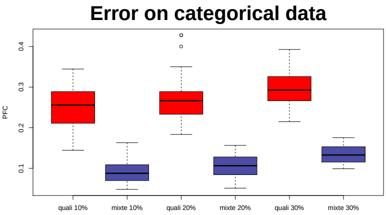


Categorical data improved the imputation on continuous data ...

Simulations

Imputation using continuous data only Imputation using categorical data only
Imputation using both continuous and categorical data

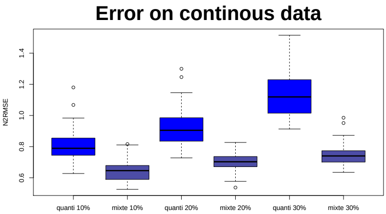


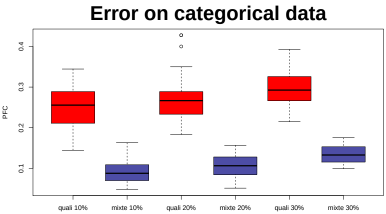


Categorical data improved the imputation on continuous data ...

Simulations

Imputation using continuous data only Imputation using categorical data only
Imputation using both continuous and categorical data

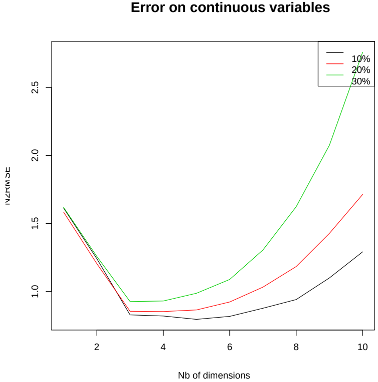


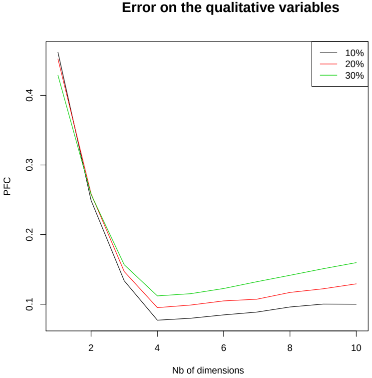


Categorical data improved the imputation on continuous data ...

... and continuous data improved the imputation on categorical data

Simulations

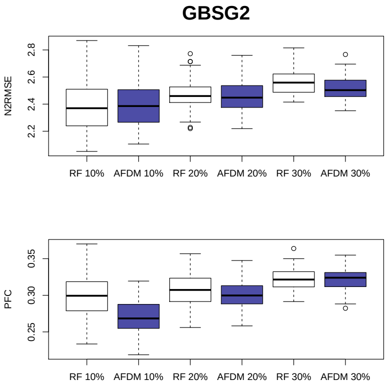




⇒ The error on the estimation of the number of dimensions has not an important impact on the imputation error ... if the estimation is not too bad

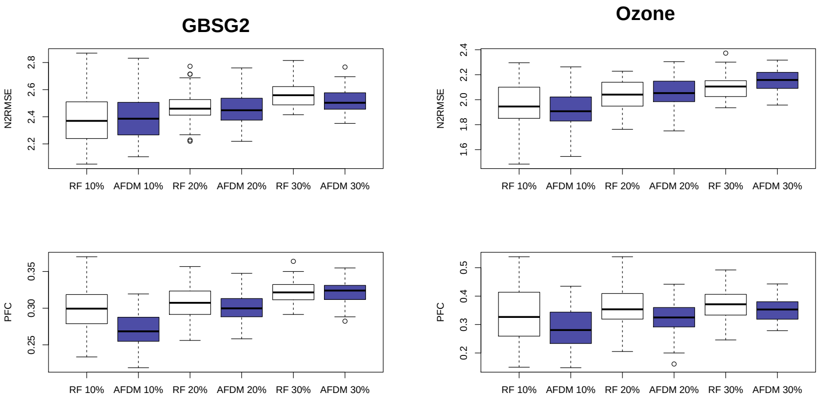
Comparison with random forest on real data sets

Imputations obtained with random forest & iterative algorithm



Comparison with random forest on real data sets

Imputations obtained with random forest & iterative algorithm



Comparison with random forest

Compared to random forest, imputations are quite similar

Imputations are slightly better:

- for categorical variables
- especially for rare categories

and imputations are worse:

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

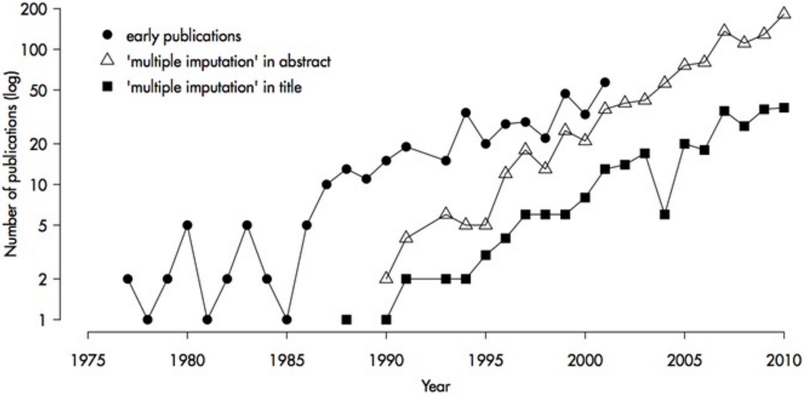
Mixed imputation in practice

```
> library(missMDA)
> imputeFAMD(mydata,ncp=2)

> library(missForest)
> missForest(mydata)

> library(mice)
> mice(mydata)
> mice(mydata, defaultMethod = "rf") ## mice with random forests
```

Outline

Muliple Imputation uses

Number of publications (log) on multiple imputation during the period 1977-2010

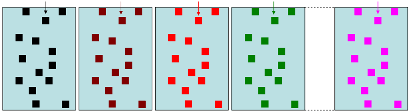


Source: S. Van Buuren webpage

Multiple imputation

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

1. Generating M imputed data sets



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

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

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 $m = 1, \dots, M$, 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

Multiple imputation: 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: Bayesian regression

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

Joint modeling

$\Rightarrow$ Hypothesis $\mathbf{x}_{i.} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$

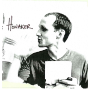
Algorithm:

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

Easy to parallelized

Implemented in `Amelia` (website)

Amelia Earhart



James Honaker

Gary King

Matt Blackwell

Conditional modeling

$\Rightarrow$ Hypothesis: one model/variable

Algorithm:

1. Initial imputation: mean imputation
2. For a variable j
   2.1 $(\beta^{-j}, \sigma^{-j})$ drawn from a Bootstrap or a posterior distribution
   2.2 Imputation: stochastic regression x_{ij} drawn from $\mathcal{N}\left(\mathbf{X}_{-j}\beta^{-j}, \sigma^{-j}\right)$
3. Cycling through variables
4. Repeat M times steps 2 and 3

Implemented in `mice` (website)

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

Stef van Buuren

Joint / Conditional modeling

$\Rightarrow$ 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!
- Appears to work quite well in practice

$\Rightarrow$ Drawbacks: one model/variable... tedious...

Joint / Conditional modeling

$\Rightarrow$ 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!
- Appears to work quite well in practice

$\Rightarrow$ Drawbacks: one model/variable... tedious...

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

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

Multiple imputation with PCA and Bootstrap

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

1. Variability of the parameters, M plausible: $(\hat{x}_{ij})^1, ..., (\hat{x}_{ij})^M$
   Bootstrap residuals: $\mathbf{X}^1 = \hat{\mathbf{X}} + \varepsilon^1, ..., \mathbf{X}^M = \hat{\mathbf{X}} + \varepsilon^M$
   Iterative PCA: $\hat{\mathbf{X}}^1 = \mathbf{U}^1 \mathbf{\Lambda}^1 \mathbf{V}^1$, ..., $\hat{\mathbf{X}}^M = \mathbf{U}^M \mathbf{\Lambda}^M \mathbf{V}^M$
2. Noise: for $m = 1, ..., M$, missing values x_{ij}^m are imputed by drawing from the predictive distribution $\mathcal{N}(\hat{x}_{ij}^m, \hat{\sigma}^2)$

Implemented in missMDA (website)

François Husson

Julie Josse

Joint, conditional and PCA

$\Rightarrow$ 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 high 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)

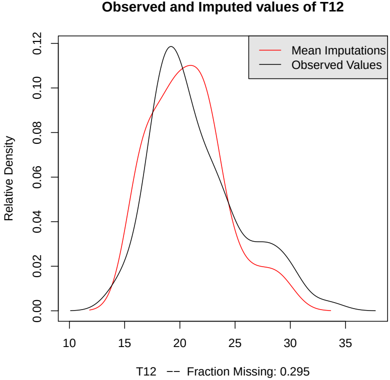
Multiple imputation in practice

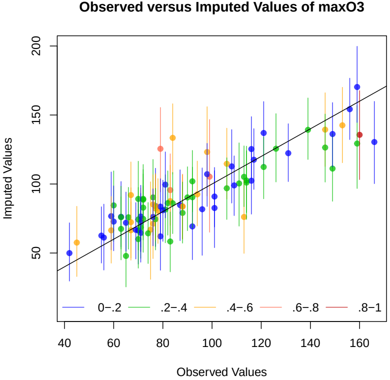
⇒ Step 1: Generate M imputed data sets

```
> library(Amelia)
> res.amelia <- amelia(don,m=100)  ##  in combination with zelig

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

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

Multiple imputation in practice

⇒ Step 2: visualization





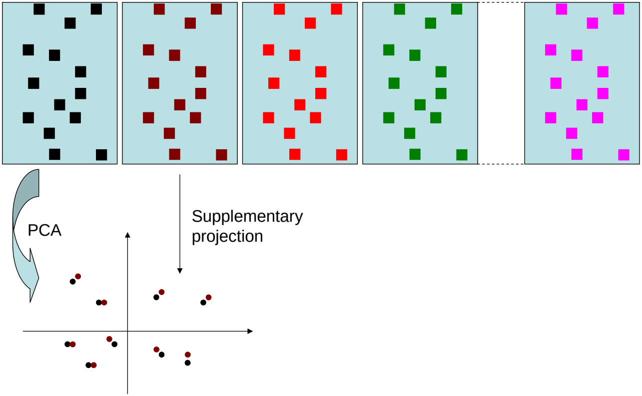

```
> library(Amelia)
> res.amelia <- amelia(don,m=100)
> compare.density(res.amelia, var="T12")
> overimpute(res.amelia, var="maxO3")
```

function `stripplot` in `mice`

Multiple imputation in practice

⇒ Step 2: visualization

⇒ Individuals position (and variables) with other predictions



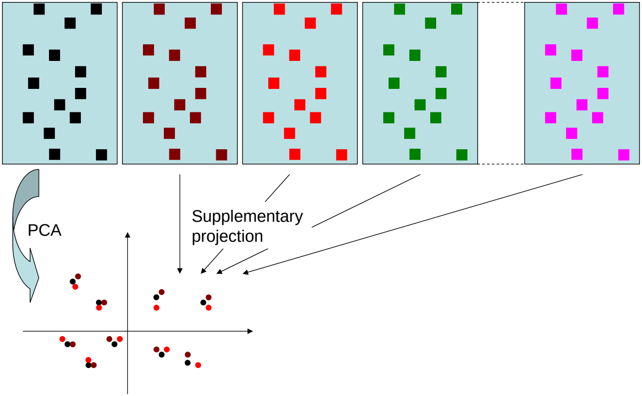
Regularized iterative PCA

⇒ reference configuration

Multiple imputation in practice

⇒ Step 2: visualization

⇒ Individuals position (and variables) with other predictions

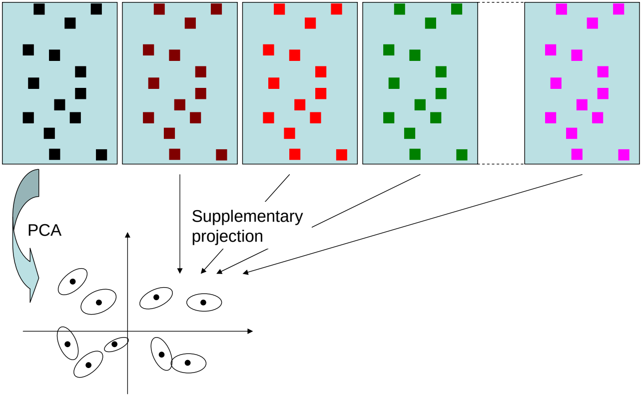


Regularized iterative PCA
⇒ reference configuration

Multiple imputation in practice

⇒ Step 2: visualization

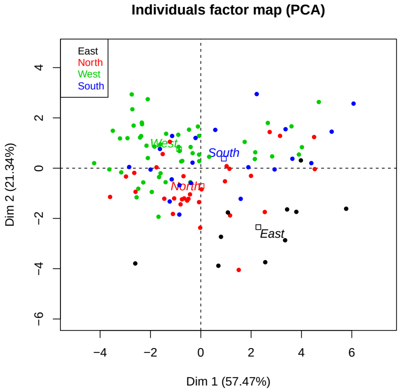
⇒ Individuals position (and variables) with other predictions

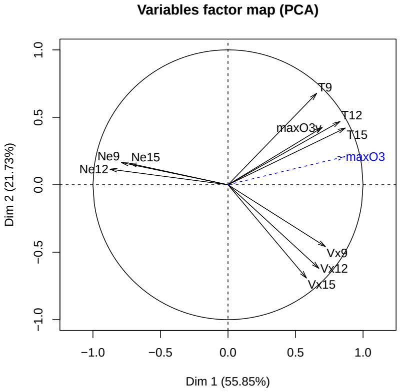


Regularized iterative PCA

⇒ reference configuration

PCA representation



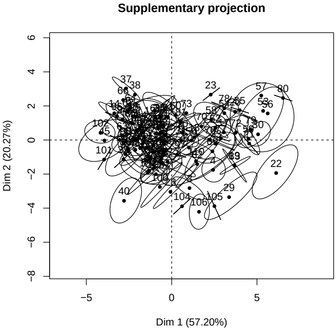


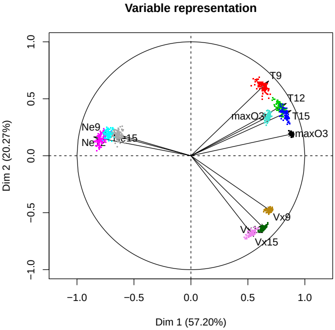

```
> imp <- cbind.data.frame(res.comp$completeObs,ozone[,12])
> res.pca <- PCA(imp,quanti.sup=1,quali.sup=12)
> plot(res.pca, hab=12, lab="quali"); plot(res.pca, choix="var")
> res.pca$ind$coord #scores (principal components)
```

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)
> imp.mice <- mice(don,m=100,defaultMethod="norm")
> lm.mice.out <- with(imp.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
```

Remarks

$\Rightarrow$ MI theory: good theory for regression parameters. Others?

$\Rightarrow$ Imputation model as complex as the analysis model (interaction)

Remarks

$\Rightarrow$ MI theory: good theory for regression parameters. Others?

$\Rightarrow$ Imputation model as complex as the analysis model (interaction)

$\Rightarrow$ Some practical issues:

- Imputation not in agreement (X and X^2): missing passive
- Imputation out of range?
- Problems of logical bounds (> 0) $\Rightarrow$ truncation?

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)
- Advanced methods are available to estimate parameters and their variance (taking into account the variability due to missing values)
- Multiple imputation is an appealing method .... but ... how can we do with big data?
- Still an active area of research

Ressources

⇒ Softwares:

- van Buuren webpage: http://www.stefvanbuuren.nl/mi/Software.html
- R task View: Official Statistics & Survey Methodology

⇒ Books:

- van Buuren (2012). *Flexible Imputation of Missing Data*. Chapman & Hall/CRC
- Carpenter & Kenward (2013). *Multiple Imputation and its Application*. Wiley
- G. Molenberghs, G. Fitzmaurice, M.G. Kenward, A. Tsiatis & G. Verbeke (nov 2014). *Handbook of Missing Data*. Chapman & Hall/CRC

⇒ J.L. Schafer & J.W. Graham, 2002. Missing Data: Our View of the State of the Art. *Psychological Methods*, **7** 147-177

Contributors on the topic of multiple imputation

- J. Honaker - G. King - M. Blackwell (Harvard): `Amelia`
- S. van Buuren (Utrecht): `mice`
- F. Husson - J. Josse (Rennes): `missMDA`
- A. Gelman - J. Hill - Y. Su (Colombia): `mi`
- J. Reiter (Duke): `NPBayesImpute` Non-Parametric Bayesian Multiple Imputation for Categorical Data
- J. Bartlett - J. Carpenter - M. Kenward (UCL): `smcfcs` Substantive model compatible FCS multiple imputation
- H. Goldstein (Bristol) : `realcom` for multi-level data
- J.K. Vermunt (Tilburg): `poLCA` latent class models

Conference on missing data

Thank you for your attention

http://missdata2015.agrocampus-ouest.fr/