## Supervised learning with missing values

Julie Josse
Ecole Polytechnique, INRIA
8 September 2020
Workshop on Missing Data Challenges in Computation, Statistics and Applications, Princeton


Introduction

## Collaborators on supervised learning with missing values

- M. Le Morvan, Postdoc at INRIA, Paris.
- E. Scornet, Associate Professor at Ecole Polytechnique, IP Paris.

Topic: random forests.

- G. Varoquaux, Senior researcher at INRIA, Paris.

Topic: machine learning. Creator of Scikitlearn in python.

$\Rightarrow$ Random Forests with missing values

1. Consistency of supervised learning with missing values. (2019). Revis JMLR.
$\Rightarrow$ Linear regression with missing values - MultiLayer perceptron
2. Linear predictor on linearly-generated data with missing values: non consistency and solutions. AISTAT2020.
3. Neumann networks: differential programming for supervised learning with missing values. Submitted Neurips2020.

## Traumabase project: decision support for trauma patients.

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

| Center | Accident | Age | Sex | Lactactes | BP | Shock | Platelet | $\ldots$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Beaujon | fall | 54 | m | NM | 180 | yes | 292000 |  |
| Pitie | gun | 26 | m | NA | 131 | no | 323000 |  |
| Beaujon | moto | 63 | m | 3.9 | NR | yes | 318000 |  |
| Pitie | moto | 30 | w | Imp | 107 | no | 211000 |  |
| HEGP | knife | 16 | m | 2.5 | 118 | no | 184000 |  |
| $\vdots$ |  |  |  |  |  |  |  | $\ddots$ |

## Traumabase project: decision support for trauma patients.

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

| Center | Accident | Age | Sex | Lactactes | BP | Shock | Platelet | $\ldots$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Beaujon | fall | 54 | m | NM | 180 | yes | 292000 |  |
| Pitie | gun | 26 | m | NA | 131 | no | 323000 |  |
| Beaujon | moto | 63 | m | 3.9 | NR | yes | 318000 |  |
| Pitie | moto | 30 | w | Imp | 107 | no | 211000 |  |
| HEGP | knife | 16 | m | 2.5 | 118 | no | 184000 |  |
| $\vdots$ |  |  |  |  |  |  |  | $\ddots$ |

$\Rightarrow$ Estimate causal effect: Administration of the treatment
"tranexamic acid" (within 3 hours after the accident) on the outcome mortality for traumatic brain injury patients. ${ }^{1}$

[^0]
## Traumabase project: decision support for trauma patients.

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

| Center | Accident | Age | Sex | Lactactes | BP | Shock | Platelet | $\ldots$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Beaujon | fall | 54 | m | NM | 180 | yes | 292000 |  |
| Pitie | gun | 26 | m | NA | 131 | no | 323000 |  |
| Beaujon | moto | 63 | m | 3.9 | NR | yes | 318000 |  |
| Pitie | moto | 30 | w | Imp | 107 | no | 211000 |  |
| HEGP | knife | 16 | m | 2.5 | 118 | no | 184000 |  |
| $\vdots$ |  |  |  |  |  |  |  | $\ddots$ |

$\Rightarrow$ Predict platelet levels given pre-hospital features
Ex linear regression/ random forests with covariates with missing values

## Missing values

Percentage of missing values


Different pattern: sporadic \& systematic (missing variable in one hospital) Different types: MCAR, MAR, MNAR

## Random Forests with missing

 values
## Missing values in a predictive framework (not inferential)

- Aim: target an outcome $Y$ (not estimate parameters and their variance)
- Specificities: train \& test sets with missing values
- Methods ${ }^{1}$ : (in practice) imputation prior to prediction
- Separate: impute train and test separately (with a different model)
- Grouped/ semi-supervised: impute train and test simultaneously but the predictive model is learned only on the training imputed data set.
- Imputation train and test sets with the same model Issue: methods (missForest) are "black-boxes" i.e. take as an input the incomplete data and output the completed data

Easy for univariate imputation: mean of each colum of the train.

[^1]
## Mean imputation is bad for estimation



Individuals factor map (PCA)


Variables factor map (PCA)



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

## EM-PCA

library (missMDA) imp <- imputePCA (ecolo) PCA (imp\$comp)
J. (2016). missMDA: Handling Missing Values in Multivariate Data Analysis, JSS.

Ecological data: ${ }^{2} n=69000$ species -6 traits. Estimated correlation between
Pmass \& Rmass $\approx 0$ (mean imputation) or $\approx 1$ (EM PCA)
${ }^{2}$ Wright, I. et al. (2004). The worldwide leaf economics spectrum. Nature.

## Constant (mean) imputation is consistent for prediction

$\tilde{X}=X \odot(1-M)+N A \odot M$. New feature space is $\widetilde{\mathbb{R}}^{d}=(\mathbb{R} \cup\{N A\})^{d}$.
$Y=\left(\begin{array}{l}4.6 \\ 7.9 \\ 8.3 \\ 4.6\end{array}\right) \quad \tilde{X}=\left(\begin{array}{lll}9.1 & \text { NA } & 1 \\ 2.1 & \text { NA } & 3 \\ \text { NA } & 9.6 & 2 \\ \text { NA } & 5.5 & 6\end{array}\right) \quad X=\left(\begin{array}{lll}9.1 & 8.5 & 1 \\ 2.1 & 3.5 & 3 \\ 6.7 & 9.6 & 2 \\ 4.2 & 5.5 & 6\end{array}\right) \quad M=\left(\begin{array}{lll}0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0\end{array}\right)$

Find a prediction function that minimizes the risk.

$$
\begin{aligned}
& \text { Bayes rule: } f^{*} \in \underset{f: \widetilde{\mathbb{R}}^{d} \rightarrow \mathbb{R}}{\arg \min } \mathbb{E}\left[(Y-f(\tilde{X}))^{2}\right] \\
& \begin{aligned}
f^{*}(\tilde{X}) & =\mathbb{E}[Y \mid \tilde{X}]=\mathbb{E}\left[Y \mid X_{o b s(M), M}\right] \\
& =\sum_{m \in\{0,1\}^{d}} \mathbb{E}\left[Y \mid X_{o b s(m)}, M=m\right] \mathbb{1}_{M=m}
\end{aligned}
\end{aligned}
$$

$\Rightarrow$ One model per pattern $\left(2^{d}\right)$ (Rubin, 1984, generalized propensity score)

## Constant (mean) imputation is consistent for prediction

$\tilde{X}=X \odot(1-M)+N A \odot M$. New feature space is $\widetilde{\mathbb{R}}^{d}=(\mathbb{R} \cup\{N A\})^{d}$.
$Y=\left(\begin{array}{l}4.6 \\ 7.9 \\ 8.3 \\ 4.6\end{array}\right) \quad \tilde{X}=\left(\begin{array}{lll}9.1 & \text { NA } & 1 \\ 2.1 & \text { NA } & 3 \\ \text { NA } & 9.6 & 2 \\ \text { NA } & 5.5 & 6\end{array}\right) \quad X=\left(\begin{array}{lll}9.1 & 8.5 & 1 \\ 2.1 & 3.5 & 3 \\ 6.7 & 9.6 & 2 \\ 4.2 & 5.5 & 6\end{array}\right) \quad M=\left(\begin{array}{lll}0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0\end{array}\right)$

Find a prediction function that minimizes the risk.

$$
\begin{aligned}
& \text { Bayes rule: } f^{*} \in \underset{f: \underset{\mathbb{R}^{d} \rightarrow \mathbb{R}}{ }}{\arg \min } \mathbb{E}\left[(Y-f(\tilde{X}))^{2}\right] \\
& \begin{aligned}
f^{*}(\tilde{X}) & =\mathbb{E}[Y \mid \tilde{X}]=\mathbb{E}\left[Y \mid X_{o b s(M), M}\right] \\
& =\sum_{m \in\{0,1\}^{d}} \mathbb{E}\left[Y \mid X_{o b s(m)}, M=m\right] \mathbb{1}_{M=m}
\end{aligned}
\end{aligned}
$$

$\Rightarrow$ One model per pattern $\left(2^{d}\right)$ (Rubin, 1984, generalized propensity score)

## Constant (mean) imputation is consistent

Framework - assumptions

- $Y=f(X)+\varepsilon$
- $X=\left(X_{1}, \ldots, X_{d}\right)$ has a continuous density $g>0$ on $[0,1]^{d}$
- $\|f\|_{\infty}<\infty$
- Missing data MAR on $X_{1}$ with $M_{1} \Perp X_{1} \mid X_{2}, \ldots, X_{d}$.
- $\left(x_{2}, \ldots, x_{d}\right) \mapsto \mathbb{P}\left[M_{1}=1 \mid X_{2}=x_{2}, \ldots, X_{d}=x_{d}\right]$ is continuous
- $\varepsilon$ is a centered noise independent of $\left(X, M_{1}\right)$
(remains valid when missing values occur for several variables $X_{1}, \ldots, X_{j}$ )


## Constant (mean) imputation is consistent

Constant imputed entry $x^{\prime}=\left(x_{1}^{\prime}, x_{2}, \ldots, x_{d}\right): x_{1}^{\prime}=x_{1} \mathbb{1}_{M_{1}=0}+\alpha \mathbb{1}_{M_{1}=1}$

## Theorem. (J. et al. 2019)

$$
\begin{aligned}
f_{\text {impute }}^{\star}\left(x^{\prime}\right)= & \mathbb{E}\left[Y \mid X_{2}=x_{2}, \ldots, X_{d}=x_{d}, M_{1}=1\right] \\
& \mathbb{1}_{\left.x_{1}^{\prime}=\alpha\right]} \mathbb{1}_{\mathbb{P}\left[M_{1}=1 \mid X_{2}=x_{2}, \ldots, X_{d}=x_{d}\right]>0} \\
& +\mathbb{E}\left[Y \mid X=x^{\prime}\right] \mathbb{1}_{x_{1}^{\prime}=\alpha} \mathbb{1}_{\mathbb{P}\left[M_{1}=1 \mid X_{2}=x_{2}, \ldots, X_{d}=x_{d}\right]=0} \\
& +\mathbb{E}\left[Y \mid X_{1}=x_{1}, X_{2}=x_{2}, \ldots, X_{d}=x_{d}, M_{1}=0\right] \mathbb{1}_{x_{1}^{\prime} \neq \alpha} .
\end{aligned}
$$

Prediction with mean is equal to the Bayes function almost everywhere

$$
f_{\text {impute }}^{\star}\left(X^{\prime}\right)=f^{\star}(\tilde{X})=\mathbb{E}[Y \mid \tilde{X}=\tilde{x}]
$$

Rq: pointwise equality if using a constant out of range.
$\Rightarrow$ Learn on the mean-imputed training data, impute the test set with the same means and predict is optimal if the missing data are MAR and the learning algorithm is universally consistent

## Consistency of supervised learning with NA: Rationale

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


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

Empirically good results for MNAR

## Consistency of supervised learning with NA: Rationale

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


Train


Test

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

Empirically good results for MNAR

## CART (Breiman, 1984)

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

$$
\begin{aligned}
\left(j^{\star}, z^{\star}\right) \in \underset{(j, z) \in \mathcal{S}}{\arg \min } \mathbb{E} & {\left[\left(Y-\mathbb{E}\left[Y \mid X_{j} \leq z\right]\right)^{2} \cdot \mathbb{1}_{X_{j} \leq z}\right.} \\
& \left.+\left(Y-\mathbb{E}\left[Y \mid X_{j}>z\right]\right)^{2} \cdot \mathbb{1}_{X_{j}>z}\right] .
\end{aligned}
$$



## CART (Breiman, 1984)

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

$$
\begin{aligned}
\left(j^{\star}, z^{\star}\right) \in \underset{(j, z) \in \mathcal{S}}{\arg \min } \mathbb{E} & {\left[\left(Y-\mathbb{E}\left[Y \mid X_{j} \leq z\right]\right)^{2} \cdot \mathbb{1}_{X_{j} \leq z}\right.} \\
& \left.+\left(Y-\mathbb{E}\left[Y \mid X_{j}>z\right]\right)^{2} \cdot \mathbb{1}_{X_{j}>z}\right] .
\end{aligned}
$$



## CART (Breiman, 1984)

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

$$
\begin{aligned}
\left(j^{\star}, z^{\star}\right) \in \underset{(j, z) \in \mathcal{S}}{\arg \min } \mathbb{E} & {\left[\left(Y-\mathbb{E}\left[Y \mid X_{j} \leq z\right]\right)^{2} \cdot \mathbb{1}_{X_{j} \leq z}\right.} \\
& \left.+\left(Y-\mathbb{E}\left[Y \mid X_{j}>z\right]\right)^{2} \cdot \mathbb{1}_{X_{j}>z}\right]
\end{aligned}
$$



## CART with missing values

root

|  | $X_{1}$ | $X_{2}$ | Y |
| :--- | :--- | :--- | :--- |
| 1 |  |  |  |
| 2 | NA |  |  |
| 3 | NA |  |  |
| 4 |  |  |  |

## CART with missing values

|  | $X_{1}$ | $X_{2}$ | $Y$ |
| ---: | :---: | :---: | :---: |
| 1 |  |  |  |
| 2 | NA |  |  |
| 3 | NA |  |  |
| 4 |  |  |  |



1) Select variable and threshold on observed values ( $1 \& 4$ for $X_{1}$ )
$\mathbb{E}\left[\left(Y-\mathbb{E}\left[Y \mid X_{j} \leq z, M_{j}=0\right]\right)^{2} \cdot \mathbb{1}_{X_{j} \leq z, M_{j}=0}+\left(Y-\mathbb{E}\left[Y \mid X_{j}>z, M_{j}=0\right]\right)^{2} \cdot \mathbb{1}_{X_{j}>z, M_{j}=0}\right]$.

## CART with missing values

|  | $X_{1}$ | $X_{2}$ | $Y$ |
| :---: | :---: | :---: | :---: |
| 1 |  |  |  |
| 2 | NA |  |  |
| 3 | NA |  |  |
| 4 |  |  |  |

$$
x_{1} \leq \stackrel{s}{1}_{\text {root }}^{X_{1}>s_{1}}
$$

1) Select variable and threshold on observed values ( $1 \& 4$ for $X_{1}$ )
$\mathbb{E}\left[\left(Y-\mathbb{E}\left[Y \mid X_{j} \leq z, M_{j}=0\right]\right)^{2} \cdot \mathbb{1}_{X_{j} \leq z, M_{j}=0}+\left(Y-\mathbb{E}\left[Y \mid X_{j}>z, M_{j}=0\right]\right)^{2} \cdot \mathbb{1}_{X_{j}>z, M_{j}=0}\right]$.
2) Propagate observations $(2 \& 3)$ with missing values?


- Block: Send all to a side by minimizing the error (xgboost, lightgbm)
- Surrogate split: Search another variable that gives a close partition (rpart)


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

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

1. $\left\{\widetilde{X}_{j} \leq z\right.$ or $\left.\widetilde{X}_{j}=\mathrm{NA}\right\}$ vs $\left\{\widetilde{X}_{j}>z\right\}$
2. $\left\{\widetilde{X}_{j} \leq z\right\}$ vs $\left\{\widetilde{X}_{j}>z\right.$ or $\left.\widetilde{X}_{j}=\mathrm{NA}\right\}$
3. $\left\{\widetilde{X}_{j} \neq \mathrm{NA}\right\}$ vs $\left\{\widetilde{X}_{j}=\mathrm{NA}\right\}$.

- The splitting location $z$ depends on the missing values
- Missing values treated like a category (well to handle $\mathbb{R} \cup N A$ )
- Good for informative pattern ( $M$ explains $Y$ )

Targets one model per pattern:

$$
\mathbb{E}[Y \mid \tilde{X}]=\sum_{m \in\{0,1\}^{d}} \mathbb{E}\left[Y \mid X_{o b s(m)}, M=m\right] \mathbb{1}_{M=m}
$$

- Implementation ${ }^{3}$ : grf package, scikit-learn, partykit
$\Rightarrow$ Extremely good performances in practice for any mechanism.

[^2]
## Consistency: 40\% missing values MCAR

Linear problem (high noise)


Sample size


- Surrogates (rpart)
- Mean imputation

Friedman problem (high noise)


Sample size


- Gaussian imputation - MIA

Non-linear problem (low noise)



## - Bayes rate

- Block (XGBoost)

Linear regression with missing values - MLP

## Explicit Bayes predictor with missing values

## Linear model:

$$
Y=\beta_{0}+\langle X, \beta\rangle+\varepsilon, \quad X \in \mathbb{R}^{d}, \varepsilon \text { gaussian. }
$$

Bayes predictor for the linear model:

$$
\begin{aligned}
f^{\star}(\tilde{X}) & =\mathbb{E}[Y \mid \tilde{X}]=\mathbb{E}\left[\beta_{0}+\beta^{\top} X \mid M, X_{o b s(M)}\right] \\
& =\beta_{0}+\beta_{o b s(M)}^{\top} X_{o b s(M)}+\beta_{\operatorname{mis}(M)}^{\top} \mathbb{E}\left[X_{\operatorname{mis}(M)} \mid M, X_{o b s(M)}\right]
\end{aligned}
$$

## Assumptions on covariates and missing values

Gaussian pattern mixture model (PMM): $X \mid(M=m) \sim \mathcal{N}\left(\mu_{m}, \Sigma_{m}\right)$ Gaussian assumption $X \sim \mathcal{N}(\mu, \Sigma)+$ MCAR and MAR

## Under Assump. the Bayes predictor is linear per pattern

$f^{\star}\left(X_{o b s}, M\right)=\beta_{0}^{\star}+\left\langle\beta_{o b s}^{\star}, X_{o b s}\right\rangle+\left\langle\beta_{m i s}^{\star}, \mu_{m i s}+\sum_{m i s, o b s}\left(\sum_{o b s}\right)^{-1}\left(X_{o b s}-\mu_{o b s}\right)\right\rangle$ use of obs instead of $o b s(M)$ for lighter notations
(Also for Gaussian assumption + MNAR self mask gaussian)

## Estimation of the bayes predictor

## Under Assumpt. the Bayes predictor is linear per pattern

$f^{\star}\left(X_{o b s}, M\right)=\beta_{0}^{\star}+\left\langle\beta_{o b s}^{\star}, X_{o b s}\right\rangle+\left\langle\beta_{m i s}^{\star}, \mu_{m i s}+\Sigma_{m i s, o b s}\left(\Sigma_{o b s}\right)^{-1}\left(X_{o b s}-\mu_{o b s}\right)\right\rangle$
Classical method: use Max Likelihood (EM algo) to estimate $\Sigma$.
Issues: available implementation strugle with large $d$.
Most methods for MAR data or few MNAR variables.

## Theorem. Bayes consistency of a MLP. Le Morvan et al. (2020)

Under linear model + Gaussian pattern mixture model, a MLP:

- with one hidden layer containing $2^{d}$ hidden units
- ReLU activation functions
- fed with $[X \odot(1-M), M]$ ( $\tilde{X}$ imputed by 0 concatenated with mask) can achieve the Bayes rate.

Rationale: The MLP produces a prediction function piecewise affine.
Rq: reduce the model capacity by reducing the number of hidden units.

## Neuman Networks to approximate the covariance matrix

The Bayes predictor is linear per pattern
$f^{\star}\left(X_{o b s}, M\right)=\beta_{0}^{\star}+\left\langle\beta_{o b s}^{\star}, X_{o b s}\right\rangle+\left\langle\beta_{m i s}^{\star}, \mu_{m i s}+\Sigma_{m i s, o b s}\left(\sum_{o b s}\right)^{-1}\left(X_{o b s}-\mu_{o b s}\right)\right\rangle$
Order- $\ell$ approx of $\left(\Sigma_{o b s(m)}^{-1}\right)$ for any $m$ defined recursively:

$$
S_{o b s(m)}^{(\ell)}=\left(I d-\Sigma_{o b s(m)}\right) S_{o b s(m)}^{(\ell-1)}+I d .
$$

Neuman Series, $S^{(0)}=I d, \ell=\infty:\left(\Sigma_{o b s(m)}\right)^{-1}=\sum_{k=0}^{\infty}\left(I d-\Sigma_{o b s(m)}\right)^{k}$

## Neuman Networks to approximate the covariance matrix

## Order- $\ell$ approx of the Bayes predictor in MAR

$$
f_{\ell}^{\star}\left(X_{o b s}, M\right)=\left\langle\beta_{o b s}, X_{o b s}\right\rangle+\left\langle\beta_{m i s}, \mu_{m i s}+\sum_{m i s, o b s} S_{o b s(m)}^{(\ell)}\left(X_{o b s}-\mu_{o b s}\right)\right\rangle .
$$

Order- $\ell$ approx of $\left(\Sigma_{o b s(m)}^{-1}\right)$ for any $m$ defined recursively:

$$
S_{o b s(m)}^{(\ell)}=\left(I d-\Sigma_{o b s(m)}\right) S_{o b s(m)}^{(\ell-1)}+I d .
$$

Neuman Series, $S^{(0)}=I d, \ell=\infty:\left(\Sigma_{o b s(m)}\right)^{-1}=\sum_{k=0}^{\infty}\left(I d-\Sigma_{o b s(m)}\right)^{k}$
$\Rightarrow$ Neural network architecture to approximate the Bayes predictor


Figure 1: Depth of $3, \bar{m}=1-m$. Each weight matrix $W^{(k)}$ corresponds to a simple transformation of the covariance matrix indicated in blue.

## Networks with missing values: $\odot M$ nonlinearity



- Implementing a network with the matrix weights $W^{(k)}=\left(I-\Sigma_{o b s(m)}\right)$ masked differently for each sample can be challenging
- Masked weights is equivalent to masking input \& output vector. Let $v$ a vector, $\bar{m}=1-m .\left(W \odot \bar{m} \bar{m}^{\top}\right) v=(W(v \odot \bar{m})) \odot \bar{m}$

Classic network with multiplications by the mask nonlinearities $\odot M$

## Networks with missing values: $\odot M$ nonlinearity



- Implementing a network with the matrix weights $W^{(k)}=\left(I-\Sigma_{o b s(m)}\right)$ masked differently for each sample can be challenging
- Masked weights is equivalent to masking input \& output vector. Let $v$ a vector, $\bar{m}=1-m .\left(W \odot \bar{m} \bar{m}^{\top}\right) v=(W(v \odot \bar{m})) \odot \bar{m}$

Classic network with multiplications by the mask nonlinearities $\odot M$

## Proposition (equivalence MLP - depth-0 Neumann network)

A MLP with ReLU activations, one hidden layer of $d$ hidden units, and which operates on the $[X \odot(1-M), M]$, the input $X$ imputed by 0 concatenated with the mask $M$, is equivalent to the 0 -depth $N N$

## Experiments for linear regression with missing values

- Max Likelihood: to estimate the parameters of the joint Gaussian distribution $\left(X_{1}, \ldots, X_{d}, Y\right)$ with EM. Predict by conditional expectation of $Y$ given $X_{\text {obs }}$.
- ICE + LR: conditional imputation with an iterative imputer followed by linear regression.
- MLP: take as input the data imputed by 0 concatenated with the mask $[X \odot(1-M), M]$ with ReLU nonlinearity,
- MLP-Wide: one hidden layer with width increased (between $d \& 2^{d}$ )
- MLP-Deep: 1 to 10 hidden layers of $d$ hidden units
- Neumann: The Neumann architecture with the $\odot M$, choosing the depth on a validation set.


## Results



Figure 2: Predictive performances in various scenarios - varying missing-value mechanisms, number of samples $n$, and number of features $d$.
$\Rightarrow$ Best performances for MNAR scenario ( $50 \%$ of NA on all variables)

- More effective to increase the capacity of the Neumann network (depth) than to increase the capacity (width) of MLP Wide.
- Neumann network learn improved weights compared to Neumann iterations

Discussion - challenges

## Take-home message. Supervised learning with missing values.

Supervised learning different from usual inferential probabilistic models. Solutions useful in practice robust to the missing-value mechanisms but needs powerful model.

## Powerful learner with missing values

- Incomplete train and test $\rightarrow$ same imputation model
- Single constant imputation is consistent with a powerful learner
- Tree-based models: Missing Incorporated in Attribute
- To be done: nonasymptotic results, uncertainty, distributional shift: No NA in the test? Proofs in MNAR


## Linear regression with missing values

- The Bayes predictor is explicit under Gaussian assumptions/ MAR and gaussian self mask but high-dimensional.
- Approx include MLP which can be consistent and Neuman Network
- New architecture for network with missing data: $\odot M$ nonlinearity.


## Ressources

R-miss-tastic https://rmisstastic.netlify.com/R-miss-tastic
J., I. Mayer, N. Tierney \& N. Vialaneix

Project funded by the R consortium (Infrastructure Steering Committee) ${ }^{4}$
Aim: a reference platform on the theme of missing data management

- list existing packages
- available literature
- tutorials
- analysis workflows on data
- main actors
$\Rightarrow$ Federate the community
$\Rightarrow$ Contribute!
${ }^{4}$ https://www.r-consortium.org/projects/call-for-proposals


## Ressources

Examples:

- Lecture ${ }^{5}$ - General tutorial : Statistical Methods for Analysis with Missing Data (Mauricio Sadinle)
- Lecture - Multiple Imputation: mice by Nicole Erler ${ }^{6}$
- Longitudinal data, Time Series Imputation (Steffen Moritz - very active contributor of $r$-miss-tastic), Principal Component Methods ${ }^{7}$

```
5}\mathrm{ https://rmisstastic.netlify.com/lectures/
6}\mathrm{ https://rmisstastic.netlify.com/tutorials/erler_course_
multipleimputation_2018/erler_practical_mice_2018
    7https://rmisstastic.netlify.com/tutorials/Josse_slides_imputation_PCA_2018.pdf
```


## Thank you



1) Those who can extrapolate from incomplete data

[^0]:    ${ }^{1}$ Doubly robust treatment effect estimation with incomplete confounders. Mayer, Wager, J. Annals Of Applied Statistics 2020.

[^1]:    ${ }^{1}$ Rmistatic platform to organize ressources - Task view: more than 150 packages

[^2]:    ${ }^{3}$ implementation trick, J. Tibshirani, duplicate the incomplete columns, and replace

