

NeuMiss networks: differentiable programming for supervised learning with missing values

Marine Le Morvan^{1,2} **Julie Josse**^{1,3} **Thomas Moreau**¹
Erwan Scornet³ **Gaël Varoquaux**^{1, 4}

¹ Université Paris-Saclay, Inria, CEA, Palaiseau, 91120, France

² Université Paris-Saclay, CNRS/IN2P3, IJCLab, 91405 Orsay, France

³ CMAP, UMR7641, Ecole Polytechnique, IP Paris, 91128 Palaiseau, France

⁴ Mila, McGill University, Montréal, Canada

NeurIPS 2020



Linear regression with missing values

- Assume the data follows a linear model:

$$Y = \beta_0^* + \sum_{j=1}^d \beta_j^* X_j + \epsilon, \quad X \sim \mathcal{N}(\mu, \Sigma)$$

Complete data
(unavailable)

- The optimal predictor (Bayes predictor) is given by:

$$f^* \in \underset{f: (\mathbb{R} \cup \{\text{NA}\})^d \mapsto \mathbb{R}}{\operatorname{argmin}} \mathbb{E} \left[\left(Y - f(\tilde{X}) \right)^2 \right]$$

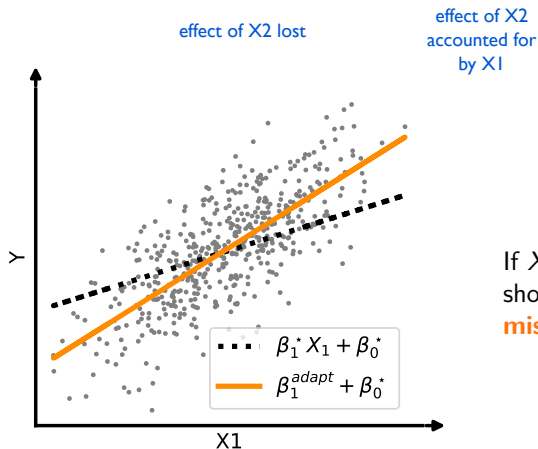
Incomplete data
(available)

- In this work, we show that **for certain missing data mechanisms**, the Bayes predictor is linear per pattern :

$$f^*(\tilde{X}, M) = \beta_0(M) + \sum_{j \in \operatorname{obs}(M)} \beta_j(M) X_j$$

Slopes depend on the
missing data pattern

Intuition



$$Y = \beta_1^* X_1 + \beta_2^* X_2$$

$$\text{cor}(X_1, X_2) = 0.5.$$

If X_2 is missing, the coefficient of X_1 should **compensate for the missingness of X_2** .

For a general problem in d dimensions, there are 2^d **possible missing data patterns**, making the Bayes predictor hard to approximate.

Content

- 1 Optimal predictors in the presence of missing values
- 2 NeuMiss networks: learning by approximating the Bayes predictor
- 3 Empirical results

Outline

- 1 Optimal predictors in the presence of missing values
- 2 NeuMiss networks: learning by approximating the Bayes predictor
- 3 Empirical results

Notations and assumptions

Random variables

- $X \in \mathbb{R}^d$: complete data (unavailable)
- $\tilde{X} \in \{\mathbb{R} \cup \{\text{NA}\}\}^d$: incomplete data (available)
- $M \in \{0, 1\}^d$: mask.

$obs(M)$ (resp. $mis(M)$) are the indices of the observed (resp. missing) entries.

Notation abuse: $A_{obs(m), obs(m)} = A_{obs(m)}$

Assumptions:

linear model + Gaussian data:

$$Y = \beta_0^* + \sum_{j=1}^d \beta_j^* X_j + \epsilon,$$

$$X \sim \mathcal{N}(\mu, \Sigma)$$

Ex. of realizations

$$x = (1.1, 2.3, 3.1, 8, 5.27)$$

$$\tilde{x} = (1.1, \text{NA}, -3.1, 8, \text{NA})$$

$$m = (0, 1, 0, 0, 1)$$

$$x_{obs(m)} = (1.1, 3.1, 8),$$

$$x_{mis(m)} = (2.3, 5.27)$$

Bayes predictor:

$$f^* \in \underset{f: (\mathbb{R} \cup \{\text{NA}\})^d \mapsto \mathbb{R}}{\operatorname{argmin}} \mathbb{E} \left[\left(Y - f(\tilde{X}) \right)^2 \right]$$

The Bayes predictor under M(C)AR

- **MCAR:** For all $m \in \{0, 1\}^d$, $P(M = m|X) = P(M = m)$.
- **MAR:** For all $m \in \{0, 1\}^d$, $P(M = m|X) = P(M = m|X_{obs(m)})$.

Proposition (M(C)AR Bayes predictor)

Under the linear model and Gaussian data assumptions, and a MCAR or MAR missing data mechanism, the Bayes predictor f^ takes the form:*

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis, obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$

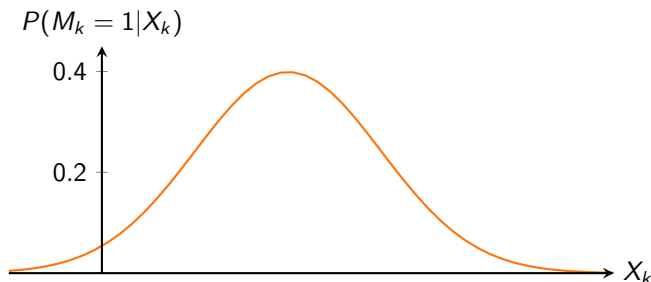
General idea of the proof:

$$\begin{aligned} f^*(X_{obs}, M) &= \mathbb{E}[Y | X_{obs(M)}, M] \\ &= \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mathbb{E}[X_{mis} | X_{obs}, M] \rangle \\ &= \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mathbb{E}[X_{mis} | X_{obs}] \rangle \end{aligned}$$

The Bayes predictor under Gaussian self-masking (MNAR)

- **Gaussian self-masking (MNAR):** The missing data mechanism is self-masked with $P(M|X) = \prod_{k=1}^d P(M_k|X_k)$ and $\forall k \in \llbracket 1, d \rrbracket$,

$$P(M_k = 1|X_k) = K_k \exp\left(-\frac{1}{2} \frac{(X_k - \tilde{\mu}_k)^2}{\tilde{\sigma}_k^2}\right) \quad \text{with } 0 < K_k < 1.$$



The Bayes predictor under Gaussian self-masking (MNAR)

- **Gaussian self-masking (MNAR):** The missing data mechanism is self-masked with $P(M|X) = \prod_{k=1}^d P(M_k|X_k)$ and $\forall k \in \llbracket 1, d \rrbracket$,

$$P(M_k = 1|X_k) = K_k \exp\left(-\frac{1}{2} \frac{(X_k - \tilde{\mu}_k)^2}{\tilde{\sigma}_k^2}\right) \quad \text{with } 0 < K_k < 1.$$

Proposition (Gaussian self-masking (MNAR) Bayes predictor)

Under the linear model and Gaussian data assumptions, and a Gaussian self-masking (MNAR) missing data mechanism, the Bayes predictor f^ takes the form:*

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, (Id + D_{mis} \Sigma_{mis|obs}^{-1})^{-1} \\ \times (\tilde{\mu}_{mis} + D_{mis} \Sigma_{mis|obs}^{-1} (\mu_{mis} + \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}))) \rangle$$

where $\Sigma_{mis|obs} = \Sigma_{mis,mis} - \Sigma_{mis,obs} \Sigma_{obs}^{-1} \Sigma_{obs,mis}$ and $D = \text{diag}(\tilde{\sigma}_1^2, \dots, \tilde{\sigma}_d^2)$.

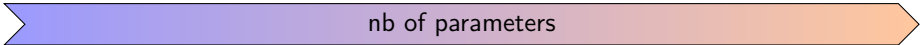
Outline

- 1 Optimal predictors in the presence of missing values
- 2 NeuMiss networks: learning by approximating the Bayes predictor
- 3 Empirical results

How to approximate the Bayes predictors?

M(C)AR Bayes predictor:

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$

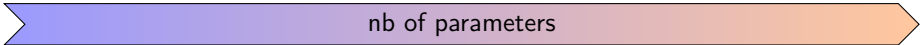


nb of parameters

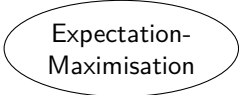
How to approximate the Bayes predictors?

M(C)AR Bayes predictor:

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$



nb of parameters



Expectation-
Maximisation

$O(d^2)$ parameters

No robustness to the missing data mech.

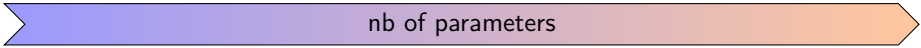
High computational complexity!!!

(untractable when d reaches a few dozens)

How to approximate the Bayes predictors?

M(C)AR Bayes predictor:

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$



nb of parameters

Expectation-
Maximisation

MLP

$O(d^2)$ parameters

$O(2^d)$ parameters

No robustness to the
missing data mech.

Largely
over-parametrized.

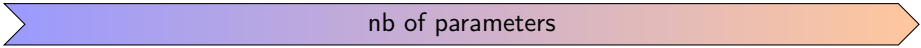
High computational
complexity!!!

(untractable when d
reaches a few dozens)

How to approximate the Bayes predictors?

M(C)AR Bayes predictor:

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$



nb of parameters

Expectation-
Maximisation

NeuMiss
networks

MLP

$O(d^2)$ parameters

$O(d^2)$ parameters

$O(2^d)$ parameters

No robustness to the
missing data mech.

Sharing parameters across
missing data patterns

Largely
over-parametrized.

High computational
complexity!!!

$O(d^2)$ computational complexity.

(untractable when d
reaches a few dozens)

Differentiable approximations of the inverse covariances

- Neumann series:

$$(\Sigma_{obs(m)})^{-1} = \sum_{k=0}^{\infty} (Id_{obs(m)} - \Sigma_{obs(m)})^k$$

if the spectral radius of Σ is strictly smaller than 1.

- We propose to approximate $(\Sigma_{obs(m)})^{-1}$, for any m , by an order- ℓ approximation $S_{obs(m)}^{(\ell)}$, defined recursively as:

$$S_{obs(m)}^{(\ell)} = (Id_{obs(m)} - \Sigma_{obs(m)})S_{obs(m)}^{(\ell-1)} + Id.$$

Differentiable approximations of the inverse covariances

Define the order- ℓ approximation of the Bayes predictor in M(C)AR settings

$$f_\ell^*(X_{obs}, M) = \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis, obs} S_{obs(m)}^{(\ell)} (X_{obs} - \mu_{obs}) \rangle.$$

Proposition (Risk of the order- ℓ approximation)

Suppose that the spectral radius of Σ is strictly smaller than one. Then under the linear model and Gaussian data assumptions, and a MCAR or MAR missing data mechanism, for all $\ell \geq 1$,

$$\mathbb{E} \left[(f_\ell^*(X_{obs}, M) - f^*(X_{obs}, M))^2 \right] \leq \frac{(1 - \nu)^{2\ell} \|\beta^*\|_2^2}{\nu} \mathbb{E} \left[\left\| Id - S_{obs(M)}^{(0)} \Sigma_{obs(M)} \right\|_2^2 \right]$$

where ν is the smallest eigenvalue of Σ .

NeuMiss network architecture

- M(C)AR Bayes predictor:

$$f_{\ell}^*(X_{obs}, M) \approx \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} S_{obs}^{(\ell)}(X_{obs} - \mu_{obs}) \rangle$$

- Approximation of $(\Sigma_{obs})^{-1}$: $S_{obs(m)}^{(\ell)} = (Id_{obs(m)} - \Sigma_{obs(m)})S_{obs(m)}^{(\ell-1)} + Id_{obs(m)}$
- NeuMiss network architecture (illustrated with a depth of 4):

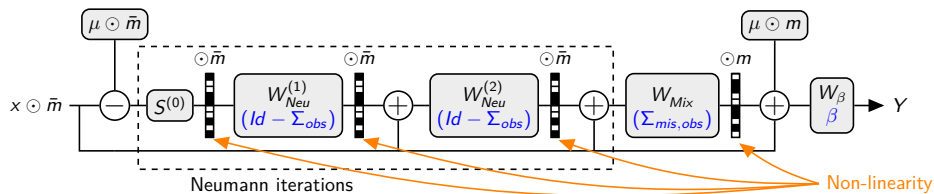


Figure: $\bar{m} = 1 - m$. Each weight matrix $W_{Neu}^{(k)}$ corresponds to a simple transformation of the covariance matrix indicated in blue.

Neumann network and Gaussian self-masking (MNAR)

- M(C)AR Bayes predictor:

$$f^*(X_{obs}, M) = \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$

- Suppose that $D_{mis} \Sigma_{mis|obs}^{-1} \approx \hat{D}_{mis}$ where \hat{D} is a diagonal matrix. Then the Gaussian self-masking Bayes predictor is:

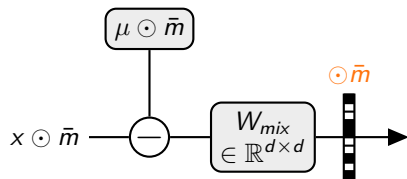
$$f^*(X_{obs}, M) \approx \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, (Id_{mis} + \hat{D}_{mis})^{-1} (\tilde{\mu}_{mis} + \hat{D}_{mis} \mu_{mis}) + (Id_{mis} + \hat{D}_{mis})^{-1} \hat{D}_{mis} \Sigma_{mis,obs} (\Sigma_{obs})^{-1} (X_{obs} - \mu_{obs}) \rangle$$

The self-masking Bayes predictor can be well approximated by **adjusting the values learned for the params μ and W_{mix}** if $D_{mis} \Sigma_{mis|obs}^{-1}$ are close to diagonal.

Link with the feedforward network

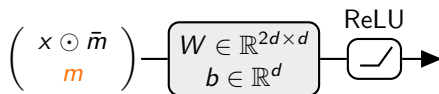
NeuMiss depth-1 layer

$$\mathcal{H}_{\odot m} : \mathbb{R}^d \mapsto \mathbb{R}^d$$



Feedforward layer (d hidden units)

$$\mathcal{H}_{ReLU} : \mathbb{R}^d \times \{0, 1\} \mapsto \mathbb{R}^d$$



Proposition (equivalence MLP - depth-1 NeuMiss network)

Denote by h_k^{ReLU} and $h_k^{\odot m}$ the output of the k^{th} hidden units of each layer. Then there exists a configuration of the weights of \mathcal{H}_{ReLU} such that

$$\forall k, \forall (m, x_{obs(m)}), \quad h_k^{ReLU}(x_{obs}, m) = h_k^{\odot m}(x_{obs}, m) + c_k, \quad c_k \in \mathbb{R}$$

Outline

- 1 Optimal predictors in the presence of missing values
- 2 NeuMiss networks: learning by approximating the Bayes predictor
- 3 Empirical results

The $\odot m$ nonlinearity is crucial to the performance

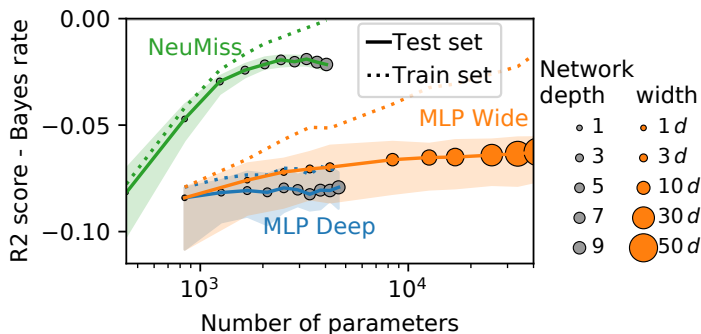


Figure: Performance as a function of capacity across architectures — Data are generated under a linear model with Gaussian covariates in a MCAR setting (50% missing values, $n = 10^5$, $d = 20$).

Approximation learned by the Neumann network

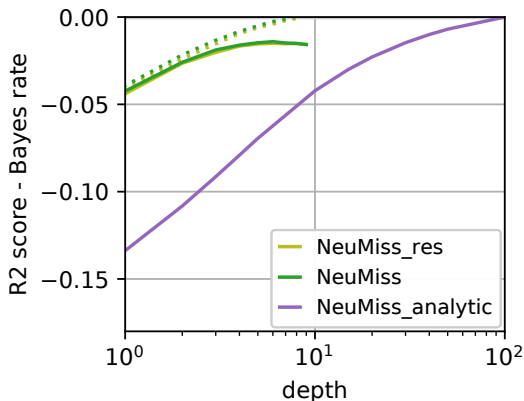


Figure: Left: learned versus analytic Neumann iterates — Neumann analytic is the Neumann architecture with weights set to represent the Bayes predictor, supposing we have access to the ground truth parameters, Neumann (resp. Neumann res) corresponds to the network without (resp. with) residual connections. MCAR setting (50% missing values, $n = 10^5$, $d = 20$).

Neumann networks require $O(d^2)$ samples

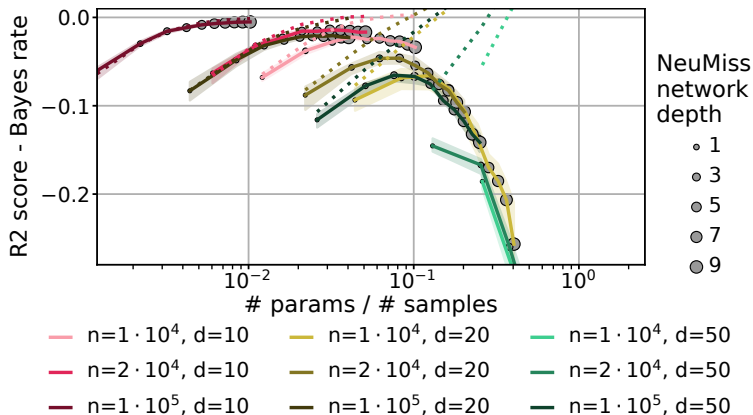


Figure: Required capacity in various settings — Performance of Neumann networks varying the depth in simulations with different number of samples n and of features d . MCAR setting (50% missing values)

Comparison of performances with competitors

- Data
 - ▶ linear model
 - ▶ Gaussian data
 - ▶ SNR = 10
- Missing data mechanisms (50% missing values)
 - ▶ MCAR
 - ▶ MAR
 - ▶ Gaussian self-masking (MNAR)
 - ▶ Probit self-masking (MNAR)
- Methods
 - ▶ **EM**: Expectation-Maximisation.
 - ▶ **Mice + LR**: Conditional imputation followed by linear regression.
 - ▶ **MLP**: 1 hidden layer with varied nb of hidden units (between d and $100d$), ReLU nonlinearities, data imputed by 0 concatenated with the mask as input, ADAM, adaptative learning rate.
 - ▶ **NeuMiss** The NeuMiss architecture, depth varied between 0 and 10, SGD, adaptative learning rate.

Comparison of performances with competitors

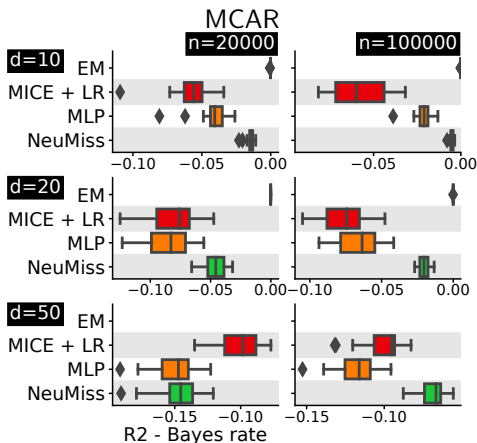
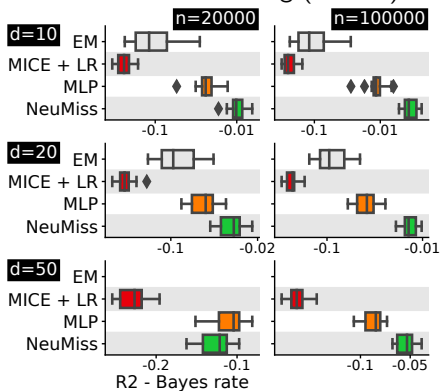


Figure: Predictive performances in various scenarios — varying missing-value mechanisms, number of samples n , and number of features d . All experiments are repeated 20 times. For self-masking settings, the x-axis is in log scale, to accommodate the large difference between methods.

Comparison of performances with competitors

Gaussian self-masking (MNAR)



Probit self-masking (MNAR)

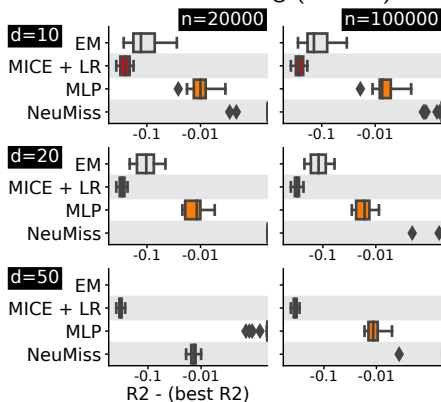
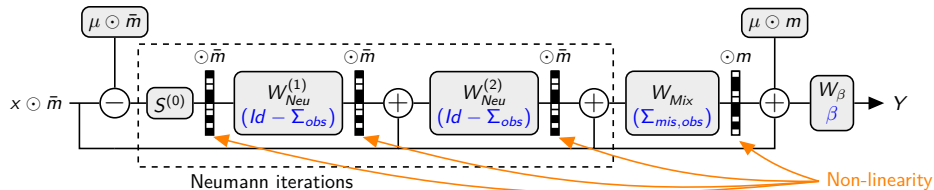


Figure: Predictive performances in various scenarios — varying missing-value mechanisms, number of samples n , and number of features d . All experiments are repeated 20 times. For self-masking settings, the x-axis is in log scale, to accommodate the large difference between methods.

Take away



- **Theoretically-grounded** architecture,
- with a new type of non-linearity: **the \odot non-linearity**.
- **Robustness to the missing data mechanism**.
- **Suited for medium-sized datasets** thanks to weight sharing across missing data patterns.

Thank you for your attention!