NeuMiss networks: differentiable programming for supervised learning with missing values

Marine Le Morvan$^{1,2}$

Julie Josse$^{1,3}$  
Thomas Moreau$^1$

Erwan Scornet$^3$  
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

Montpellier - December 2020
Incomplete data is ubiquitous in many fields

Sources of missingness:
- Survey nonresponse.
- Sensor failure.
- Changing data gathering procedure.
- Database join.
- ...

Missing data is frequent in economics, social, political or health sciences.

Traumabase clinical health records.
The classical literature on missing values

Since the 70s, an abundant literature on missing data has flourished.

- Missing data mechanisms are usually divided into 3 categories:
  - MCAR (Missing Completely at Random)
  - MAR (Missing at Random)
  - MNAR (Missing Non At Random)

- The literature has been mainly focused on inference and imputation tasks:
  - Likelihood based methods under MAR.
  - Multiple imputation under MAR.
  - Inverse probability weighting under MAR.

But very few works have addressed supervised learning with missing values, whatever the missing data mechanism.
Intuition: linear regression with missing values

Y = \beta_1^* X_1 + \beta_2^* X_2 + \beta_0^*

\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.

The difficulty of supervised learning with missing values is to handle up to \(2^d\) missing data patterns (i.e. \(2^d\) possible inputs of varying length).
The literature on supervised learning with missing values

Some recent works:

▶ Josse et al. 2019: Imputation by a constant is Bayes consistent, but the function to be learned can be overly complex (hyp: MAR).

▶ Le Morvan et al. 2020: In the simple case of linear regression, a single layer MLP is Bayes consistent, but provided $2^d$ hidden units.

▶ Many adaptations of neural networks to missing values, often involving imputing by 0 and concatenating with the mask, but no underlying theory.

What architecture should we use to handle missing values? How complex should it be? What would be a good architecture design?
The NeuMiss network

For the case of linear regression under various missing data mechanisms:

- We propose a **theoretically grounded neural network architecture**, designed to approximate the Bayes predictor.

- The complexity of the architecture stays small thanks to the **sharing of parameters across missing data patterns**.

- Its originality and strength comes from the use of a **new type of non-linearity**: the multiplication by the missingness indicator.

- It is **robust to the missing data mechanism**, including difficult MNAR settings such as self-masking.
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.

$\text{obs}(M)$ (resp. $\text{mis}(M)$) are the indices of the observed (resp. missing) entries.

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

**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**

- $x = (1.1, 2.3, 3.1, 8, 5.27)$
- $\tilde{x} = (1.1, \text{NA}, -3.1, 8, \text{NA})$
- $m = (0, 1, 0, 0, 1)$
- $x_{\text{obs}(m)} = (1.1, 3.1, 8)$,
  $x_{\text{mis}(m)} = (2.3, 5.27)$

**Bayes predictor:**

$$f^* \in \arg\min_{f: (\mathbb{R} \cup \{\text{NA}\})^d \rightarrow \mathbb{R}} \mathbb{E} \left[ (Y - f(\tilde{X}))^2 \right]$$
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:**

$$f^*(X_{obs}, M) = \mathbb{E}[Y|X_{obs}(M), M]$$

$$= \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mathbb{E}[X_{mis}|X_{obs}, M] \rangle$$

$$= \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mathbb{E}[X_{mis}|X_{obs}] \rangle$$
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 \( \forall k \in \llbracket 1, d \rrbracket \),

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

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

**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:

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

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

\[ f^*(X_{\text{obs}}, M) = \beta_0^* + \langle \beta_{\text{obs}}^*, X_{\text{obs}} \rangle + \langle \beta_{\text{mis}}^*, \mu_{\text{mis}} + \Sigma_{\text{mis,obs}}(\Sigma_{\text{obs}})^{-1}(X_{\text{obs}} - \mu_{\text{obs}}) \rangle \]
How to approximate the Bayes predictors?

M(C)AR Bayes predictor:

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

- 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:

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

Expectation-Maximisation

MLP

\[ O(d^2) \] parameters

No robustness to the missing data mech.

High computational complexity!!!
(untractable when \( d \) reaches a few dozens)

\[ O(2^d) \] parameters

Largely over-parametrized.
How to approximate the Bayes predictors?

M(C)AR Bayes predictor:

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

<table>
<thead>
<tr>
<th>nb of parameters</th>
<th>Expectation-Maximisation</th>
<th>NeuMiss networks</th>
<th>MLP</th>
</tr>
</thead>
<tbody>
<tr>
<td>(O(d^2)) parameters</td>
<td>No robustness to the missing data mech.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(O(d^2)) parameters</td>
<td>Sharing parameters across missing data patterns</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(O(2^d)) parameters</td>
<td>Largely over-parametrized.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(O(d^2)) computational complexity.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

High computational complexity!!! (untractable when \(d\) reaches a few dozens)
Differentiable approximations of the inverse covariances

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

$$S_{\text{obs}(m)}^{(\ell)} = (\text{Id}_{\text{obs}(m)} - \frac{1}{L} \Sigma_{\text{obs}(m)}) S_{\text{obs}(m)}^{(\ell-1)} + \frac{1}{L} \text{Id}.$$ 

where $L \in \mathbb{R}^+$ is greater than the largest eigenvalue of $\Sigma_{\text{obs}(m)}$.

- The iterates converge linearly to $(\Sigma_{\text{obs}(m)})^{-1}$.

- Note: the iterates can be expressed as a series, corresponding to a Neumann series if $S^{(0)} = \text{Id}$ and $\ell = \infty$, i.e,

$$\left(\Sigma_{\text{obs}(m)}\right)^{-1} = \frac{1}{L} \sum_{k=0}^{\infty} \left(\text{Id}_{\text{obs}(m)} - \frac{1}{L} \Sigma_{\text{obs}(m)}\right)^k$$
Define the order-$\ell$ approximation of the Bayes predictor in M(C)AR settings

$$f^*_\ell(X_{\text{obs}}, M) = \langle \beta^*_{\text{obs}}, X_{\text{obs}} \rangle + \langle \beta^*_{\text{mis}}, \mu_{\text{mis}} + \Sigma_{\text{mis}, \text{obs}} S^{(\ell)}_{\text{obs}(m)} (X_{\text{obs}} - \mu_{\text{obs}}) \rangle.$$ 

**Proposition (Risk of the order-$\ell$ approximation)**

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

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

where $\nu$ is the smallest eigenvalue of $\Sigma$. 

Marine Le Morvan

NeuMiss networks

Montpellier - December 2020 15 / 25
NeuMiss network architecture

- M(C)AR Bayes predictor:

\[ f_\ell^*(X_{obs}, M) \approx \beta_0^* + \langle \beta_{obs}^*, X_{obs} \rangle + \langle \beta_{mis}^*, \mu_{mis} + \Sigma_{mis,obs} S^{(\ell)}_{obs}(X_{obs} - \mu_{obs}) \rangle \]

- Approximation of \((\Sigma_{obs})^{-1}\): \(S_{obs}(\ell) = (Id_{obs}(m) - \frac{1}{L} \Sigma_{obs}(m))S_{obs}(\ell-1) + \frac{1}{L} Id\).

- NeuMiss network architecture (illustrated with a depth of 4):

\[ \begin{align*}
& x \odot \bar{m} \\
& \mu \odot \bar{m} \\
& S^{(0)}_{Neu}(Id - \Sigma_{obs}) \\
& W^{(1)}_{Neu}(Id - \Sigma_{obs}) \\
& \oplus \oplus \oplus \oplus \\
& W^{(2)}_{Neu}(Id - \Sigma_{obs}) \\
& \oplus \oplus \oplus \oplus \\
& W_{Mix}(\Sigma_{mis,obs}) \\
& \oplus \oplus \oplus \oplus \\
& W_{\beta} \beta \\
& \rightarrow Y
\end{align*} \]

**Figure:** \(\bar{m} = 1 - m\). Each weight matrix \(W^{(k)}_{Neu}\) corresponds to a simple transformation of the covariance matrix indicated in blue.
NeuMiss network and Gaussian self-masking (MNAR)

- **M(C)AR Bayes predictor:**
  \[
  f^*(X_{obs}, M) = \beta^*_0 + \langle \beta^*_0, X_{obs} \rangle + \langle \beta^*_m, \mu_m + \Sigma_{mis,obs}(\Sigma_{obs})^{-1}(X_{obs} - \mu_{obs}) \rangle
  \]

- 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:
  \[
  f^*(X_{obs}, M) \approx \beta^*_0 + \langle \beta^*_0, X_{obs} \rangle + \langle \beta^*_m, (Id_{mis} + \hat{D}_{mis})^{-1}(\tilde{\mu}_m + \hat{D}_{mis}\mu_{mis}) \rangle + (Id_{mis} + \hat{D}_{mis})^{-1}\hat{D}_{mis}\Sigma_{mis,obs}(\Sigma_{obs})^{-1}(X_{obs} - \mu_{obs}) \rangle
  \]

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

**NeuMiss depth-1 layer**

\[ H_{\odot m} : \mathbb{R}^d \mapsto \mathbb{R}^d \]

**Feedforward layer (d hidden units)**

\[ H_{ReLU} : \mathbb{R}^d \times \{0, 1\} \mapsto \mathbb{R}^d \]

---

**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 \( H_{ReLU} \) 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, $n = 10^5$, $d = 20$).
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)

<table>
<thead>
<tr>
<th>d=10</th>
<th>EM</th>
<th>MICE + LR</th>
<th>MLP</th>
<th>NeuMiss</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=20000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
<tr>
<td>n=100000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.02</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d=20</th>
<th>EM</th>
<th>MICE + LR</th>
<th>MLP</th>
<th>NeuMiss</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=20000</td>
<td>-0.1</td>
<td>-0.02</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
<tr>
<td>n=100000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d=50</th>
<th>EM</th>
<th>MICE + LR</th>
<th>MLP</th>
<th>NeuMiss</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=20000</td>
<td>-0.2</td>
<td>-0.1</td>
<td>-0.1</td>
<td>-0.05</td>
</tr>
<tr>
<td>n=100000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
</tbody>
</table>

Probit self-masking (MNAR)

<table>
<thead>
<tr>
<th>d=10</th>
<th>EM</th>
<th>MICE + LR</th>
<th>MLP</th>
<th>NeuMiss</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=20000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
<tr>
<td>n=100000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d=20</th>
<th>EM</th>
<th>MICE + LR</th>
<th>MLP</th>
<th>NeuMiss</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=20000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
<tr>
<td>n=100000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>d=50</th>
<th>EM</th>
<th>MICE + LR</th>
<th>MLP</th>
<th>NeuMiss</th>
</tr>
</thead>
<tbody>
<tr>
<td>n=20000</td>
<td>-0.2</td>
<td>-0.1</td>
<td>-0.1</td>
<td>-0.05</td>
</tr>
<tr>
<td>n=100000</td>
<td>-0.1</td>
<td>-0.01</td>
<td>-0.1</td>
<td>-0.01</td>
</tr>
</tbody>
</table>

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.

\[ \sum m - \mu \circ \bar{m} \]

\[ W^{(1)}_{\text{Neu}} (Id - \Sigma_{\text{obs}}) \]

\[ W^{(2)}_{\text{Neu}} (Id - \Sigma_{\text{obs}}) \]

\[ W_{\text{Mix}} (\Sigma_{\text{mis,obs}}) \]

\[ W_{\beta} \]

\[ Y \]

Neumann iterations

Non-linearity

Thank you for your attention!