Missing Data Imputation using Optimal Transport

Boris Muzellec Julie Josse Claire Boyer Marco Cuturi

ENSAE IP PARIS · Inria · École Polytechnique IP PARIS · Sorbonne Université · Google AI

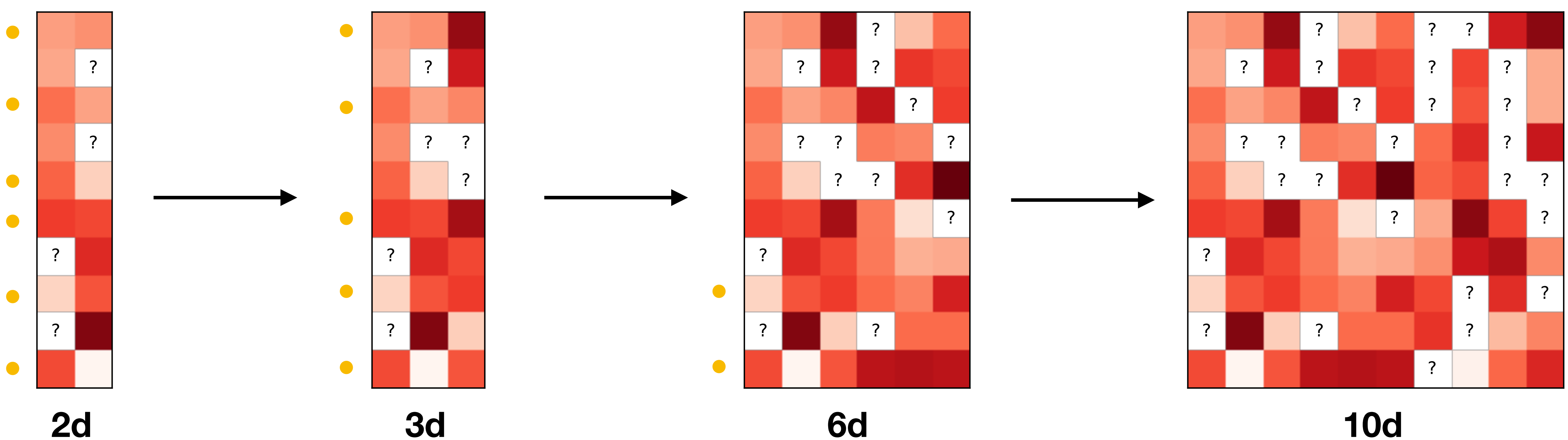
The missing data issue

- **Big data is plagued with missing values**

- **What to do?**

Option 1: Remove entries with missing values $\Longrightarrow$ **information loss, not sustainable**

Example with 25% missing rate:



With 1% missing rate: **5d: 95% rows kept** ⟶ **300d: 5% rows kept**

Option 2: Impute with reasonable guesses

Outline

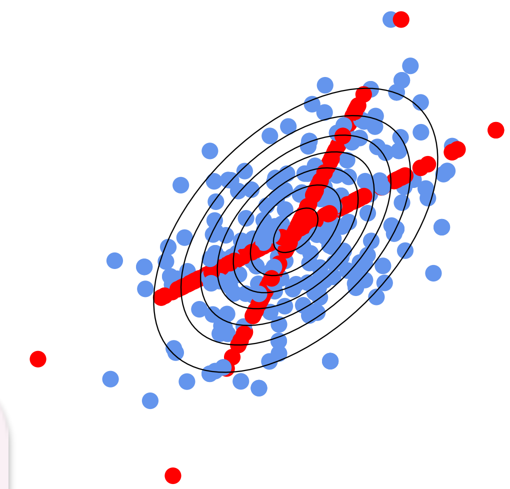
1. **Missing data and Optimal Transport**

2. **Non-parametric imputation with OT**

3. **Fitting parametric imputation models with OT**

How to impute?

- Mean imputation
- Regression (conditional expectation)



⚠ Deforms joint and marginal distributions

✔ Preserves distributions

- Using a conditional model:
  - With logistic, multinomial, Poisson regressions: R's *mice* (Van Buuren, 2011)
- Assuming a joint model:
  - EM + Gaussian distribution: *Amelia* (Honacker et al., 2011)
  - Low-rank models: *Softimpute* (Mazumder et al., 2010)
  - VAE and GAN: *MIWAE* (Mattei & Frellsen, 2019), *GAIN* (Yoon et al., 2018)
  - ...

This work: Preserves distributions

Parametric assumption not necessary

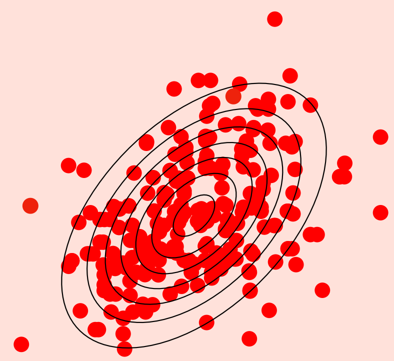
Imputing to preserve batch distributions

- **Contribution:** $\text{distribution}(\mathbf{X}^{(imp)}) \simeq \text{distribution}(\mathbf{X}^{(obs)})$

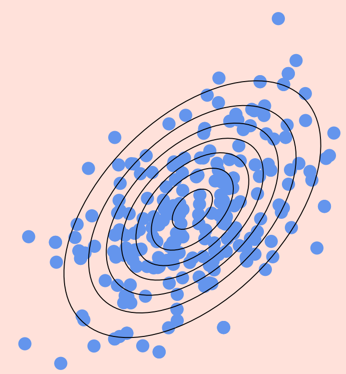
  Parametric assumption not necessary

Two batches from the same dataset should have similar distributions.
Measure this with a divergence:

$$D\left(X^{(imp)} \;\middle\|\; X^{(obs)} \right) \simeq 0$$



$X^{(imp)}$



$X^{(obs)}$

- **What divergence should we use ?**

Wishlist

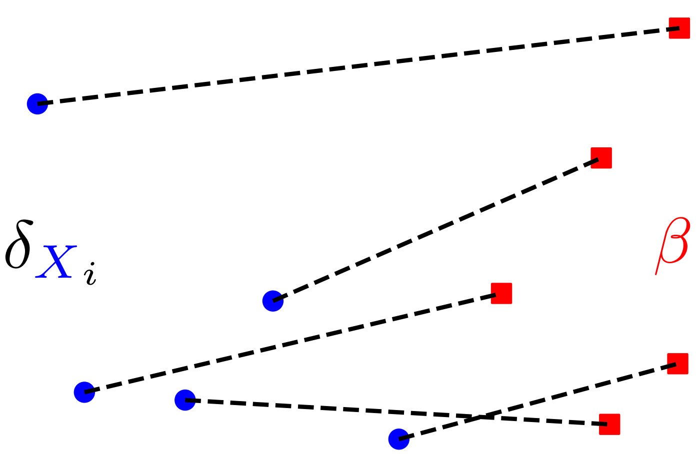
- **Handles disjoint supports**
- **Differentiable**
- **Affordable computing times**

Optimal Transport

- **Find the most efficient way of transporting distributions, according to a ground cost**
- **Defines a distance for probability distributions**

$$\mathbf{OT}(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\substack{\mathbf{P} \in \mathbb{R}_+^{n \times m} \\ \mathbf{P}\mathbb{1} = \mathbb{1}/n,\ \mathbf{P}^T \mathbb{1} = \mathbb{1}/m}} \langle \mathbf{P}, \mathbf{M}_{XY} \rangle$$

$$\alpha = \frac{1}{n} \sum_{i=1}^{n} \delta_{X_i}$$



$$\beta = \frac{1}{m} \sum_{j=1}^{m} \delta_{Y_j}$$

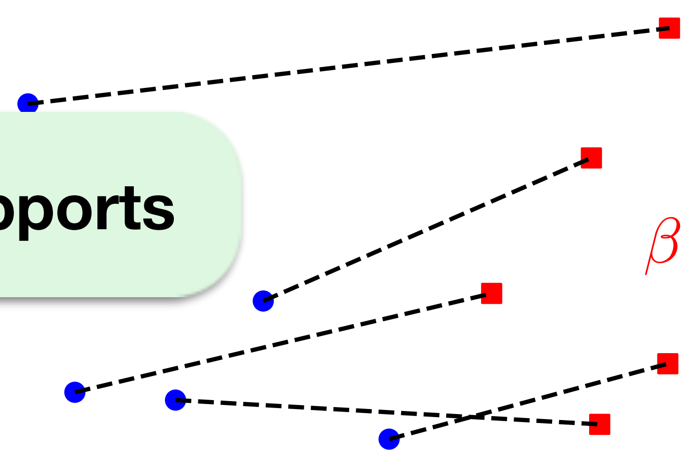
$$\mathbf{M}_{XY} = [\|X_i - Y_j\|^2]_{ij} \in \mathbb{R}^{n \times m}$$

Optimal Transport

- **Find the most efficient way of transporting distributions, according to a ground cost**
- **Defines a distance for probability distributions**

$$\mathbf{OT}(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\substack{\mathbf{P} \in \mathbb{R}_+^{n \times m} \\ \mathbf{P}\mathbb{1} = \mathbb{1}/n,\ \mathbf{P}^T\mathbb{1} = \mathbb{1}/m}} \langle \mathbf{P}, \mathbf{M}_{XY} \rangle$$

✔ **Handles disjoint supports**



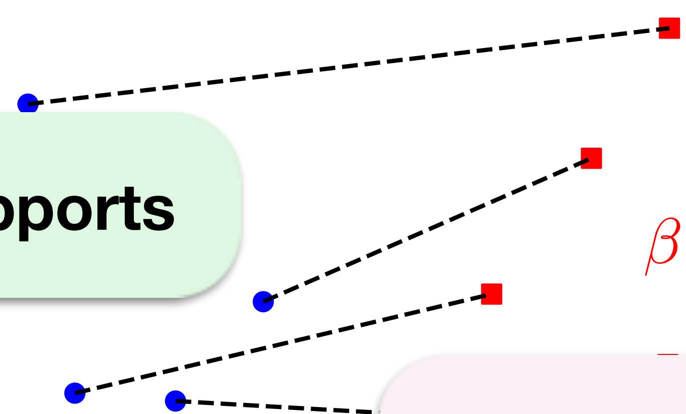
$$\beta = \frac{1}{m} \sum_{j=1}^{m} \delta_{Y_j}$$

$$\mathbf{M}_{XY} = [\|X_i - Y_j\|^2]_{ij} \in \mathbb{R}^{n \times m}$$

Optimal Transport

- **Find the most efficient way of transporting distributions, according to a ground cost**
- **Defines a distance for probability distributions**

$$\mathbf{OT}(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\substack{\mathbf{P} \in \mathbb{R}_+^{n \times m} \\ \mathbf{P}\mathbb{1} = \mathbb{1}/n,\ \mathbf{P}^T\mathbb{1} = \mathbb{1}/m}} \langle \mathbf{P}, \mathbf{M}_{XY} \rangle$$



$$\beta = \frac{1}{m} \sum_{j=1}^{m} \delta_{Y_j}$$

$$\mathbf{M}_{XY} = [\|X_i -$$

✔ **Handles disjoint supports**

Costly: $O(n^3 \log n)$

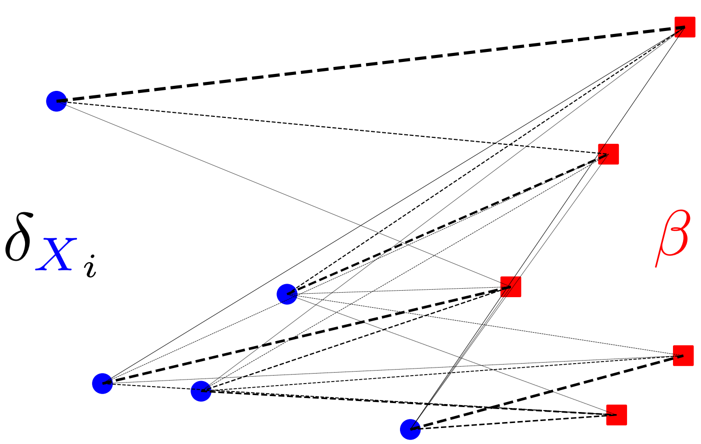
Not differentiable

Regularized Optimal Transport

$$\mathbf{OT}_\varepsilon(\alpha,\beta) \stackrel{\text{def}}{=} \min_{\substack{\mathbf{P}\in\mathbb{R}_+^{n\times m}\\ \mathbf{P}\mathbb{1}=\mathbb{1}/n,\ \mathbf{P}^T\mathbb{1}=\mathbb{1}/m}} \langle \mathbf{P}, \mathbf{M}_{XY}\rangle + \varepsilon \sum_{ij} p_{ij}\log p_{ij}$$

(Cuturi, 2013)

$$\alpha = \frac{1}{n}\sum_{i=1}^{n} \delta_{X_i}$$



$$\beta = \frac{1}{m}\sum_{j=1}^{m} \delta_{Y_j}$$

$$\mathbf{M}_{XY} = [\|X_i - Y_j\|^2]_{ij} \in \mathbb{R}^{n\times m}$$

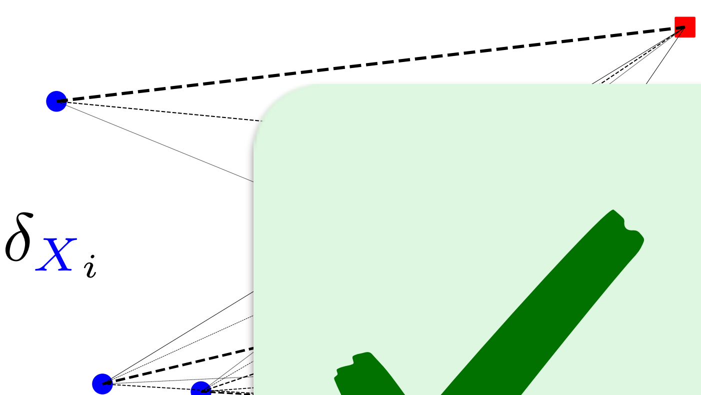
- **Sinkhorn divergence:**

$$S_\varepsilon(\alpha,\beta) \stackrel{\text{def}}{=} \mathbf{OT}_\varepsilon(\alpha,\beta) - \frac{1}{2}(\mathbf{OT}_\varepsilon(\alpha,\alpha) + \mathbf{OT}_\varepsilon(\beta,\beta))$$

Regularized Optimal Transport

$$\mathbf{OT}_\varepsilon(\alpha, \beta) \stackrel{\text{def}}{=} \min_{\substack{\mathbf{P} \in \mathbb{R}_+^{n \times m} \\ \mathbf{P}\mathbb{1} = \mathbb{1}/n,\ \mathbf{P}^T\mathbb{1} = \mathbb{1}/m}} \langle \mathbf{P}, \mathbf{M}_{XY} \rangle + \varepsilon \sum_{ij} p_{ij} \log p_{ij}$$

(Cuturi, 2013)



$$\alpha = \frac{1}{n} \sum_{i=1}^{n} \delta_{X_i}$$

$$\mathbf{M}_{XY} = [\|X_i \quad Y_j\| \quad]_{ij} \in \mathbb{R}$$

Handles disjoint supports

Differentiable

Fast computation with Sinkhorn's algorithm

- **Sinkhorn divergence:**

$$S_\varepsilon(\alpha, \beta) \stackrel{\text{def}}{=} \mathbf{OT}_\varepsilon(\alpha, \beta) - \frac{1}{2}(\mathbf{OT}_\varepsilon(\alpha, \alpha) + \mathbf{OT}_\varepsilon(\beta, \beta))$$

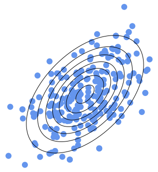
Imputation Algorithm

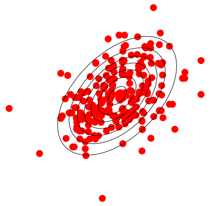
- **Input:** $\mathbf{X} = (1 - \mathbf{M}) \odot \mathbf{X}^{(obs)} + \mathbf{M} \odot \mathrm{NA}\,, \quad \mathbf{M} \in \{0,1\}^{n \times d}$

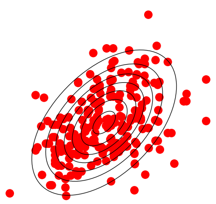
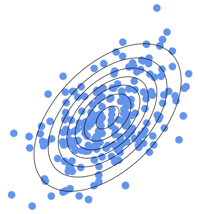
$(m_{ij} = 1 \iff x_{ij} \text{ missing})$

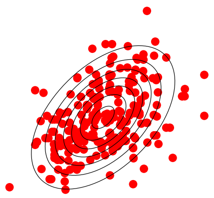
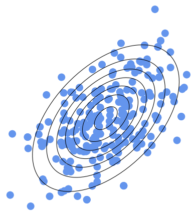
- **Initial imputations:** $x_{ij}^{(imp)} = \overline{x_{:j}^{(obs)}} + \varepsilon \text{ if } m_{ij} = 1$ **(column mean of observed values + noise)**

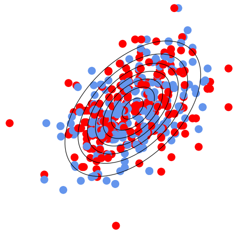
- for $t = 1, 2 \ldots, T$

  Sample batch with no missing values:  $\mathbf{X}_{i_1,\ldots,i_K}^{(obs)}$

  Sample batch with missing values:  $\mathbf{X}_{j_1,\ldots,j_K}^{(imp)}$

  Compute Sinkhorn batch loss: $S_\varepsilon\Big($  $\|$  $\Big)$

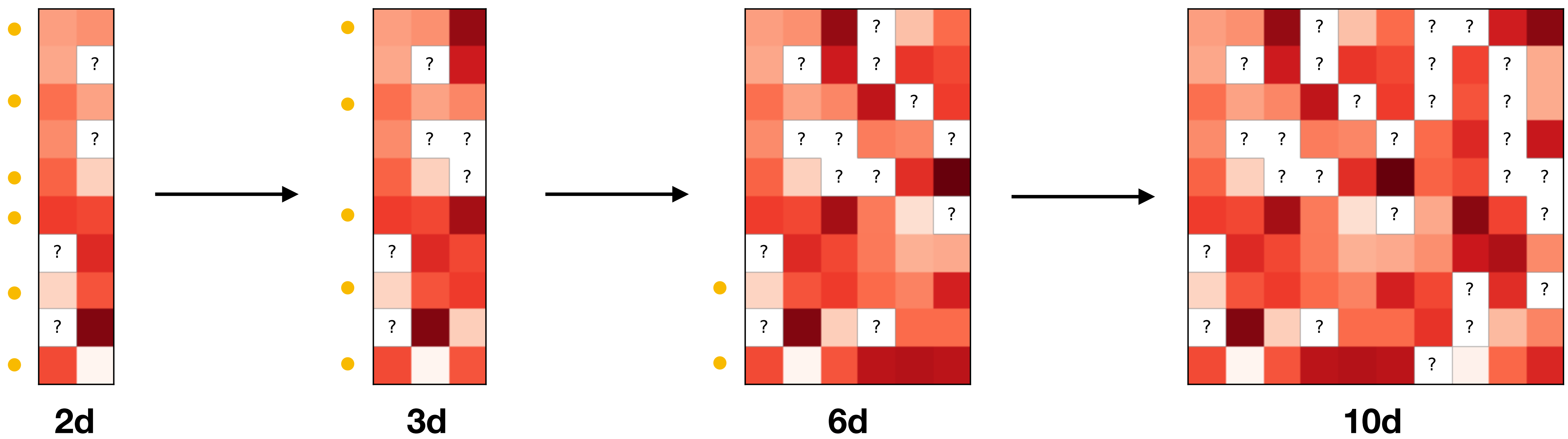
  Update imputations: $\mathbf{X}^{(imp)} \leftarrow \mathbf{X}^{(imp)} - \eta \nabla_{\mathbf{X}^{(imp)}} S_\varepsilon\Big($  $\|$  $\Big)$

- **Output:** $\hat{\mathbf{X}} = (1 - \mathbf{M}) \odot \mathbf{X}^{(obs)} + \mathbf{M} \odot \mathbf{X}^{(imp)}$ 

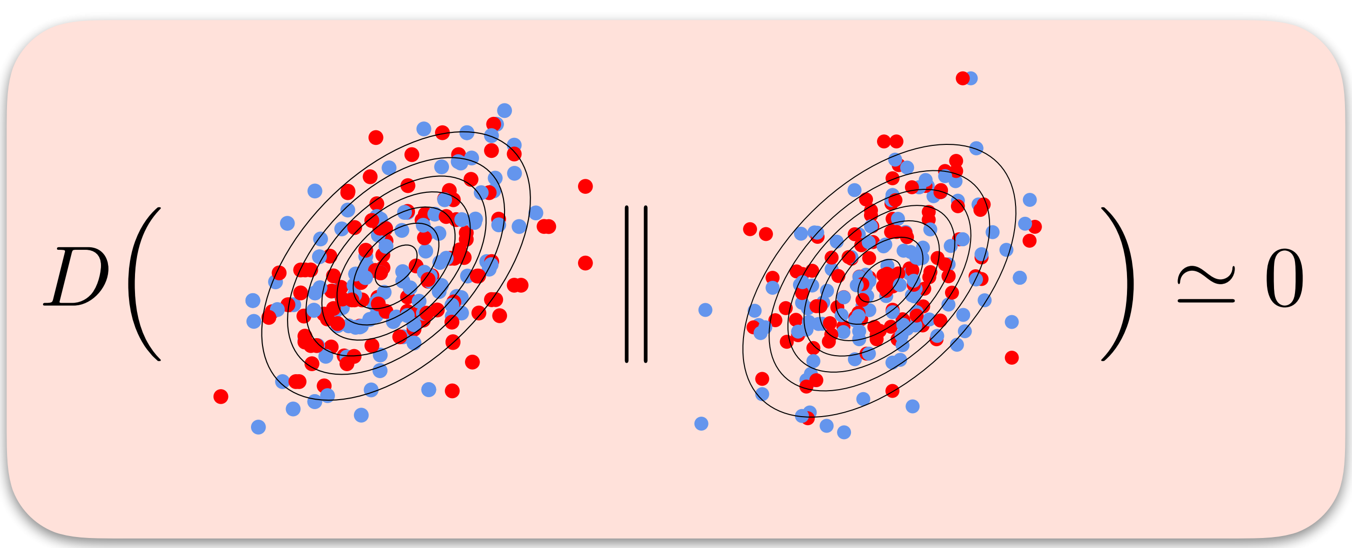
Observed values vs dimension

- **Problem: as dimension increases, almost all entries have NAs.**

Example with 25% missing rate:



- **But 2 sampled batches should still have similar distributions.**



Imputation Algorithm

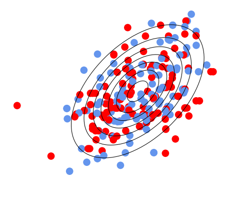
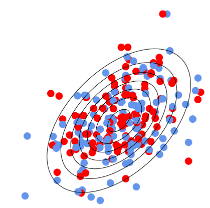
- **Input:** $\mathbf{X} = (1 - \mathbf{M}) \odot \mathbf{X}^{(obs)} + \mathbf{M} \odot \mathrm{NA}\,, \quad \mathbf{M} \in \{0,1\}^{n \times d}$

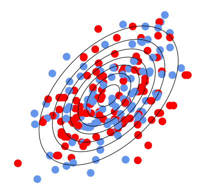
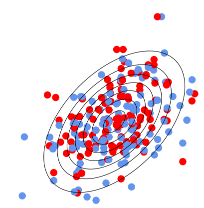
$(m_{ij} = 1 \iff x_{ij} \text{ missing})$

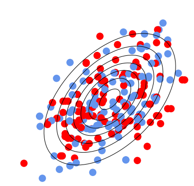
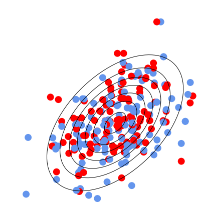
- **Initial imputations:** $x_{ij}^{(imp)} = \overline{x_{:j}^{(obs)}} + \varepsilon \text{ if } m_{ij} = 1$ **(column mean of observed values + noise)**

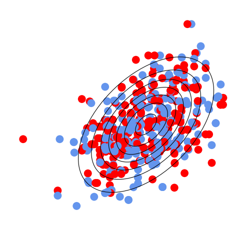
- for $t = 1, 2 \ldots, T$

  Mix observations and imputations: $\hat{\mathbf{X}} \leftarrow (1 - \mathbf{M}) \odot \mathbf{X}^{(obs)} + \mathbf{M} \odot \mathbf{X}^{(imp)}$

  Sample 2 batches:  $\hat{\mathbf{X}}_{i_1, \ldots, i_K}$ **and**  $\hat{\mathbf{X}}_{j_1, \ldots, j_K}$

  Compute Sinkhorn batch loss: $S_\varepsilon\Big($  $\|$  $\Big)$

  Update imputations: $\mathbf{X}^{(imp)} \leftarrow \mathbf{X}^{(imp)} - \eta \nabla_{\mathbf{X}^{(imp)}} S_\varepsilon\Big($  $\|$  $\Big)$

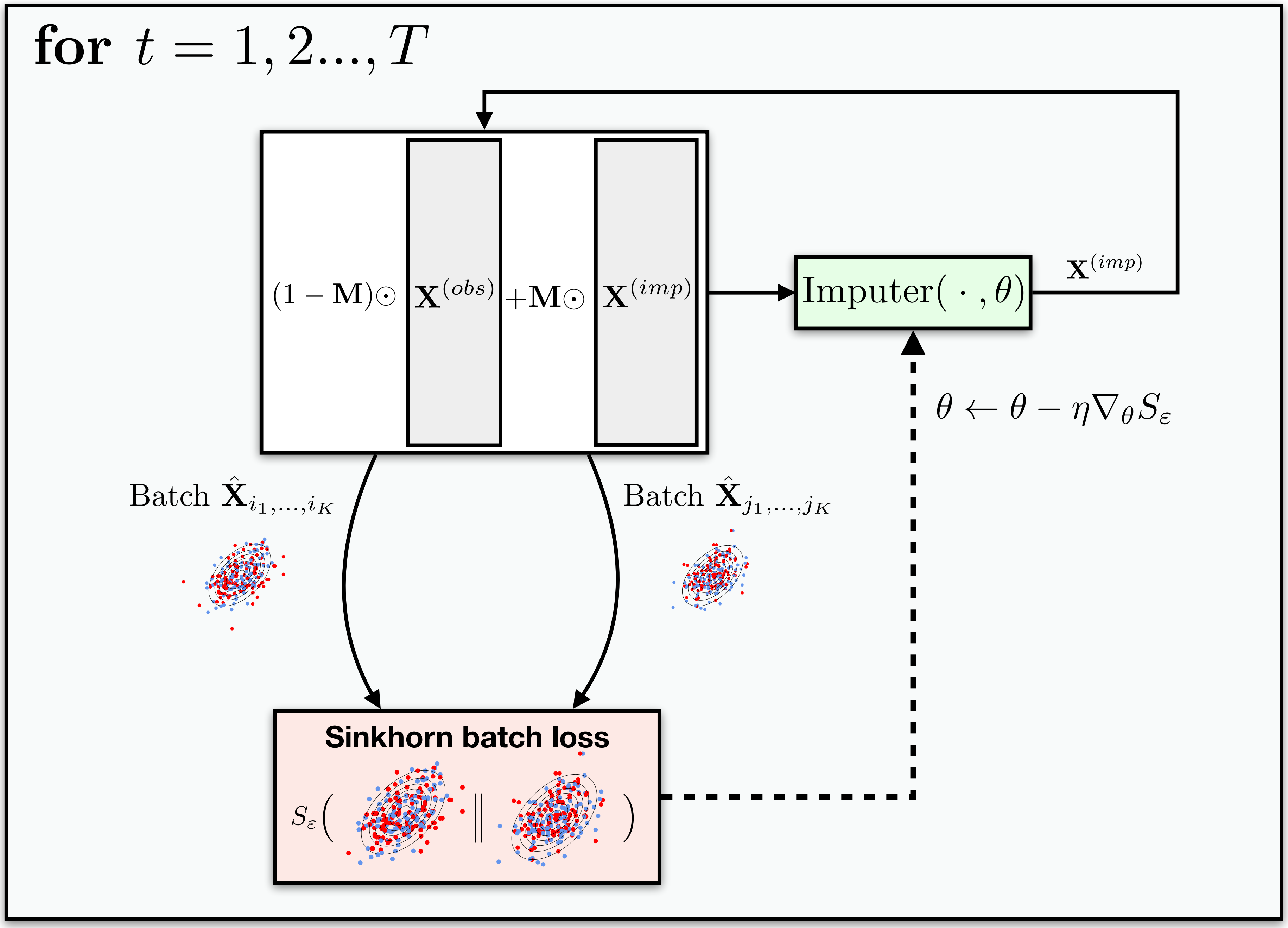
- **Output:** $\hat{\mathbf{X}} = (1 - \mathbf{M}) \odot \mathbf{X}^{(obs)} + \mathbf{M} \odot \mathbf{X}^{(imp)}$ 

What if we want a parametric model?

OT as an imputation criterion

- **We used OT to directly fit imputation values by gradient descent.**

💡 **We could use it to fit *any* parametric imputation model.** **e.g. linear model, MLP, …**

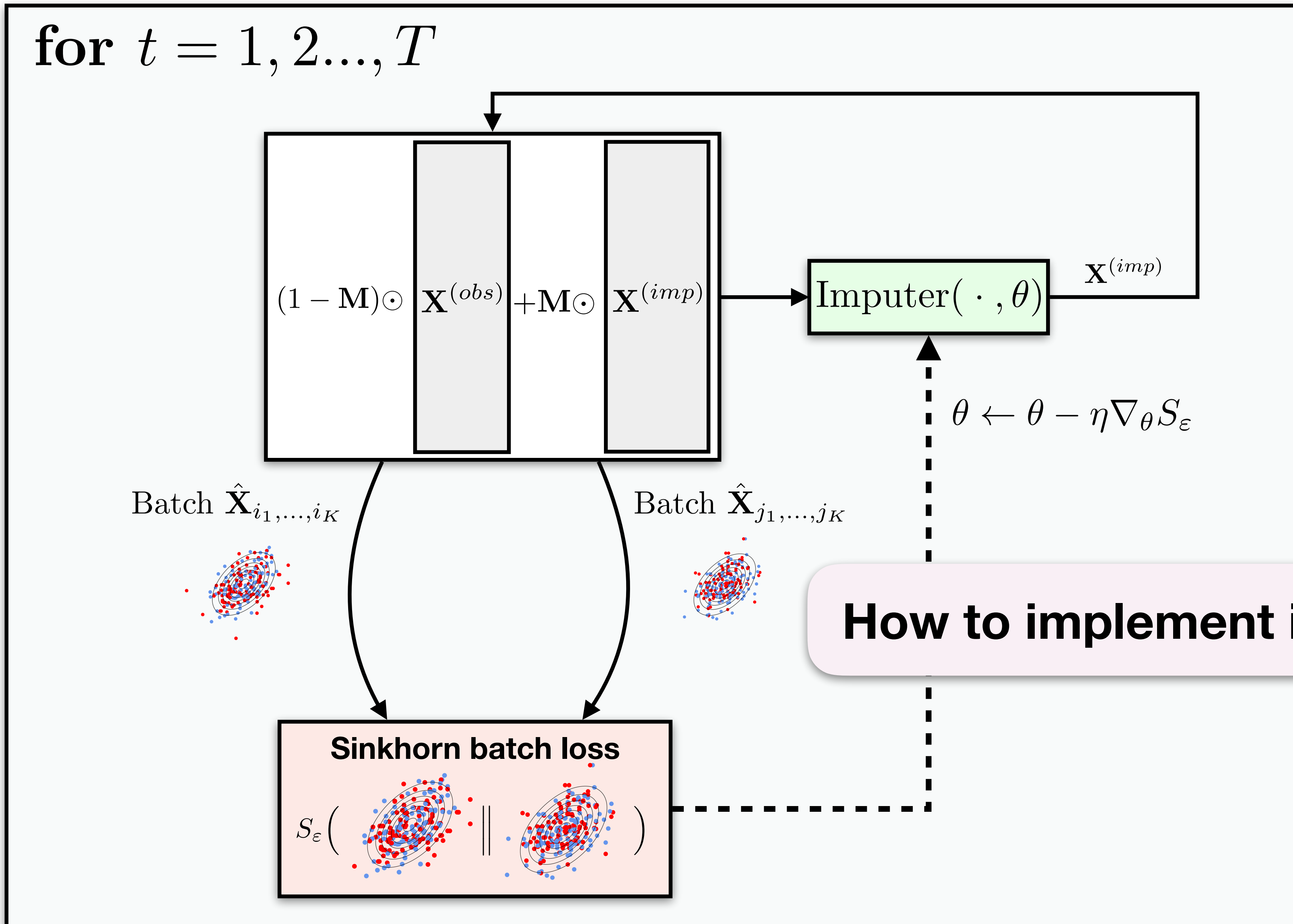


Allows out-of-sample imputation

OT as an imputation criterion

- **We used OT to directly fit imputation values by gradient descent.**

We could use it to fit *any* parametric imputation model. **e.g. linear model, MLP, …**



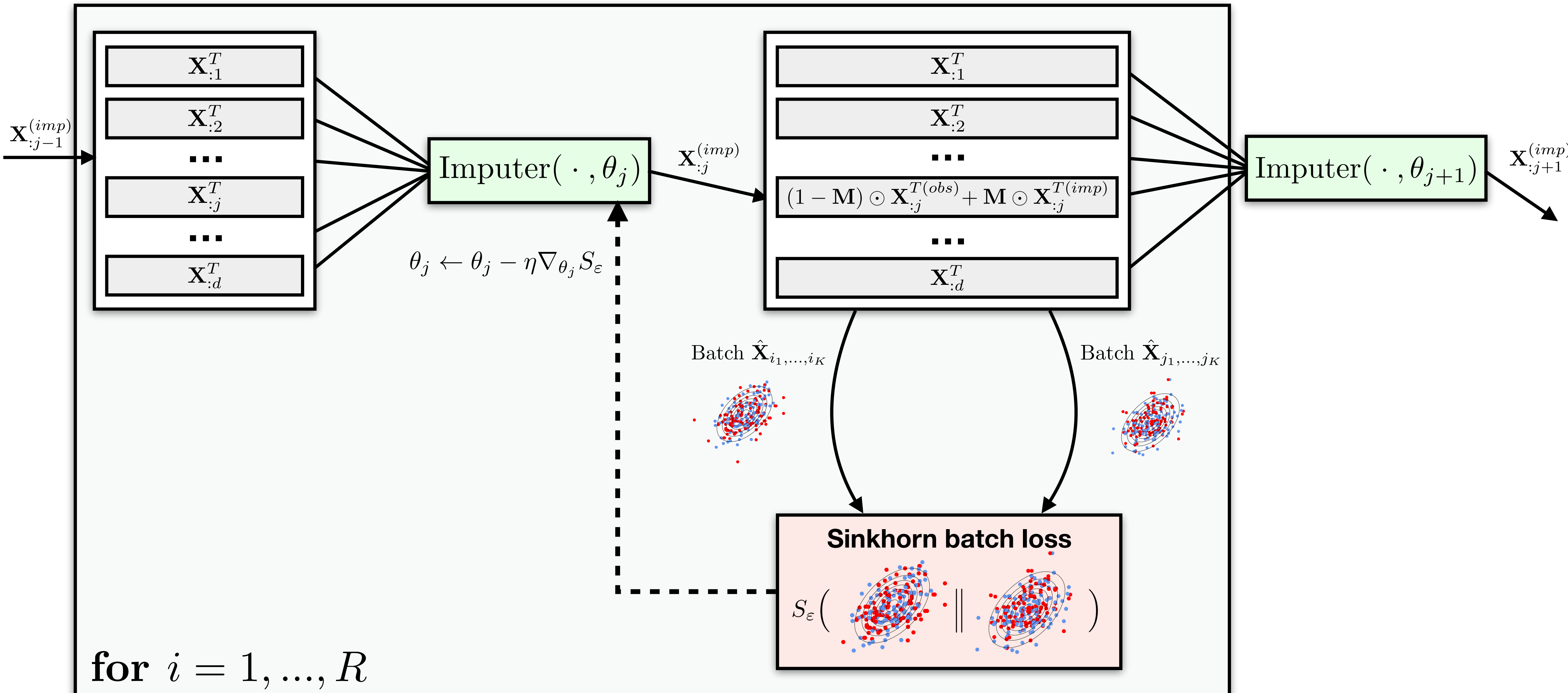
How to implement in practice?

Allows out-of-sample imputation

Round-robin imputation

Impute variables one by one, using all other variables as inputs

Use d **parametric models** $\theta_1, \theta_2, \ldots, \theta_d$ **(one for each variable)**



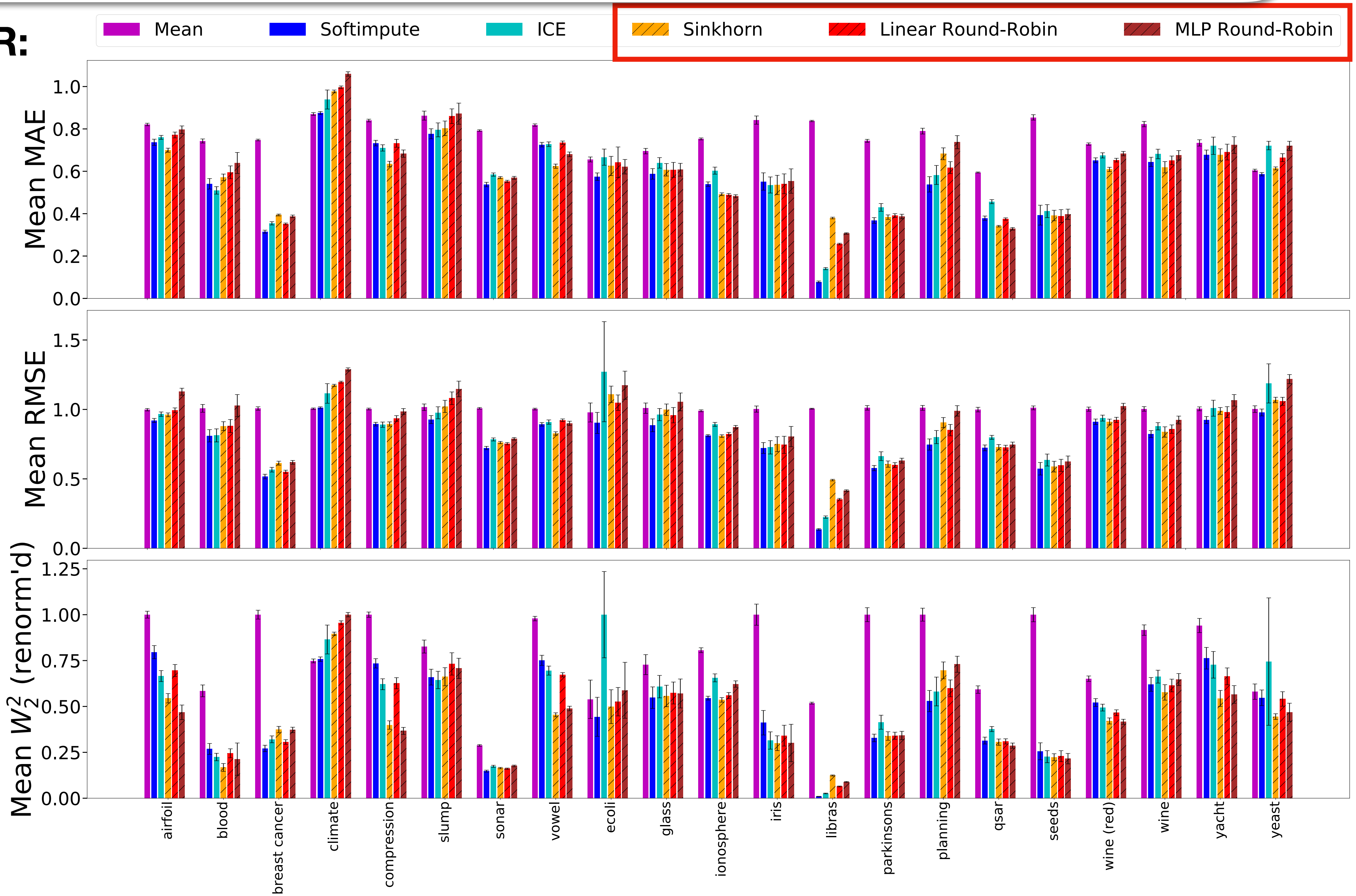
Generalization of Imputation by Chained Equations (e.g. R's *mice*)

Comparison with baselines

Extensive experiments on UCI datasets in MCAR, MAR and MNAR settings.

Three performance metrics:
- **Mean Absolute Error (MAE)**
- **Root Mean Square Error (RMSE)**
- **Optimal Transport (W_2^2)**

50% MCAR:



Comparison with Deep Learning

MIWAE **(Mattei & Frellsen, 2019),** ***GAIN*** **(Yoon et al., 2018),** ***VAEs*** **(Ivanov et al., 2019)**

30% MNAR:
(masked quantiles)

github.com/BorisMuzellec/MissingDataOT