Random Forests and Autoencoders with Missing Values

Irving Gómez Méndez
Advisor: Emilien Joly

June, 2021

CIMAT

Outline of the Talk

1. Introduction, Motivation and Goals
2. Data-Missing Mechanisms
3. Random Forests
   - Introduction
   - Simulation Study of an RF Algorithm with Missing Entries
   - Consistency of an RF Algorithm with Missing Entries
4. Autoencoders with Missing Values
   - Autoencoders
   - Denoising Autoencoders
   - Variational Inference
   - Variational Autoencoders (VAEs)
   - VAEs with missing Data

With the progress in data generation, new techniques and algorithms from the field of **machine learning have been developed** as powerful tools **for the analysis of complex and large data**.

However, most of these techniques have been developed **considering that all the variables are available**. Although, in practice it **is common to deal with** data sets that have **missing values**.

The present work might be divided in two parts:

1. The **first part** is dedicated to the estimation of the regression function through **random forests** when there are missing entries.
2. The **second part** focuses on the reconstruction of the original observations through the use of **autoencoders**.

The firs part is divided in three sections for this talk.

- Introduction of the random forest algorithm, explaining the problems face by the original method when there are missing entries, and **introducing a new algorithm to deal with these problems**.
- Development of a **simulation study** comparing the proposed approach with other algorithms to handle missing values using random forests.
- **Prove of the consistency** of the proposal when the introduction of missing values is made completely at random and when the regression function can be expressed as an additive model.

The second part is divided in 4 sections for this talk.

- Introduction to autoencoders and denoising autoencoders to reconstruct missing data.
- Introduction to variational inference.
- Introduction to variational autoencoders.
- Use of variational autoencoders for missing data, **trained with a new loss function (EMMELBO)**.

Data Set with Missing Entries

Ideally the matrix of observations is complete, although in practice there can be blank spaces.

$\mathbf{X}^{(1)}$	$\mathbf{X}^{(2)}$	$\mathbf{X}^{(3)}$	$\mathbf{X}^{(4)}$	Y
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩

$\mathbf{X}^{(1)}$	$\mathbf{X}^{(2)}$	$\mathbf{X}^{(3)}$	$\mathbf{X}^{(4)}$	Y
	🟩		🟩	🟩
🟩		🟩		🟩
		🟩		🟩
🟩	🟩		🟩	🟩
	🟩			🟩

In its more generality, those blank spaces represent values for which we have no information, although some context like censorship are especial cases of missing information.

Mechanisms of Missingness[1]

- **Missing Completely at Random (MCAR)** A variable is missing completely at random if the probability of missingness **does not depend on** either, **the observed values non the missing values**.
- **Missing at Random (MAR)** A variable is missing at random if the probability of missingness depends on observed variables but **does not depend on missing values**.
- **Not Missing at Random (NMAR)** If the probability of missingness **depends on missing values**, it is called not missing at random.

[1] Donald B Rubin. "Inference and missing data". In: *Biometrika* 63.3 (1976), pp. 581–592.

Regression Random Forests

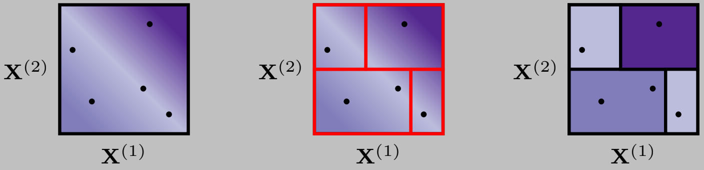
Regression Framework

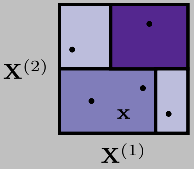
In our framework, we assume to have access to a training set $\mathcal{D}_n = (\mathbf{X}_i, Y_i)_{i=1,\ldots,n}$ where the response variables Y_i are real-valued and the input variables $\mathbf{X}_i$ belong to some space $\mathcal{X}$. **In most applications the space $\mathcal{X}$ is a compact** portion of a p dimensional space. **Hence, we assume that** $\mathcal{X} = [0,1]^p$. The aim is to estimate the regression function

$$m(\cdot) \equiv \mathbb{E}[Y|\mathbf{X} = \cdot]$$

Estimation of the Regression Function

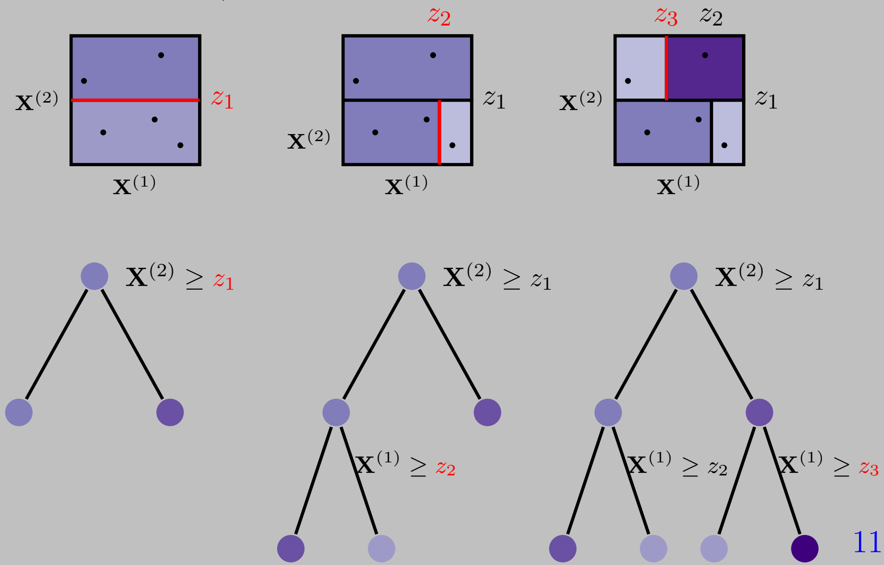
We can estimate m dividing (somehow) $\mathcal{X}$ in disjoint regions and predicting with a constant in each region.





$m_n(\mathbf{x}\,;\Theta) =$ ●

Recursive Trees / Creation of Disjoint Regions


z_2
z_3 z_2
$\mathbf{X}^{(2)}$
z_1
z_1
$\mathbf{X}^{(2)}$
z_1
$\mathbf{X}^{(2)}$
$\mathbf{X}^{(1)}$
$\mathbf{X}^{(1)}$
$\mathbf{X}^{(1)}$
$\mathbf{X}^{(2)} \geq z_1$
$\mathbf{X}^{(2)} \geq z_1$
$\mathbf{X}^{(2)} \geq z_1$
$\mathbf{X}^{(1)} \geq z_2$
$\mathbf{X}^{(1)} \geq z_2$
$\mathbf{X}^{(1)} \geq z_3$

CART-split Criterion

Let $d = (h, z)$ be a cut in direction h at position z,

$$\begin{aligned} L_n(A,d) =& \frac{1}{N(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_A\right)^2 \mathbb{1}_{\mathbf{X}_i \in A} \\ & - \frac{1}{N_n(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_{A_L} \mathbb{1}_{\mathbf{X}_i^{(h)} < z} - \bar{Y}_{A_R} \mathbb{1}_{\mathbf{X}_i^{(h)} \geq z}\right)^2 \mathbb{1}_{\mathbf{X}_i \in A} \\ & \vdots \\ =& \frac{N(A_L)N(A_R)}{N(A)N(A)} \left(\bar{Y}_{A_L} - \bar{Y}_{A_R}\right)^2 . \end{aligned}$$

The **CART-criterion** attempts to **minimize the variance inside the cells**, making the cells as distinct as possible (in terms of the value of Y) but **maintaining the balance in the number of points** in the cells.

Other Criteria

Other criteria to construct the trees minimizes different impurity measures:

- Shannon entropy (ID3 and C4.5)
- Missclassfication error
- Gini index
- Hypothesis test (conditional trees)

We are going to focus on the CART-criterion.

We construct the cells maximizing the CART criterion over all possible cuts in cell A,

$$\widehat{d} = (\widehat{h}, \widehat{z}) \in \arg\max_{d \in \mathcal{C}_A} L_n(A, d),$$

where $\mathcal{C}_A$ is the set of all possible cuts in node A.

Random Forest

A random forest is an ensemble of trees, i.e.

$$m_{M,n}(\mathbf{x};\Theta_1,\ldots,\Theta_M)=\frac{1}{M}\sum_{k=1}^{M}m_n(\mathbf{x};\Theta_k).$$

We expect the **predictor**:

- To be more stable
- To have less variance

Conditions:

- Independent or non-correlated trees

However the trees are not independent because they use the same data $\mathcal{D}_n$.

Solution:

- Introduce sources of randomness

Parameters

We add two sources of randomness in each tree:

1. We **select randomly** a_n (with or without replacement) **observations** prior to the construction of each tree.
2. We **select randomly `mtry` candidate directions** to perform the cut.

These are parameters of the random forest together with:

1. M, which is the **number of trees**. It is only restricted by computational power.
2. `nodesize`, which is the **maximum number of points in a final cell**.
3. We have added q_n, which is the **minimum number of points in a final cells**.

Decision Trees with Missing Values

Data Set with Missing Entries

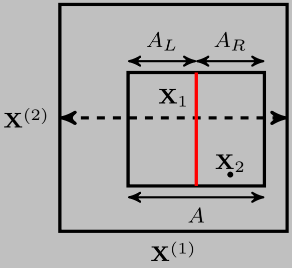
Ideally the matrix of observations is complete, although in practice there can be blank spaces.

$\mathbf{X}^{(1)}$	$\mathbf{X}^{(2)}$	$\mathbf{X}^{(3)}$	$\mathbf{X}^{(4)}$	Y
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩
🟩	🟩	🟩	🟩	🟩

$\mathbf{X}^{(1)}$	$\mathbf{X}^{(2)}$	$\mathbf{X}^{(3)}$	$\mathbf{X}^{(4)}$	Y
	🟩		🟩	🟩
🟩		🟩		🟩
		🟩		🟩
🟩	🟩		🟩	🟩
	🟩			🟩

We want to estimate m when **there are missing entries in the training data set** $\mathcal{D}_n$, we assume that there are **no missing values for the target** Y.

The CART-Criterion Cannot be Computed

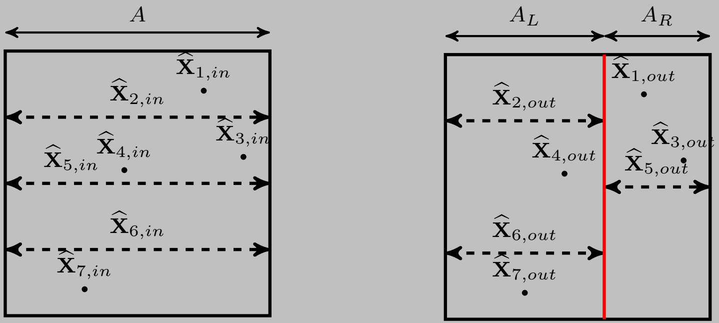


$$\begin{aligned} L_n(A,d) =& \frac{1}{N(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_A\right)^2 \mathbb{1}_{\mathbf{X}_i \in A} \\ &- \frac{1}{N(A)} \sum_{i=1}^{n} \left(Y_i - \bar{Y}_{A_L} \mathbb{1}_{\mathbf{X}_i^{(h)} < z} - \bar{Y}_{A_R} \mathbb{1}_{\mathbf{X}_i^{(h)} \geq z}\right)^2 \mathbb{1}_{\mathbf{X}_i \in A} \end{aligned}$$

The red parts in the CART criterion **cannot be computed**.

Assign the Observations

We can solve this problem **assigning the observations with missing values to left or right** given some cut that splits the cell.


A
$\widehat{\mathbf{X}}_{1,in}$
$\widehat{\mathbf{X}}_{2,in}$
$\widehat{\mathbf{X}}_{3,in}$
$\widehat{\mathbf{X}}_{4,in}$
$\widehat{\mathbf{X}}_{5,in}$
$\widehat{\mathbf{X}}_{6,in}$
$\widehat{\mathbf{X}}_{7,in}$
A_L
A_R
$\widehat{\mathbf{X}}_{1,out}$
$\widehat{\mathbf{X}}_{2,out}$
$\widehat{\mathbf{X}}_{3,out}$
$\widehat{\mathbf{X}}_{4,out}$
$\widehat{\mathbf{X}}_{5,out}$
$\widehat{\mathbf{X}}_{6,out}$
$\widehat{\mathbf{X}}_{7,out}$

Notation

Define the **indicator of missing value** as

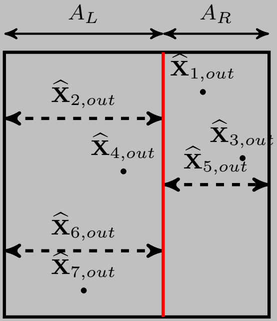
$$\mathbf{M}^{(h)} = \begin{cases} 1 & \text{if } \mathbf{X}^{(h)} \text{ is missing} \\ 0 & \text{otherwise} \end{cases}, \quad 1 \leq h \leq p.$$

Let be

- $\widehat{N}^{(h)}_{miss}(A)$ the number of observations assigned to cell A whose variable $\mathbf{X}^{(h)}$ is missing.
- $\mathcal{W}^{(h)}_{A} = \{0,1\}^{\widehat{N}^{(h)}_{miss}(A)}$ the collection of binary vectors w.
- $w_k = 1$ means that the observation $(\mathbf{X}_{j_k}, Y_{j_k})$ is assigned to the left.
- $w_k = 0$ means that the observation $(\mathbf{X}_{j_k}, Y_{j_k})$ is assigned to the right.

Example of Assignation

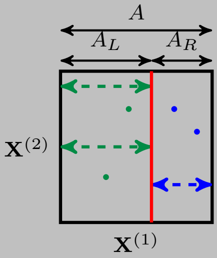
In this example we assign $\mathbf{X}_2$ to the left, $\mathbf{X}_5$ to the right and $\mathbf{X}_6$ to the left. So the assigantion is represented with the vector $w = (1, 0, 1)$.


A_L
A_R
$\widehat{\mathbf{X}}_{1,out}$
$\widehat{\mathbf{X}}_{2,out}$
$\widehat{\mathbf{X}}_{3,out}$
$\widehat{\mathbf{X}}_{4,out}$
$\widehat{\mathbf{X}}_{5,out}$
$\widehat{\mathbf{X}}_{6,out}$
$\widehat{\mathbf{X}}_{7,out}$

CART-Criterion with Assignation

With these assignations, we can compute the CART-criterion.

$$\begin{aligned} L_n(A,d,w) =& \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_A\right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A} \\ & - \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A_L}\right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, a^{(h)} \leq \widehat{\mathbf{X}}_{i,out}^{(h)} < z} \\ & - \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A_R}\right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, z \leq \widehat{\mathbf{X}}_{i,out}^{(h)} \leq b^{(h)}} \end{aligned}$$



Best Cut and Assignation

For a cell A and an input imputation vector $\widehat{\mathbf{X}}_{in}$, **the algorithm chooses a cut and assignation** $(\widehat{d}, \widehat{w})$ by maximizing $L_n(A, d, w)$ over $\mathcal{C}_A \times \mathcal{W}_A^{(h)}$,

$$(\widehat{d}, \widehat{w}) \in \underset{\substack{d \in \mathcal{C}_A \\ w \in \mathcal{W}_A^{(h)}}}{\arg\max}\, L_n(A, d, w)\,.$$

Finally, the **"imputed" intervals are updated**, symbolized by $\widehat{\mathbf{X}}_{i,in}^{\widehat{(h)}} \leftarrow \widehat{\mathbf{X}}_{i,out}^{\widehat{(h)}}$.

Different to previous techniques, this approach optimizes over the assignations.

Simulation

Simulation Framework

A simulation was conducted to inspect the behavior of the proposed approach, in a benchmark study. **Comparing** the proposal **against 6 different methods** to handle missing values through random forests.
We consider the **regression function** “friedman1”:

$$m(\mathbf{x}) = 10\sin\left(\pi\mathbf{x}^{(1)}\mathbf{x}^{(2)}\right) + 20\left(\mathbf{x}^{(3)} - 0.5\right)^2 + 10\mathbf{x}^{(4)} + 5\mathbf{x}^{(5)}$$

And **7 data-missing mechanisms**, similar to those introduced by Rieger, Hothorn, and Strobl.[2]

[2] Anna Rieger, Torsten Hothorn, and Carolin Strobl. “Random forests with missing values in the covariates”. In: *Technical report* (2010). URL: http://epub.ub.uni-muenchen.de/11481.

Data Sets

Training data sets

- We create 100 training data sets.
- We simulate 200 observations from $\mathbf{X} \sim \mathcal{U}[0,1]^5$.
- Missing values are introduced in $\mathbf{X}^{(1)}$, $\mathbf{X}^{(3)}$ and $\mathbf{X}^{(4)}$.

Testing data set

- We create 1 data set.
- We simulate 2000 observations from $\mathbf{X} \sim \mathcal{U}[0,1]^5$.
- All observations are complete.

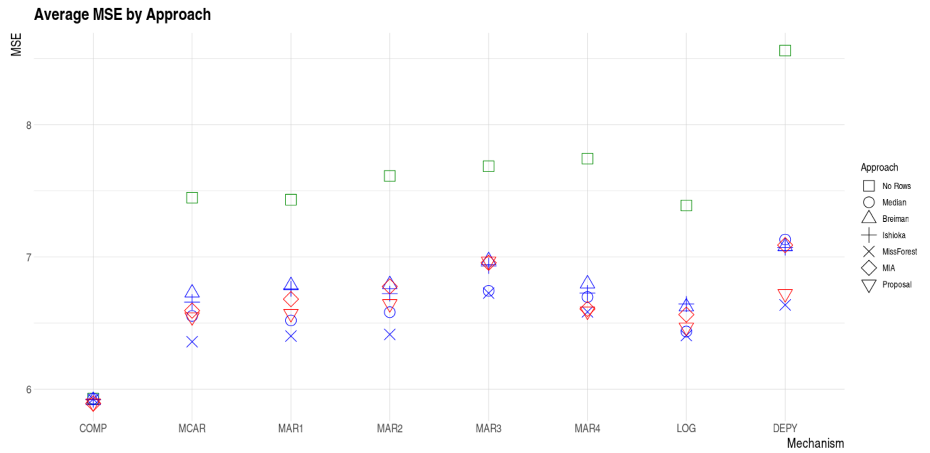
Random Forests

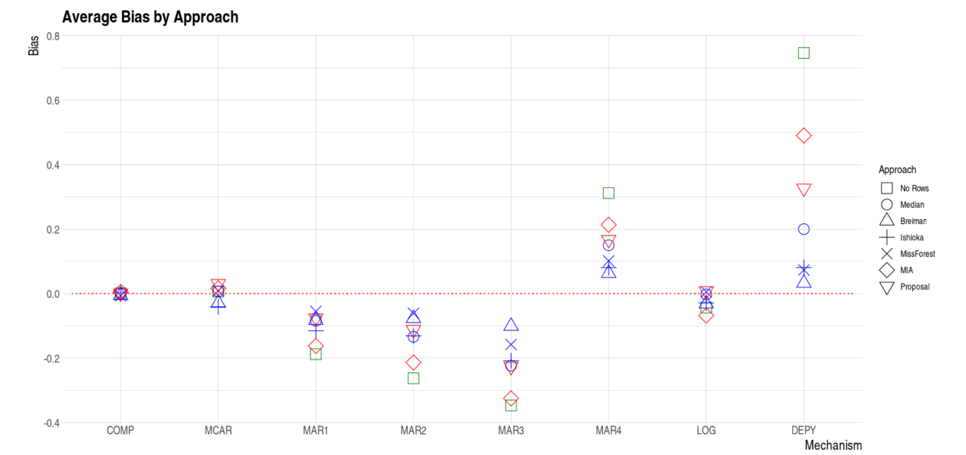
- For each training data set and each mechanism of missingness (including complete data) we create a random forest.
- Each forest is built with the parameters:
    - $M = 50$ trees.
    - `mtry` $= 1$ variable selected at random to perform the cut.
    - $a_n = 127$ observations selected at random and without replacement for each tree.
    - `nodesize` $= 5$, maximum number of observations in the final nodes.

Percentage of Missingness

Missing values are introduced in the training data sets in $\mathbf{X}^{(1)}$, $\mathbf{X}^{(3)}$ and $\mathbf{X}^{(4)}$ accordingly to the following percentages.

Determinant Variable	Missing Variable	% Missing Data
$\mathbf{X}^{(2)}$, Y	$\mathbf{X}^{(1)}$	20%
$\mathbf{X}^{(5)}$, Y	$\mathbf{X}^{(3)}$	10%
	$\mathbf{X}^{(4)}$	20%


Average MSE by Approach
MSE
8
7
6
COMP
MCAR
MAR1
MAR2
MAR3
MAR4
LOG
DEPY
Mechanism
Approach
No Rows
Median
Breiman
Ishioka
MissForest
MIA
Proposal

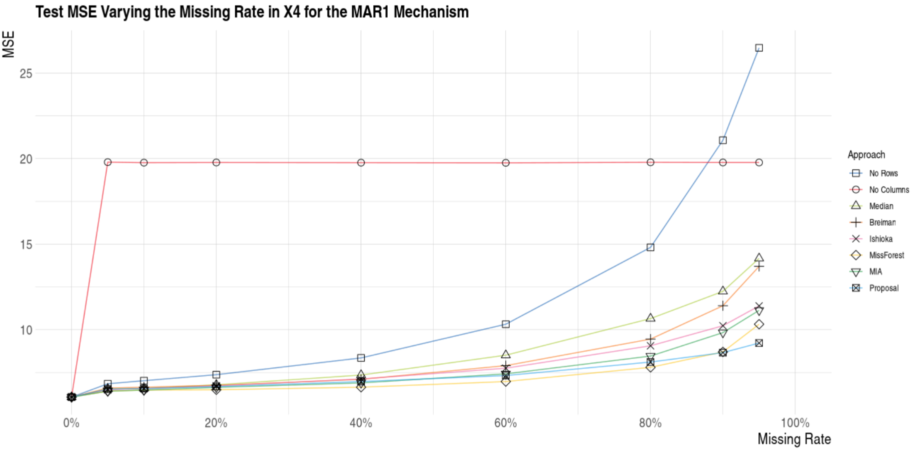

Average Bias by Approach
Bias
0.8
0.6
0.4
0.2
0.0
-0.2
-0.4
COMP
MCAR
MAR1
MAR2
MAR3
MAR4
LOG
DEPY
Mechanism
Approach
No Rows
Median
Breiman
Ishioka
MissForest
MIA
Proposal

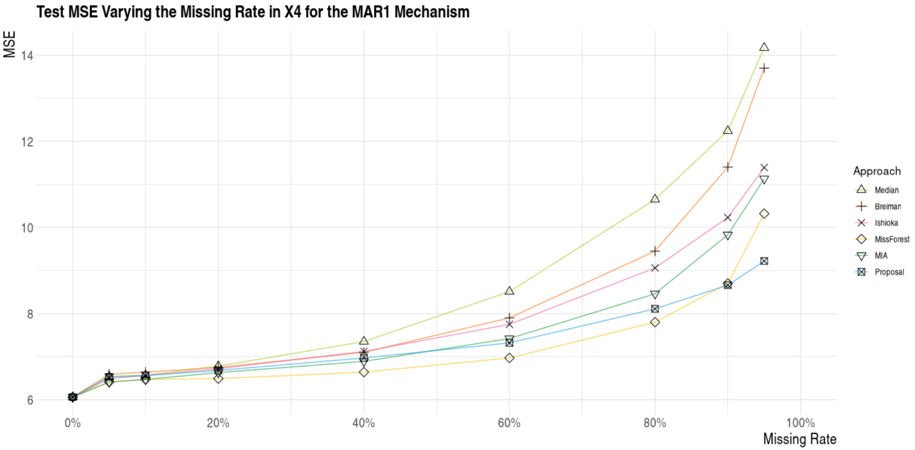
Changing the Percentage of Missingness

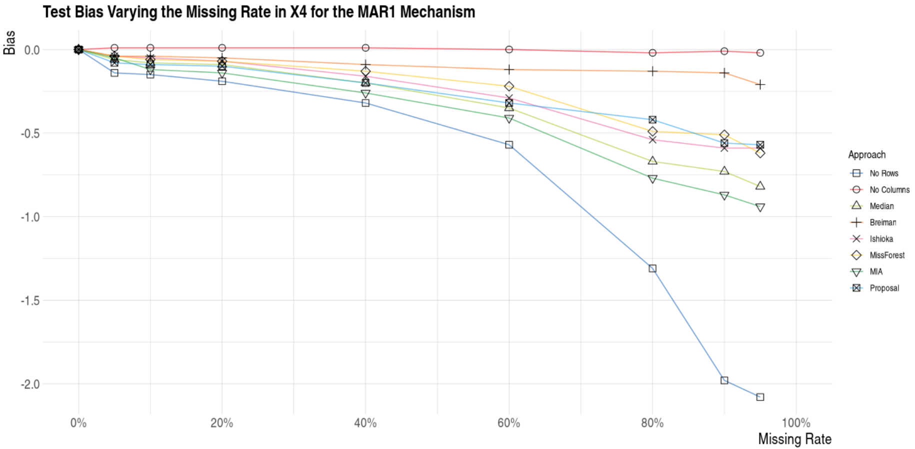
- MissForest and our approach generate the best estimators in terms MSE.
- Our method requires the construction of a single random forest.
- It does not require the computation of the proximity matrix.

Extending the study, we **change the percentage of missingness for $\mathbf{X}^{(4)}$**, without changing the percentage of the other variables.

$\mathbf{X}^{(1)}$	$\mathbf{X}^{(3)}$	$\mathbf{X}^{(4)}$
20%	10%	5%, 10%, 20%, 40%, 60%, 80%, 90%, 95%


Test MSE Varying the Missing Rate in X4 for the MAR1 Mechanism
MSE
25
20
15
10
0%
20%
40%
60%
80%
100%
Missing Rate
Approach
No Rows
No Columns
Median
Breiman
Ishioka
MissForest
MIA
Proposal


Test MSE Varying the Missing Rate in X4 for the MAR1 Mechanism
MSE
14
12
10
8
6
0%
20%
40%
60%
80%
100%
Missing Rate
Approach
Median
Breiman
Ishioka
MissForest
MIA
Proposal


Test Bias Varying the Missing Rate in X4 for the MAR1 Mechanism
Bias
0.0
-0.5
-1.0
-1.5
-2.0
0%
20%
40%
60%
80%
100%
Missing Rate
Approach
No Rows
No Columns
Median
Breiman
Ishioka
MissForest
MIA
Proposal

Conclusions

We observe that:

- When the percentage of **missingness is less than 40% most of the methods present similar MSE and bias**.
- The **differences** between the approach are **clear when** the percentage of missingness is **larger than 60%**.
- For extremely large percentage of missing values, **the proposal surpass all the other methods**.

Consistency

Importance of Consistency

The first and weakest property ***an estimate*** *should have is that, as the sample size grows, it* ***should converge to the estimated quantity****, i.e., the error of the estimate should converge to zero for a sample size tending to infinity. Estimates which have this property are called consistent.*[3]

We show that our proposed method generates consistent estimators under some conditions.

[3] László Györfi et al. *A distribution-free theory of nonparametric regression.* Springer Science & Business Media, 2002.

Consistency in Probability / Hypothesis

A sequence of regression function estimates $(m_n)_n$ is called **consistent in probability** for a certain distribution of $(\mathbf{X}, Y)$ if, for all $\xi, \rho > 0$ there exists $N \in \mathbb{N}^\star$, such that for all $n \geq N$

$$\mathbb{P}\left[|m_n(\mathbf{X}) - m(\mathbf{X})| \leq \xi\right] \geq 1 - \rho$$

Hypothesis (1)

The response variable Y is of the form

$$Y = \sum_{j=1}^{p} m_j(\mathbf{X}^{(j)}) + \varepsilon$$

where $\mathbf{X}$ is uniformly distributed over $[0,1]^p$, ε is an independent Gaussian centered noise with finite variance $\sigma^2 > 0$ and each component m_j is continuous.

Hypothesis / Theorem

Hypothesis (2)

The random variables $\mathbf{X}_i^{(h)}$ *are missing accordingly to an MCAR mechanism. The probability of missingness* $p_n^{(h)} = \mathbb{P}\left[\mathbf{M}^{(h)} = 1\right]$ *only depends on the size* n *of the sample* $\mathcal{D}_n$ *and* $\lim_{n\to\infty} p_n^{(h)} = c^{(h)}$ *where* $0 < c^{(h)} < 1$ *is constant for all* $h \in \{1, \ldots, p\}$.

Theorem 1

Assume that Hypothesis 1 and 2 hold. Then, under the condition $q_n \to \infty$, the random forest estimator with missing values is consistent in probability.

Sketch of the Proof

We define, for any cell A, the variation of m within A as

$$\Delta(m, A) = \sup_{\mathbf{x}, \mathbf{x}' \in A} |m(\mathbf{x}) - m(\mathbf{x}')|$$

1. Show that asymptotically **the regression function has no variation on the final cells**. Thus the regression function can be properly approximated by a function constant by pieces (Proposition 1).
2. Show that **if the theoretical CART-criterion $L^\star$ satisfies**

   $$\mathbb{P}\left[L^\star\left(A_{s(n)}, d^\star_{s(n)}, w^\star_{s(n)}\right) \leq \xi\right] \geq 1 - \rho$$

   then $\Delta(m, A_{s(n)}(\mathbf{x})) \to 0$ almost surely (Lemma 1).

3. Show that the **empirical CART-criterion L_n satisfies**

$$\mathbb{P}\left[L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \leq \xi\right] \geq 1-\rho$$

   (Lemma 2).

4. By Law of Large Numbers, and Proposition 1. Show that **there exists a consistent estimator m_n'**, where

$$m_n'(\mathbf{X},\Theta) = \frac{1}{N(A_{s(n)}(\mathbf{X},\Theta))}\sum_{i=1}^{n} Y_i \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X},\Theta)}.$$

5. Show that **our estimator** is asymptotically **equivalent to m_n'.**

Relevance of Hypothesis 1

If m is non-additive then we could have that the CART criterion equals zero even when m is not constant. For example, consider the function $m(\mathbf{x}^{(1)}, \mathbf{x}^{(2)}) = \mathbf{x}^{(1)} + \mathbf{x}^{(2)} - 2\mathbf{x}^{(1)}\mathbf{x}^{(2)}$, the cell $A = [0,1]^2$ and a cut $d = (1, z)$ $(z \in (0,1))$, then

$$\begin{aligned}\mu_{A_L} &= \frac{1}{z}\int_0^z \int_0^1 (\mathbf{x}^{(1)} + \mathbf{x}^{(2)} - 2\mathbf{x}^{(1)}\mathbf{x}^{(2)}) d\mathbf{x}^{(2)} d\mathbf{x}^{(1)} \\ &= \frac{1}{z}\int_0^z (\mathbf{x}^{(1)} + \frac{1}{2} - \mathbf{x}^{(1)}) d\mathbf{x}^{(1)} = \frac{1}{2}\end{aligned}$$

Note that μ_{A_L} does not depend on z, similarly we can show that $\mu_{A_R} = 1/2$ and **therefore $L^\star = 0$ for all cuts even when m is not constant**.

The conditions over the **Gaussian errors and the uniformity of X are more technical requirements**, changes on these conditions, like sub-Gaussian errors or distributions with support in $[0, 1]^p$ for $\mathbf{X}$, would require to adapt the technical parts of the proof with little relevance (Lemma 1).

The conditions on the **MCAR mechanism and** $p_n^{(h)} \to c^{(h)} \neq 1$ are used extensively to ensure that we have **enough number of observations in the final cells**, and that the balance in the the number of points is maintained (Lemma 2).

Autoencoders

Imputing with Autoencoders

- We impute the missing values using autoencoders.
- **We derived a loss function to trained variational autoencoders (VAEs) with missing data, that we have called EMMELBO**.

Autoencoders are neural networks:

- which generate a non-linear representation of an observation in an smaller dimension (the encoder portion).
- which recover the original observation from the latent representation (the decoder portion).

Dimensionality Reduction / Encoder and Decoder Framework

We want to find a space of smaller dimension than the input space to represent the data.

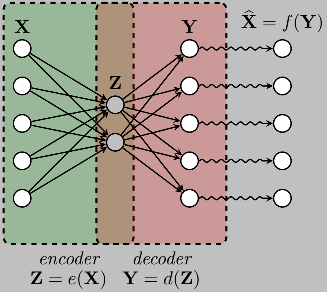
- **Encoder:** produces the new representation from the “old features”, $e \in \mathcal{E}$.
- **Decoder:** recover the input from the small.dimensional representation, $d \in \mathcal{D}$.

We want to to find the best encoder/decoder pair among a given family, i.e. we want to minimize a loss function

$$(e^\star, d^\star) \in \underset{(e,d)\in\mathcal{E}\times\mathcal{D}}{\arg\min}\ L(\mathbf{X}, d(e(\mathbf{X})))$$

Autoencoder

In autoendores the encoder and the decoder are $\mathcal{E}$ and $\mathcal{D}$ are families of neural networks.

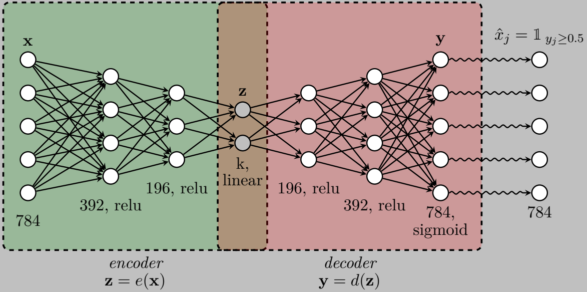


$\mathbf{X}$
$\mathbf{Z}$
$\mathbf{Y}$
$\widehat{\mathbf{X}} = f(\mathbf{Y})$
encoder
$\mathbf{Z} = e(\mathbf{X})$
decoder
$\mathbf{Y} = d(\mathbf{Z})$

$$(e^\star, d^\star) \in \operatorname*{arg\,min}_{(e,d)\in\mathcal{E}\times\mathcal{D}} \mathbb{E}_{\mathcal{D}_n} L(\mathbf{x}, d(e(\mathbf{x})))$$

If x_j follows a Bernoulli distribution we could choose $L(x_j, y_j)$ as the negative of the log-likelihood

$$L(\mathbf{x},\mathbf{y}) = \frac{1}{p}\sum_{j=1}^{p} L(x_j, y_j) = \frac{1}{p}\sum_{j=1}^{p} -x_j \log(y_j) - (1-x_j)\log(1-y_j)$$



ReLu: $f(\mathbf{x}) = \max\{0, \mathbf{w}^T\mathbf{x} + b\}$, sigmoid: $f(\mathbf{x}) = \frac{1}{1+e^{(-\mathbf{w}^T\mathbf{x}+b)}}$, linear: $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$.

Autoencoder vs PCA

The autoencoder is able to reconstruct the images, while PCA cannot do it.

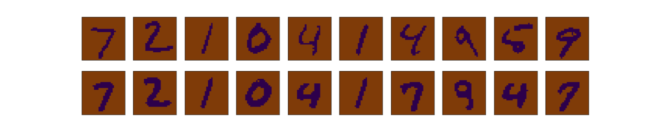


Figure: Reconstruction using our autoencoder when the dimension of the latent space is $k = 2$.

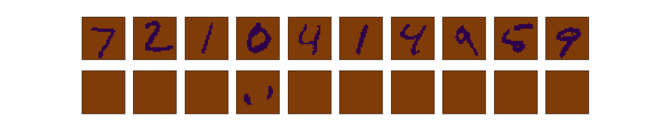


Figure: Reconstruction using the first 2 principal components.

Latent Representation, AE vs PCA

The **autoencoder learn the important feature**s of the images to distinguish between the digits that they represent, while **PCA cannot learn these features**.

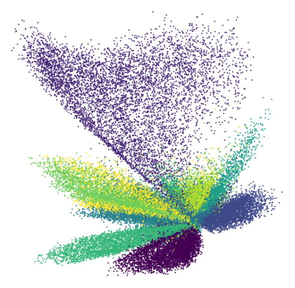


Figure: Latent space using the autoencoder.

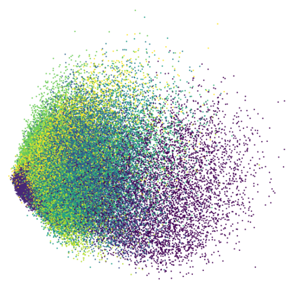


Figure: Induced space by the first two principal components.

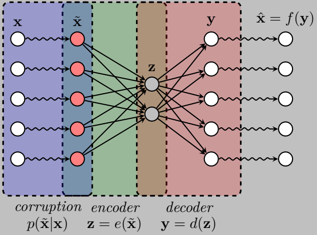
Denoising Autoencoders (DAEs)

Denoising Autoencoders (DAEs)

- A denoising autoencoder is an autoencoder that receives a corrupted data point as its input and is trained to predict the original, uncorrupted data point as its output.
- A denoising autoencoder minimizes

$$L(\mathbf{x}, d(e(\tilde{\mathbf{x}})))$$

where $\tilde{\mathbf{x}}$ is a copy of $\mathbf{x}$ that has been corrupted.

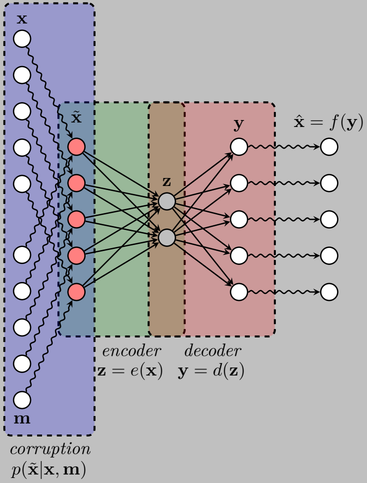

$\mathbf{x}$
$\tilde{\mathbf{x}}$
$\mathbf{z}$
$\mathbf{y}$
$\hat{\mathbf{x}} = f(\mathbf{y})$
corruption
$p(\tilde{\mathbf{x}}|\mathbf{x})$
encoder
$\mathbf{z} = e(\tilde{\mathbf{x}})$
decoder
$\mathbf{y} = d(\mathbf{z})$

Use of DAEs

- Autoencoders encourage $d \circ e$ to be merely the identity function if they have the capacity to do so.
- Denoising autoencoders (**DAEs**) **must undo the corruption** rather than simply copy their input.
- **DAEs** were first introduced by Vincent et al.[4] as a **robust procedure** to get a good representation.
- **DAEs** make **possible to reconstruct from partial observation**.
- The corruption process $p(\tilde{\mathbf{x}}|\mathbf{x}, \mathbf{m})$ represents the conditional distribution over corrupted samples given the original data sample and the missing pattern.

[4] Pascal Vincent et al. "Extracting and composing robust features with denoising autoencoders". In: *Proceedings of the 25th international conference on Machine learning.* 2008, pp. 1096–1103.

Corruption Process / Rough Imputation


$\mathbf{x}$
$\tilde{\mathbf{x}}$
$\mathbf{z}$
$\mathbf{y}$
$\hat{\mathbf{x}} = f(\mathbf{y})$
$\mathbf{m}$
encoder
$\mathbf{z} = e(\mathbf{x})$
decoder
$\mathbf{y} = d(\mathbf{z})$
corruption
$p(\tilde{\mathbf{x}}|\mathbf{x}, \mathbf{m})$

$$\tilde{x}_j|\mathbf{x}, \mathbf{m} = \begin{cases} 1 & \text{if } x_j = 1 \text{ and } m_j = 0 \\ 0 & \text{if } x_j = 0 \text{ and } m_j = 0 \\ 0.5 & \text{if } m_j = 1 \end{cases}$$

Reconstruction of Images Using DAEs

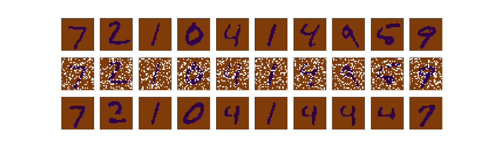


Figure: Reconstruction using our DAE when the dimension of the latent space is $k = 2$ and 157 pixels have missing values.

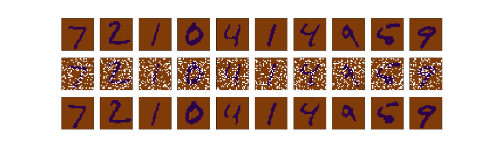


Figure: Reconstruction using our DAE when the dimension of the latent space is $k = 98$ and 157 pixels have missing values.

Variational Inference

Bayesian Framework

- $p(\mathbf{z})$ is the prior density for the latent variables.
- $p(\mathbf{x}|\mathbf{z})$ is the likelihood of the observations.
- **We want to compute the posterior density** $p(\mathbf{z}|\mathbf{x})$.

$$p(\mathbf{z}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p(\mathbf{x})}$$

The denominator is the density of the observations, also called the **evidence**.

$$p(\mathbf{x}) = \int p(\mathbf{x}, \mathbf{z})d\mathbf{z} = \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

It is quite common that this integral **is intractable**.

Variational Inference

The idea of variational inference is to **propose a family** $\mathcal{Q}$ of densities over the latent variable and then find the member of the family that **minimizes the Kullback-Leibler (KL) divergence to the exact posterior**,

$$q^\star \in \arg\min_{q \in \mathcal{Q}} D_{KL}(q(\mathbf{z}|\mathbf{x})||p(\mathbf{z}|\mathbf{x}))$$

However, it cannot be computed since it requires to calculate the evidence $\log p(\mathbf{x})$.

ELBO

Fortunately, we can derive an equivalent expression which enables us to solve the optimization problem. This expression is known as the Evidence Lower Bound (ELBO).

$$\mathrm{ELBO}(q) = \mathbb{E}_{\mathbf{z}\sim q}\left[\log p(\mathbf{x},\mathbf{z})\right] - \mathbb{E}_{\mathbf{z}\sim q}\left[\log q(\mathbf{z}|\mathbf{x})\right]$$

$$q^{\star} \in \arg\max_{q\in\mathcal{Q}} \mathrm{ELBO}(q)$$

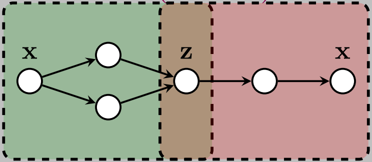
We can rearrange $\mathrm{ELBO}(q)$ to get a more convenient form

$$\mathrm{ELBO}(q) = \mathbb{E}_{\mathbf{z}\sim q}\left[\log p(\mathbf{x}|\mathbf{z})\right] - D_{\mathrm{KL}}(q(\mathbf{z}|\mathbf{x})||p(\mathbf{z}))$$

This form looks like the well-known **trade-off between the likelihood and the prior**.

Variational Autoencoders (VAEs)

Variational Autoencoders (VAEs)



encoder
$q(\mathbf{z}|\mathbf{x})$

decoder
$p(\mathbf{x}|\mathbf{z})$

The ELBO is the core and loss function of a VAE since $q(\mathbf{z}|\mathbf{x})$ is encoding $\mathbf{x}$ into $\mathbf{z}$ and $p(\mathbf{x}|\mathbf{z})$ is decoding it to reconstruct $\mathbf{x}$.

- $\mathbf{z} \sim \mathcal{N}_k(0, I)$.
- $x_j|\mathbf{z} \overset{iid}{\sim} \text{Ber}(d_j(\mathbf{z}))$, $d = (d_1, \dots, d_p) \in \mathcal{D}$.
- $q(\mathbf{z}|\mathbf{x}) \equiv \mathcal{N}_k(\mu(\mathbf{x}), \sigma(\mathbf{x}))$.
  - $\sigma(\mathbf{x}) = \text{diag}(\sigma_1(\mathbf{x}), \dots, \sigma_k(\mathbf{x})) \in \mathcal{S}$.
  - $\mu(\mathbf{x}) = (\mu_1(\mathbf{x}), \dots, \mu_k(\mathbf{x})) \in \mathcal{M}$.

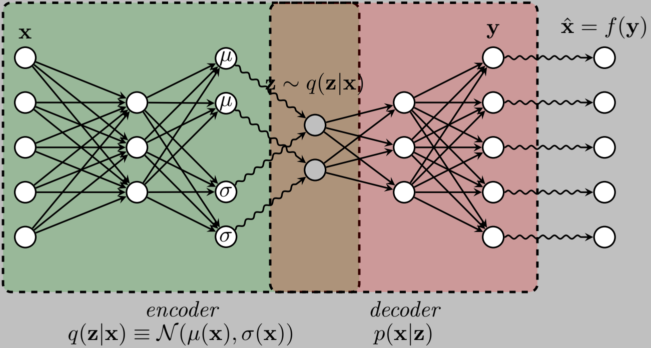
Optimization Problem

Thus, maximizing the ELBO over q is equivalent to

$$(d^\star, \mu^\star, \sigma^\star) \in$$

$$\underset{(d,\mu,\sigma)\in\mathcal{D}\times\mathcal{M}\times\mathcal{S}}{\arg\max} \mathbb{E}_{\mathbf{z}\sim q}\left[\sum_{j=1}^{p} x_j \log(d_j(\mathbf{z})) + (1-x_j)\log(1-d_j(\mathbf{z}))\right] - \frac{1}{2}\sum_{\kappa=1}^{k}\left[\sigma_\kappa(\mathbf{x}) + \mu_\kappa^2(\mathbf{x}) - \log\sigma_\kappa(\mathbf{x})\right]$$

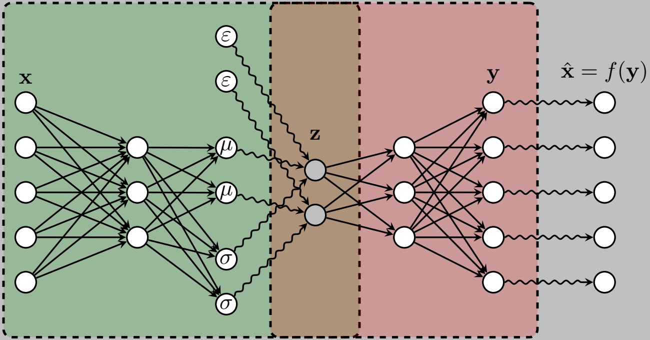
Note that **we need to calculate an expected value** with $\mathbf{z} \sim q$.


$\mathbf{x}$
μ
μ
σ
σ
$\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})$
$\mathbf{y}$
$\hat{\mathbf{x}} = f(\mathbf{y})$
encoder
$q(\mathbf{z}|\mathbf{x}) \equiv \mathcal{N}(\mu(\mathbf{x}), \sigma(\mathbf{x}))$
decoder
$p(\mathbf{x}|\mathbf{z})$

There is a significant **problem**. We need to **back-propagate the error through a layer that samples $\mathbf{z}$** from q. However this sample procedure **is not differentiable**, and this autoencoder cannot be learned by back-propagation.

Reparametrization Trick

The **solution**, called the "**reparametrization trick**", is to first sample $\varepsilon \sim \mathcal{N}_k(0, I)$ and then compute $\mathbf{z} = \mu(\mathbf{x}) + \sigma(\mathbf{x})^{1/2}\varepsilon$.



encoder
$q(\mathbf{z}|\mathbf{x}) \equiv \mathcal{N}_k(\mu(\mathbf{x}), \sigma(\mathbf{x}))$

decoder
$p(\mathbf{x}|\mathbf{z})$

Optimization Problem Using Rearametrization Trick

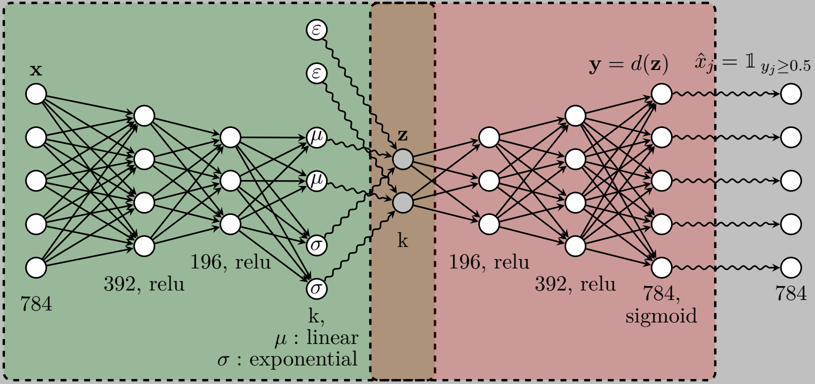
Finally, the optimization problem is

$$(d^\star, \mu^\star, \sigma^\star) \in$$

$$\underset{(d,\mu,\sigma)\in\mathcal{D}\times\mathcal{M}\times\mathcal{S}}{\arg\max}\ \mathbb{E}_{\varepsilon\sim\mathcal{N}_k(0,I)}\Bigg[$$

$$\sum_{j=1}^{p} x_j \log(d_j(\mathbf{z})) + (1-x_j)\log(1-d_j(\mathbf{z}))\Big|\mathbf{z} = \mu(\mathbf{x}) + \sigma(\mathbf{x})^{1/2}\varepsilon\Bigg]$$

$$-\frac{1}{2}\sum_{\kappa=1}^{k}\left[\sigma_\kappa(\mathbf{x}) + \mu_\kappa^2(\mathbf{x}) - \log\sigma_\kappa(\mathbf{x})\right]$$



encoder

$q(\mathbf{z}|\mathbf{x}) \equiv \mathcal{N}_k(\mu(\mathbf{x}), \sigma(\mathbf{x}))$

decoder

$p(\mathbf{x}|\mathbf{z}) = \prod_{j=1}^{784} d_j(\mathbf{z})^{x_j}(1 - d_j(\mathbf{z}))^{(1-x_j)}$

exponential: $f(\mathbf{x})=\exp\{\mathbf{w}^T\mathbf{x}+b\}$.

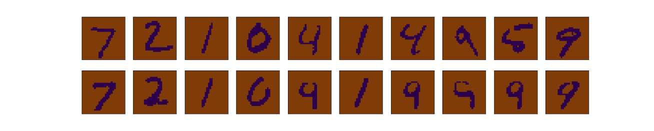


Figure: Reconstruction using our VAE when the dimension of the latent space is $k = 2$.

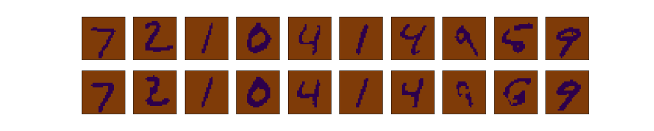


Figure: Reconstruction using our VAE when the dimension of the latent space is $k = 98$.

Latent Representation, AE vs VAE

The **variational autoencoder** forces the latent representations to follow (approx.) the prior distribution (normal), creating a **well-organized space** (continuity and completeness).

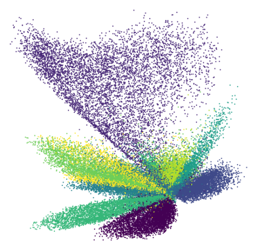


Figure: Latent space induced by our AE when the dimension of the latent space is $k = 2$.

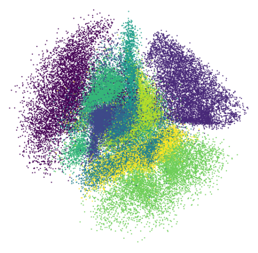


Figure: Latent space induced by our VAE when the dimension of the latent space is $k = 2$.

Variational Autoencoders (VAEs) with Missing Data

Variational Autoencoders with Missing Data

Assume that $\tilde{\mathbf{x}}$ is a corrupted version of a data point $\mathbf{x}$, built with the observed part of $\mathbf{x}$ and its missingness pattern $\mathbf{m}$. We define the **Evidence and Missingness Mechanism Lower Bound (EMMELBO)** as

Definition 1 EMMELBO

$$\begin{aligned}\mathrm{EMMELBO}(q(\mathbf{z}|\tilde{\mathbf{x}}), p(\mathbf{x},\mathbf{m}|\mathbf{z})) =& \log p(\mathbf{x}) + \log p(\mathbf{m}|\mathbf{x}) \\ & - D_{KL}(q(\mathbf{z}|\tilde{\mathbf{x}})||p(\mathbf{z}|\tilde{\mathbf{x}}))\end{aligned}$$

where $(q(\mathbf{z}|\tilde{\mathbf{x}}), p(\mathbf{x},\mathbf{m}|\mathbf{z})) \in \mathcal{Q} \times \mathcal{P}$, and $\mathcal{Q}$ and $\mathcal{P}$ are parametric families of distributions.

EMMELBO

We recognize on the right-hand side the terms $\log p(\mathbf{x}) + \log p(\mathbf{m}|\mathbf{x})$, the sum of the log-evidence and the log-data-missing mechanism, thus the **EMMELBO is a lower bound** of this sum.

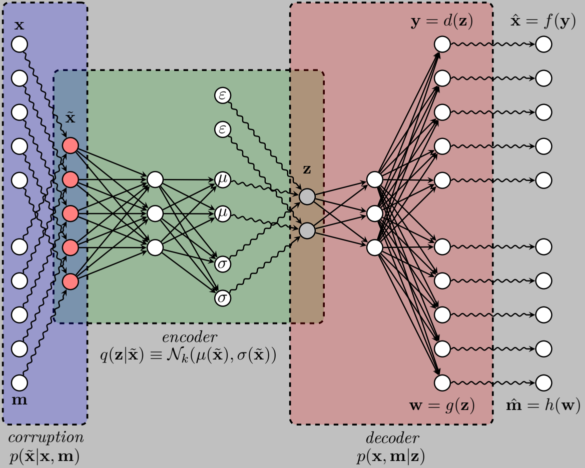
On the other hand, it can be shown that

$$p(\mathbf{z}|\tilde{\mathbf{x}}) = \frac{p(\mathbf{x}, \mathbf{m}|\mathbf{z})p(\mathbf{z})}{p(\mathbf{x}, \mathbf{m})}$$

From here, we can write the EMMELBO in the **more convenient form**

$$\text{EMMELBO}(q(\mathbf{z}|\tilde{\mathbf{x}}), p(\mathbf{x}, \mathbf{m}|\mathbf{z})) = \mathbb{E}_{\mathbf{z}\sim q(\mathbf{z}|\tilde{\mathbf{x}})}\left[\log p(\mathbf{x}, \mathbf{m}|\mathbf{z})\right] - D_{KL}(q(\mathbf{z}|\tilde{\mathbf{x}})||p(\mathbf{z}))$$

VAE Architecture for Missing Data


$\mathbf{x}$
$\tilde{\mathbf{x}}$
ε
ε
μ
μ
σ
σ
$\mathbf{z}$
$\mathbf{y} = d(\mathbf{z})$
$\hat{\mathbf{x}} = f(\mathbf{y})$
$\mathbf{m}$
$\mathbf{w} = g(\mathbf{z})$
$\hat{\mathbf{m}} = h(\mathbf{w})$
encoder
$q(\mathbf{z}|\tilde{\mathbf{x}}) \equiv \mathcal{N}_k(\mu(\tilde{\mathbf{x}}), \sigma(\tilde{\mathbf{x}}))$
corruption
$p(\tilde{\mathbf{x}}|\mathbf{x}, \mathbf{m})$
decoder
$p(\mathbf{x}, \mathbf{m}|\mathbf{z})$

Optimization Problem

The optimization problem to train a VAE with missing data is

$$(d^\star, \mu^\star, \sigma^\star) \in$$

$$\underset{(d,\mu,\sigma)\in\mathcal{D}\times\mathcal{M}\times\mathcal{S}}{\arg\max}\; \mathbb{E}_{\mathbf{z}\sim q(\mathbf{z}|\tilde{\mathbf{x}})}\left[\sum_{j=1}^{p} \tilde{x}_j \log(d_j(\mathbf{z})) + (1-\tilde{x}_j)\log(1-d_j(\mathbf{z}))\right]$$

$$-\frac{1}{2}\sum_{\kappa=1}^{k}\left[\sigma_\kappa(\tilde{\mathbf{x}}) + \mu_\kappa^2(\tilde{\mathbf{x}}) - \log\sigma_\kappa(\tilde{\mathbf{x}})\right]$$

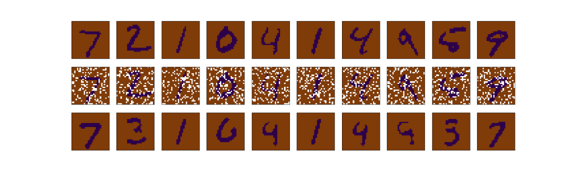


Figure: Reconstruction using our VAE for missing data when the dimension of the latent space is $k = 2$ and 157 pixels have missing values.

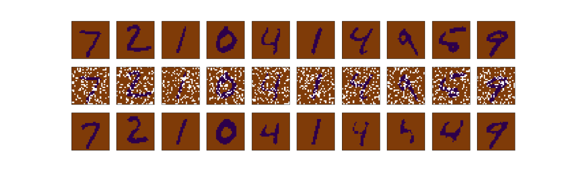


Figure: Reconstruction using our VAE for missing data when the dimension of the latent space is $k = 98$ and 157 pixels have missing values.

General Conclusions and Contributions

General Conclusions and Contributions

- We **proposed a new technique based on random forests** and the CART criterion to estimate the regression function when there are missing entries.
- We compare the proposal to other techniques based on random forests. The proposed approach has **similar MSE to state-of-the-art** methods.
- For **large percentage of missing data**, the proposal **surpass all the other techniques**.
- We show the **consistency of the estimators** created with the proposed approach.
- We **derive a loss function to train variational autoencoders with missing data**.

Thank you!

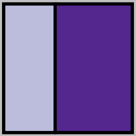
Details

Admissible Assignations

Sort the Target Variable

- Assume that $\bar{Y}_{L,obs} \leq \bar{Y}_{R,obs}$.
- $\mathbf{i}_{miss} = \{1, ..., N_{miss}\}$ indexes of observations with missing value.
- $Y_{(1)} \leq \cdots \leq Y_{(N_{miss})}$

Maximizing the CART criterion implies **assigning the lowest values** of Y **to the left** and the **largest values** of Y **to the right**.



$Y_{(1)} =$ ●

$Y_{(2)} =$ ●

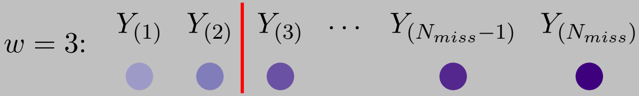
$\vdots$

$Y_{(N_{miss})} =$ ●

Split the Sorted Vector

- Denote by $w \in \{1, ..., N_{miss} + 1\}$ the position at which the vector of indexes $\mathbf{i}_{miss}$ is split.
- $Y_{(1)}, \ldots, Y_{(w-1)}$ area assigned to the left.
- $Y_{(w)}, \ldots, Y_{(N_{miss})}$ area assigned to the right.

Therefore $\mathcal{W}_A^{(h)}$ has a cardinality of $N_{miss}(A)^{(h)} + 1$.



Concavity of CART-criterion

Furthermore, we have observed that the CART criterion is concave as function of w.

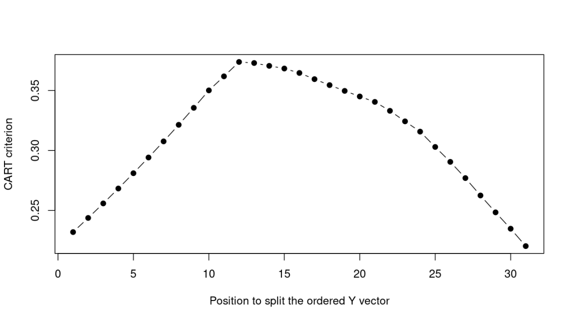


Figure: The CART-criterion is concave as function of w.

Allowing the introduction of methods like bisection to find the optimum assignation.

Random Forests Simulation Study

Other Methods

Along with our approach, we consider 3 simple (yet popular) methods used in practice to handle missing values:

- Removing the columns that have missing values and constructing a random forest.
- Removing the observations with missing values and constructing a random forest.
- Imputing the missing values with the median of the observations in the corresponding variable and constructing a random forest.

The parameters of the random forests are the same, except when we eliminate observations with missing values, we have established $a_n = \lceil 0.632n \rceil$ and n is the number of complete observations.

Breiman's and Ishioka's approaches operate through imputation of missing values in a recursive way. First, they use the original training data set, $\mathcal{D}_n$, to fill the blank spaces in a roughly way, this data set is used to build a random forest.

Then the proximity matrix is used to improve the imputation, resulting in a new data set $\mathcal{D}_{n,t_2}$. The procedure follows iteratively. Let be $K_{M,t_\ell}(i,j)$ the proximity between $\mathbf{X}_i$ and $\mathbf{X}_j$ at time t_ℓ.

Breiman's Approach. If $\mathbf{X}^{(h)}$ is a continuous variable,

$$\widehat{\mathbf{X}}^{(h)}_{j,t_{\ell+1}} = \frac{\sum_{i\in\mathbf{i}^{(h)}_{obs}} K_{M,t_\ell}(i,j)\mathbf{X}^{(h)}_i}{\sum_{i\in\mathbf{i}^{(h)}_{obs}} K_{M,t_\ell}(i,j)}, \qquad \begin{array}{l}\ell \geq 1 \\ j \in \mathbf{i}^{(h)}_{miss}\end{array}$$

Ishioka's Approach. If $\mathbf{X}^{(h)}$ is a continuous variable,

$$\widehat{\mathbf{X}}^{(h)}_{j,t_{\ell+1}} = \frac{\sum_{\substack{i\in\text{neigh}_k \\ i\neq j}} K_{M,t_\ell}(i,j)\widehat{\mathbf{X}}^{(h)}_{i,t_\ell}}{\sum_{\substack{i\in\text{neigh}_k \\ i\neq j}} K_{M,t_\ell}(i,j)}, \qquad \begin{array}{l}\ell \geq 1 \\ j \in \mathbf{i}^{(h)}_{miss}\end{array}$$

MissForest. This algorithm begins with a rough imputation for the missing values. Then, for each direction $\mathbf{X}^{(h)}$ a random forests is built using all the other directions $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(h-1)}, \mathbf{X}^{(h+1)}, \ldots, \mathbf{X}^{(p)}$ and the response. Then the missing values of $\mathbf{X}^{(h)}$ are predicted with this random forests. These steps are repeated iteratively until a stopping rule is achieved.

Missing Incorporated in Attributes (MIA). The Missing Incorporated in Attributes (MIA) approach consists in keeping all the missing values together when a split is performed. Thus, the splits with this approach assign the values according to one of the following rules:

- $\{\mathbf{X}^{(h)} < z \text{ and } \mathbf{M}^{(h)} = 1\}$ versus $\{\mathbf{X}^{(h)} \geq z\}$.
- $\{\mathbf{X}^{(h)} < z\}$ versus $\{\mathbf{X}^{(h)} \geq z \text{ and } \mathbf{M}^{(h)} = 1\}$.
- $\{\mathbf{M}^{(h)} = 0\}$ versus $\{\mathbf{M}^{(h)} = 1\}$.

Data-Missing Mechanisms

- For the missing values we create 7 different mechanisms of missingness:
    - 1 Missing Completely At Random (MCAR).
    - 5 Missing At Random (MAR1, MAR2, MAR3, MAR4, Depy).
    - 1 Not Missing At Random (LOG).

- **MAR1** The probability of `NA` is

$$\frac{2 \times \mathrm{rank}(\text{determinant variable})}{n(n+1)}$$

- **MAR2** We create two groups in the determinant variable. An observation belongs to the first group if it is bigger to the the median, otherwise it belongs to the second group. The probability of `NA` for each group is

$$0.9/\#(\text{obs. in 1st group}) \quad 0.1/\#(\text{obs. in 2nd group})$$

- **MAR3** The biggest values in the determinant variable are `NA` in the missing variable.
- **MAR4** The biggest and smallest values in determinant variable are `NA` in the missing variable.
- **DEPY** Probability of `NA` is 0.1 if $Y \geq 13$, otherwise is 0.4
- **LOG**

$$\operatorname{logit}(\mathbb{P}[\mathbf{M}^{(h)} = 1]) = -0.5 + \sum_{\substack{k=1 \\ k \neq h}}^{5} \mathbf{X}^{(k)}$$

Decomposition of the CART Criterion

Decomposition of the Empirical CART Criterion

Applying elementary algebra, we can show that our CART criterion might be written as

$$\begin{aligned} L_n\left(A,d,w\right) = & L_{1,n}(A,d) + L_{2,n}\left(A,d,w\right) \\ & + L_{3,n}\left(A,d,w\right) + L_{4,n}\left(A,d,w\right) \end{aligned}$$

where

$$\begin{aligned} L_{1,n}(A,d) = & \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A,obs}\right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, \mathbf{M}_i^{(h)}=0} \\ & - \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A_L,obs}\right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, a^{(h)} \leq \mathbf{X}_i^{(h)} \leq z,\, \mathbf{M}_i^{(h)}=0} \\ & - \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A_R,obs}\right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, z \leq \mathbf{X}_i^{(h)} \leq b^{(h)},\, \mathbf{M}_i^{(h)}=0} \end{aligned}$$

$$
\begin{aligned}
L_{2,n}(A,d,w) = & \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A,miss} \right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, \mathbf{M}_i^{(h)}=1} \\
& - \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A_L,miss} \right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, a^{(h)} \leq \widehat{\mathbf{X}}_{i,out}^{(h)} \leq z,\, \mathbf{M}_i^{(h)}=1} \\
& - \frac{1}{\widehat{N}(A)} \sum_{i=1}^{n} \left(Y_i - \widehat{Y}_{A_R,miss} \right)^2 \mathbb{1}_{\widehat{\mathbf{X}}_{i,in} \in A,\, z \leq \widehat{\mathbf{X}}_{i,out}^{(h)} \leq b^{(h)},\, \mathbf{M}_i^{(h)}=1}
\end{aligned}
$$

$$
\begin{aligned}
L_{3,n}(A,d,w) = & \frac{\widehat{N}_{obs}(A)}{\widehat{N}(A)}\left(\widehat{Y}_{A,obs} - \widehat{Y}_{A}\right)^2 \\
& - \frac{\widehat{N}^{(h)}_{obs}(A_L)}{\widehat{N}(A)}\left(\widehat{Y}_{A_L,obs} - \widehat{Y}_{A_L}\right)^2 \\
& - \frac{\widehat{N}^{(h)}_{obs}(A_R)}{\widehat{N}(A)}\left(\widehat{Y}_{A_R,obs} - \widehat{Y}_{A_R}\right)^2
\end{aligned}
$$

$$\begin{aligned} L_{4,n}(A,d,w) = & \frac{\widehat{N}_{miss}(A)}{\widehat{N}(A)} \left(\widehat{Y}_{A,miss} - \widehat{Y}_{A}\right)^2 \\ & - \frac{\widehat{N}^{(h)}_{miss}(A_L)}{\widehat{N}(A)} \left(\widehat{Y}_{A_L,miss} - \widehat{Y}_{A_L}\right)^2 \\ & - \frac{\widehat{N}^{(h)}_{miss}(A_R)}{\widehat{N}(A)} \left(\widehat{Y}_{A_R,miss} - \widehat{Y}_{A_R}\right)^2 \end{aligned}$$

Decomposition of the Theoretical CART Criterion

Analogously, the Theoretical CART criterion can be written as

$$\begin{aligned} L^\star(A,d,w) = & L_1^\star(A,d) + L_2^\star(A,d,w) \\ & + L_3^\star(A,d,w) + L_4^\star(A,d,w) \end{aligned}$$

- $L_1^\star$ (resp. $L_2^\star$) measures the change of variance of the points where the split variable is observed (missing).
- $L_3^\star$ (resp. $L_4^\star$) measures the change of the squared bias of the points where the split variable is observed (missing).

Bias-Variance Trade-Off

This leads to the well-known bias-variance trade-off.

- When $L_3^\star$ or $L_4^\star$ are different from zero, the data-missing mechanism is introducing a source of bias. **(Conjecture)**.
- We have observed in our simulations that the MCAR mechanism seems to not introduce any bias.
- We expect that $L_{3,n}$ and $L_{4,n}$ would take values near zero. Our simulation shows results that sustain these observations.

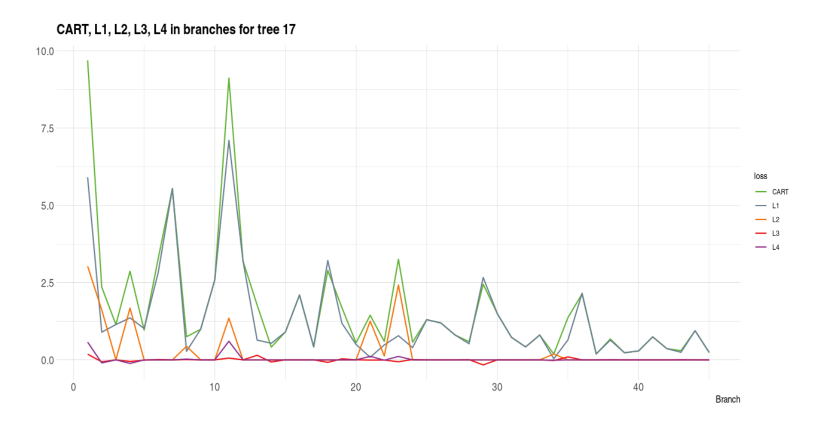


Figure: L_n, $L_{1,n}$, $L_{2,n}$, $L_{3,n}$ and $L_{4,n}$ in the cells of a tree in a random forest where an MCAR mechanism was present in the data set.

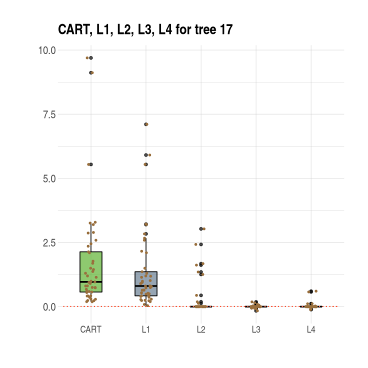


Figure: L_n, $L_{1,n}$, $L_{2,n}$, $L_{3,n}$ and $L_{4,n}$ boxplots for a tree in a random forest where an MCAR mechanism was present in the data set.

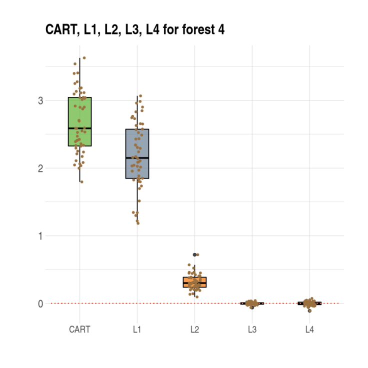


Figure: L_n, $L_{1,n}$, $L_{2,n}$, $L_{3,n}$ and $L_{4,n}$ boxplots for a random forest an MCAR mechanism was present in the data set, each point represents the mean value of a tree.

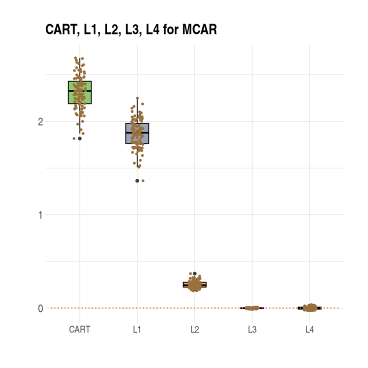


Figure: L_n, $L_{1,n}$, $L_{2,n}$, $L_{3,n}$ and $L_{4,n}$ boxplots when an MCAR mechanism was present in the data set, each point represents the mean value of a random forest.

Consistency of Random Forest’s Estimator

Proof of Theorem 1

Denote by $A_{s(n)}(\mathbf{X},\Theta)$ the cell of the tree built with the random variable Θ that contains $\mathbf{X}$, where $s(n)$ is the number of cuts necessary to construct the cell.

Let be

$$m_n(\mathbf{X},\Theta) = \frac{1}{\widehat{N}(A_{s(n)}(\mathbf{X},\Theta))}\sum_{i=1}^{n} Y_i \mathbb{1}_{\widehat{\mathbf{X}}_i \in A_{s(n)}(\mathbf{X},\Theta)}$$

our tree estimator.

Define another tree estimator that takes our partition of the input space $\mathcal{X}$ but considers the values $\mathbf{X}_i$ for the prediction,

$$m'_n(\mathbf{X}, \Theta) = \frac{1}{N(A_{s(n)}(\mathbf{X}, \Theta))} \sum_{i=1}^{n} Y_i \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X},\Theta)}$$

Note that $m'_n(\mathbf{X}) = \sum_{i=1}^{n} W_{n,i}(\mathbf{X}, \Theta) Y_i$, where

$$W_{n,i}(\mathbf{X}, \Theta) = \frac{1}{N(A_{s(n)}(\mathbf{X}, \Theta))} \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X},\Theta)}$$

$$\begin{aligned}
\mathbb{E}[m_n'(\mathbf{X}) - m(\mathbf{X})]^2 &= \mathbb{E}\left[\sum_{i=1}^{n} W_{n,i}(\mathbf{X}) Y_i - m(\mathbf{X})\right]^2 \\
&\leq 2\mathbb{E}\left[\sum_{i=1}^{n} W_{n,i}(\mathbf{X})(Y_i - m(\mathbf{X}_i))\right]^2 \\
&\qquad + 2\mathbb{E}\left[\sum_{i=1}^{n} W_{n,i}(\mathbf{X})(m(\mathbf{X}_i) - m(\mathbf{X}))\right]^2 \\
&= 2I_n + 2J_n
\end{aligned}$$

$$\begin{aligned}
I_n &= \mathbb{E}\left[\sum_{i=1}^{n} W_{n,i}(\mathbf{X})(Y_i - m(\mathbf{X}_i))\right]^2 \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} \mathbb{E}\left[W_{n,i}(\mathbf{X})W_{n,j}(\mathbf{X})(Y_i - m(\mathbf{X}_i))(Y_j - m(\mathbf{X}_j))\right] \\
&= \mathbb{E}\left[\sum_{i=1}^{n} W_{n,i}(\mathbf{X})^2 \varepsilon_i^2\right]
\end{aligned}$$

Note that

$$\begin{aligned}
\sum_{i=1}^{n} W_{n,i}(\mathbf{X})^2 \varepsilon_i^2 &= \frac{1}{N(A_{s(n)}(\mathbf{X}))}\left(\frac{1}{N(A_{s(n)}(\mathbf{X}))}\sum_{i=1}^{n} \varepsilon_i^2 \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X})}\right) \\
&\leq \frac{1}{q_n}\left(\frac{1}{N(A_{s(n)}(\mathbf{X}))}\sum_{i=1}^{n} \varepsilon_i^2 \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X})}\right)
\end{aligned}$$

For any $\alpha > 0$ there exists $N \in \mathbb{N}^\star$ such that for all $n \geq N$

$$\frac{1}{N(A_{s(n)}(\mathbf{X}))} \sum_{i=1}^{n} \varepsilon_i^2 \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X})} \leq \sigma^2 + \alpha$$

Hence, because $N(A_{s(n)}(\mathbf{X})) \geq q_n$ and $q_n \to \infty$ by assumption, we conclude that for n sufficiently large $I_n \leq \mathbb{E}\left[\frac{1}{q_n}(\sigma^2 + \alpha)\right] \leq \xi$ for any $\xi > 0$.

Applying Cauchy-Schwartz to J_n, we have

$$J_n \leq \mathbb{E}\left[\sum_{i=1}^{n} W_{n,i}(\mathbf{X})(m(\mathbf{X}_i) - m(\mathbf{X}))^2 \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X})}\right]$$

Note that $(m(\mathbf{X}_i) - m(\mathbf{X}))^2 \mathbb{1}_{\mathbf{X}_i \in A_{s(n)}(\mathbf{X})} \leq \Delta(m, A_{s(n)}(\mathbf{X}))^2$, hence

$$J_n \leq \mathbb{E}\left[\Delta(m, A_{s(n)}(\mathbf{X}))^2\right]$$

Suppose that $\Delta(m, A_{s(n)}(\mathbf{X}, \Theta)) \to 0$, almost surely.

Since $\Delta(m, A_{s(n)}(\mathbf{X}, \Theta)) \leq \Delta(m, [0,1]^p) < \infty$, we can use the dominated convergence theorem. Thus,

$$J_n \leq \mathbb{E}\left[\Delta(m, A_{s(n)}(\mathbf{X}, \Theta))^2\right] \to 0$$

Proposition 1

Assume that H1 and H2 hold. Then,

$$\Delta(m, A_{s(n)}(\mathbf{X}, \Theta)) \to 0 \quad \text{almost surely.}$$

Proof of Proposition 1

Let $\mathcal{W}_{\mathcal{Y}}$ be the collection of functions from $\mathcal{Y}$ to $[0,1]$, we define the "imputed" random variable as

$$\widehat{\mathbf{X}} = (\widehat{\mathbf{X}}^{(1)}, \ldots, \widehat{\mathbf{X}}^{(p)})$$

where

$$\widehat{\mathbf{X}}^{(h)} | \widehat{\mathbf{X}} \in A = \begin{cases} \mathbf{X}^{(h)} | \mathbf{X} \in A & \text{if } \mathbf{M}^{(h)} = 0 \\ \mathbf{B}^{(h)} & \text{if } \mathbf{M}^{(h)} = 1 \end{cases}$$

and

$$\mathbf{B}^{(h)} = \begin{cases} \left(a^{(h)}, z\right) & \text{if } \mathrm{Ber}(w(Y)) = 1 \\ \left(z, b^{(h)}\right) & \text{if } \mathrm{Ber}(w(Y)) = 0 \end{cases}, \quad w \in \mathcal{W}_{\mathcal{Y}}$$

That is, $w(Y)$ is the probability that $\widehat{\mathbf{X}}^{(h)} < z$ conditional to $\mathbf{M}^{(h)} = 1$ and Y.

We define the theoretical CART over a cut (h, z) and a function $w \in \mathcal{W}_{\mathcal{Y}}$ as

$$\begin{aligned} L^{\star}(h, z, w) =& \mathbb{V}[Y|\widehat{\mathbf{X}} \in A] - \mathbb{V}[Y|\widehat{\mathbf{X}}^{(h)} < z, \widehat{\mathbf{X}} \in A]\mathbb{P}[\widehat{\mathbf{X}}^{(h)} < z|\widehat{\mathbf{X}} \in A] \\ & - \mathbb{V}[Y|\widehat{\mathbf{X}}^{(h)} \geq z, \widehat{\mathbf{X}} \in A]\mathbb{P}[\widehat{\mathbf{X}}^{(h)} \geq z|\widehat{\mathbf{X}} \in A] \end{aligned}$$

The best cut and assignation $(h^{\star}, z^{\star}, w^{\star})$ is selected by maximizing $L^{\star}(h, z, w)$ over $\mathcal{M}_{try}$, $\mathcal{C}_A$ and $\mathcal{W}_{\mathcal{Y}}$, that is

$$(h^{\star}, z^{\star}, w^{\star}) \in \underset{\substack{h \in \mathcal{M}_{try} \\ (h,z) \in \mathcal{C}_A \\ w \in \mathcal{W}_{\mathcal{Y}}}}{\arg\max}\, L^{\star}(h, z, w)$$

Lemma 1

Assume that H1 and H2 are satisfied and fix $\mathbf{x} \in [0,1]^p$. Then for all $\rho, \xi > 0$, there exists $N \in \mathbb{N}^\star$ such that, for all $n \geq N$ if $\mathbb{P}\left[L^\star\left(A_{s(n)}, d^\star_{s(n)}, w^\star_{s(n)}\right) \leq \xi\right] \geq 1 - \rho$, then

$$\Delta(m, A_{s(n)}(\mathbf{x})) \to 0 \quad \text{almost surely.}$$

Lemma 2

Assume that H1 and H2 are satisfied and fix $\mathbf{x} \in [0,1]^p$. Then for all $\rho, \xi > 0$, there exists $N \in \mathbb{N}^\star$ such that, for all $n \geq N$

$$\mathbb{P}\left[L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \leq \xi\right] \geq 1 - \rho$$

We prove the almost sure convergence of $\Delta(m, A_{s(n)})$ to 0 by showing that the theoretical CART criterion of the sequence $(A_{s(n)})_n$ tends to 0 and use of Lemmas 1 and 2. Note that

$$\begin{aligned}
&L^{\star}\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) - L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right)\\
&= L^{\star}\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) - L_n\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right)\\
&\quad + L_n\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) - L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right)\\
&\leq L^{\star}\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) - L_n\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right)
\end{aligned}$$

Where the last inequality comes from noting that $L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \geq L_n\left(A_{s(n)}, d, w\right)$ for all cut $d \in \mathcal{C}_A$ and assignation $w \in \mathcal{W}_A$.

1. For a cell A, fix a cut $(h, z) \in \mathcal{C}_A$ and consider a function $w \in \mathcal{W}_{\mathcal{Y}}$.
2. Create a random vector W of dimension $\widehat{N}_{miss}(A) = \mathrm{Card}(\mathbf{i}_{A,miss}^{(h)})$

$$W_k = \mathrm{Ber}(w(Y_{j_k}))$$

   for $j_k \in \mathbf{i}_{A,miss}^{(h)}$.
3. Assign the observations $\widehat{\mathbf{X}}_{j_k}$ according to the random vector W.
4. Evaluate the empirical CART criterion L_n considering these assignations.

By strong law of large numbers

$$L^{\star}\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) - L_n\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) \to 0$$

almost surely.

Fix $\xi, \rho > 0$, for n sufficiently large, we have

$$L^{\star}\left(A_{s(n)}, d^{\star}_{s(n)}, w^{\star}_{s(n)}\right) - L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \leq \xi$$

almost surely.

On the other hand, by Lemma 2, there exists N_1 such that for all $n \geq N_1$, with probability at least $1 - \rho$

$$L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \leq C\xi$$

Hence, with the same probability,

$$L^\star\left(A_{s(n)}, d^\star_{s(n)}, w^\star_{s(n)}\right) \leq \xi$$

And by Lemma 1, we conclude that

$$\Delta(m, A_{s(n)}(\mathbf{X}, \Theta)) \xrightarrow{a.s.} 0$$

Sketch of the Proof of Lemma 1

Let be $\tilde{w} = \mathbb{P}\left[a^{(h)} \leq \mathbf{X}^{(h)} < z | \mathbf{x} \in A, \mathbf{M}^{(h)} = 1\right]$.

- Assume that there exists $\mathbf{x}, \mathbf{y}$ belonging to all the cells $A_{s(\phi(n))}$ s.t. $|m(\mathbf{x}) - m(\mathbf{y})| > c/2$.
- By hypothesis, $L^\star(A_{s(\psi \circ \phi(n))}, d, \tilde{w}) \to 0$ in probability.
- $\sup_{d \in \mathcal{C}_{A_{s(n)}}} L^\star(C_n, d, \tilde{w}) \to 0$ a.s. where $C_i = \cap_{k=1}^{i} A_{s(k)}$ (Technical Lemma 2).
- $|L^\star(C_i, d, \tilde{w}) - L^\star(C_\infty, d, \tilde{w})| \leq \varepsilon$, hence $L^\star(C_\infty, d, \tilde{w}) = 0$.
- Because $L^\star(C_\infty, d, \tilde{w}) = 0$, then m is constant in C_∞ (Technical Lemma 1), which contradicts $|m(\mathbf{x}) - m(\mathbf{y})| > c/2$.

Sketch of the Proof of Lemma 2

- Assume that there exists $c > 0$, $0 < p_0 < 1$, such that
$$L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) > c,$$
with probability at least p_0.
- For k sufficiently large,
$$\left|L_n\left(A_k, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) - L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right)\right| \leq \xi$$
(Technical Lemma 3).
- Therefore, with probability at least p_0, we have
$$c - \xi \leq L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) - \xi \leq L_n\left(A_k, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right).$$

$$\begin{aligned} L_n\left(A_k, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) &\leq \sup_{\substack{d \in \mathcal{C}_{A_k} \cap \mathcal{C}_{A_{s(n)}} \\ w \in \mathcal{W}_{A_k}}} L_n\left(A_k, d, w\right) \\ &\leq \sup_{\substack{d \in \mathcal{C}_{A_k} \\ w \in \mathcal{W}_{A_k}}} L_n\left(A_k, d, w\right) \leq \xi. \end{aligned}$$

Thus,

$$c - \xi \leq L_n\left(A_k, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \leq \xi$$

which is a contradiction. Therefore

$$L_n\left(A_{s(n)}, \widehat{d}_{s(n)}, \widehat{w}_{s(n)}\right) \leq \xi.$$