Minimal Learning Machine for datasets with missing values

Diego Parente P. Mesquita, Joao Paulo P. Gomes and Amauri H. Souza Junir

Department of Computer Science
Federal University of Ceara
Fortaleza, Ceara, Brazil

Today

Summary

- Problem and notation
- The Minimal Learning Machine
    - Formulation
    - ESD for missing data
- Experiments
- Conclusions

Goals and contributions

- Introduce the Minimal Learning Machine (MLM) [a]
- Propose a MLM for missing valued datasets using ESD

[a] A. H. Souza Junior, F. Corona, Y. Miche, A. Lendasse, G. Barreto, and O. Simula, "Minimal learning machine: A new distance-based method for supervised learning", in IWANN'13, LNCS 7902, pp. 408-416, 2013.

The regression problem and notation

We are given

- Set of input points $X = \{\mathbf{x}_i\}_{i=1}^{N}, \quad \mathbf{x}_i \in \mathbb{R}^D$
- Set of output points $Y = \{\mathbf{y}_i\}_{i=1}^{N}, \quad \mathbf{y}_i \in \mathbb{R}^S$

We assume

- A continuous mapping between the input and the output space ($f : \mathcal{X} \to \mathcal{Y}$)

We want to estimate such a mapping using

$$\mathbf{Y} = f(\mathbf{X}) + \mathbf{R},$$

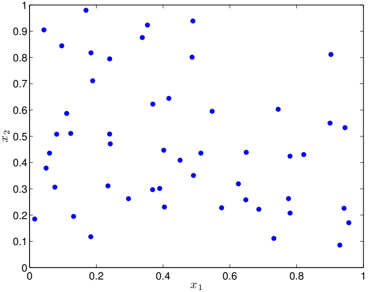
where rows of $\mathbf{X}$ and $\mathbf{Y}$ correspond to observations in the input and output space, and the matrix $\mathbf{R}$ contains the residual vectors.

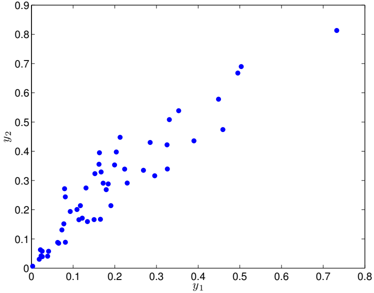
The Minimal Learning Machine

The Minimal Learning Machine algorithm can be decomposed into two main steps:

- Distance regression
- Output estimation

The Minimal Learning Machine





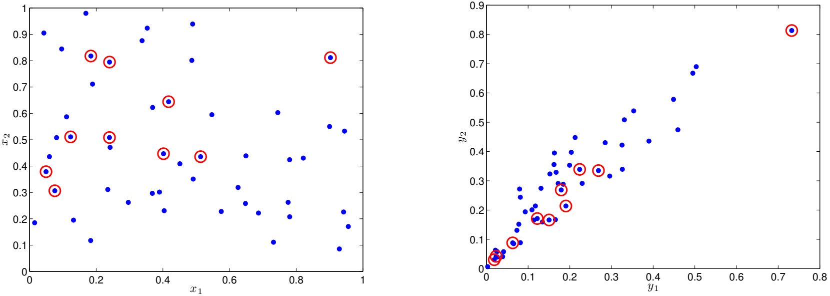
$\mathcal{X}$-space

- $X = \{\mathbf{x}_i\}_{i=1}^N$

$\mathcal{Y}$-space

- $Y = \{\mathbf{y}_i\}_{i=1}^N$

The Minimal Learning Machine



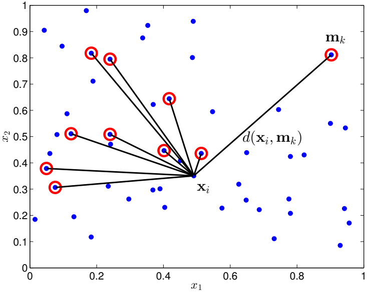
$\mathcal{X}$-space

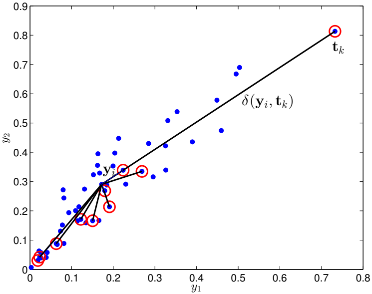
- $X = \{\mathbf{x}_i\}_{i=1}^{N}$
- $R = \{\mathbf{m}_k\}_{k=1}^{K}$

$\mathcal{Y}$-space

- $Y = \{\mathbf{y}_i\}_{i=1}^{N}$
- $T = \{\mathbf{t}_k\}_{k=1}^{K}$

The Minimal Learning Machine





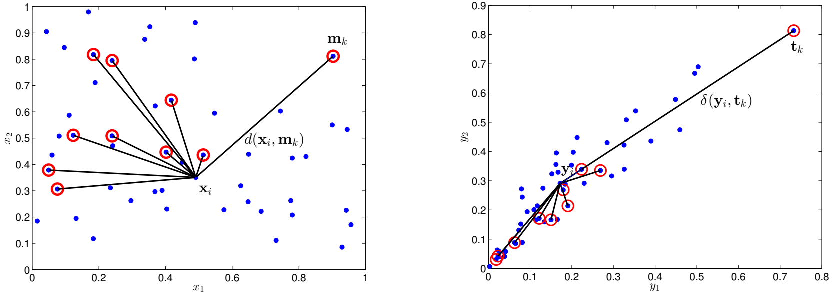
$\mathcal{X}$-space

- $X = \{\mathbf{x}_i\}_{i=1}^{N}$
- $R = \{\mathbf{m}_k\}_{k=1}^{K}$
- $\mathbf{D}_x(i,k) = [d(\mathbf{x}_i, \mathbf{m}_k)]$

$\mathcal{Y}$-space

- $Y = \{\mathbf{y}_i\}_{i=1}^{N}$
- $T = \{\mathbf{t}_k\}_{k=1}^{K}$
- $\mathbf{\Delta}_y(i,k) = [\delta(\mathbf{y}_i, \mathbf{t}_k)]$

The Minimal Learning Machine: Distance regression



We are interested in finding a mapping $g : \mathbb{R}^K \rightarrow \mathbb{R}^K$ such that

$$\mathbf{\Delta}_y = g(\mathbf{D}_x) + \mathbf{E}$$

The Minimal Learning Machine: Distance regression

Let's assume that the mapping g is linear, then

$$\mathbf{\Delta}_y = \mathbf{D}_x \mathbf{B} + \mathbf{E},$$

and the solution is given by $\hat{\mathbf{B}} = (\mathbf{D}_x^T \mathbf{D}_x)^{-1} \mathbf{D}_x^T \mathbf{\Delta}_y$.

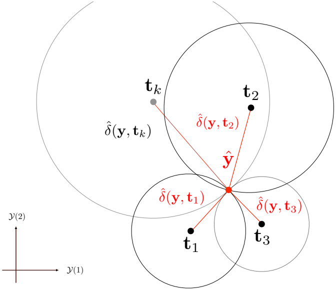
For a test point $\mathbf{x}$:
Collect the distances from the K reference points in the vector $\mathbf{d}(\mathbf{x}, R) = [d(\mathbf{x}, \mathbf{m}_1), \ldots, d(\mathbf{x}, \mathbf{m}_K)]$, then estimate distances in $\mathcal{Y}$

$$\hat{\boldsymbol{\delta}}(\mathbf{y}, T) = \mathbf{d}(\mathbf{x}, R)\hat{\mathbf{B}}.$$

The Minimal Learning Machine: Output estimation

For finding an estimate to the output $\mathbf{y}$, we need to solve

$$(\mathbf{y}-\mathbf{t}_k)^T(\mathbf{y}-\mathbf{t}_k)=\hat{\delta}^2(\mathbf{y},\mathbf{t}_k), \quad \forall k=1,\ldots,K$$



It can be formulated as an optimization problem

$$\hat{\mathbf{y}}=\underset{\mathbf{y}}{\operatorname{argmin}}\sum_{k=1}^{K}\left((\mathbf{y}-\mathbf{t}_k)^T(\mathbf{y}-\mathbf{t}_k)-\hat{\delta}^2(\mathbf{y},\mathbf{t}_k)\right)^2$$

The Expected Squared Distance (ESD)

- Estimate the squared distance of two vectors in the presence of missing data
- $\alpha, \beta \in \mathbb{R}^D$ drawn from a same multivariate probability distribution, but possibly with deleted entries

$$E[\|\alpha - \beta\|_2^2] = \sum_{i=1}^{D} E[(\alpha_i - \beta_i)^2]$$

The Expected Squared Distance (ESD)

$$E[\|\alpha - \beta\|_2^2] = \sum_{i \notin M_\alpha \cup M_\beta} (\alpha_i - \beta_i)^2 + \sum_{i \in M_\alpha \setminus M_\beta} E[(\alpha_i - \beta_i)^2] + \sum_{i \in M_\beta \setminus M_\alpha} E[(\alpha_i - \beta_i)^2] + \sum_{i \in M_\alpha \cap M_\beta} E[(\alpha_i - \beta_i)^2]$$

For entries $i \in M_\alpha \setminus M_\beta$, we have:

$$\begin{aligned} E[(\alpha_i - \beta_i)^2] &= E[\alpha_i^2 + \beta_i^2 - 2\alpha_i\beta_i] = E[\alpha_i^2] + \beta_i^2 - 2E[\alpha_i]\beta_i \\ &= E[\alpha_i^2] - E[\alpha_i]^2 + E[\alpha_i]^2 + \beta_i^2 - 2E[\alpha_i]\beta_i \\ &= (E[\alpha_i] - \beta_i)^2 + \mathrm{Var}[\alpha_i] \end{aligned}$$

Simulation Methodology

- MLM using different approaches for missing data
  - Drop entries
  - Input sample mean
  - EM
  - ESD

Simulation Methodology

- MLM using different approaches for missing data
  - Drop entries
  - Input sample mean
  - EM
  - ESD
- Six real-world datasets from UCI

Simulation Methodology

- MLM using different approaches for missing data
  - Drop entries
  - Input sample mean
  - EM
  - ESD
- Six real-world datasets from UCI
- Varying number of missing data

Datasets

Table: Datasets characteristics

Dataset	attributes	instances
Concrete compression	8	1030
Boston Housing	13	506
Servo	4	167
Stocks	9	950
Breast cancer	30	569
Wine	13	178

Results

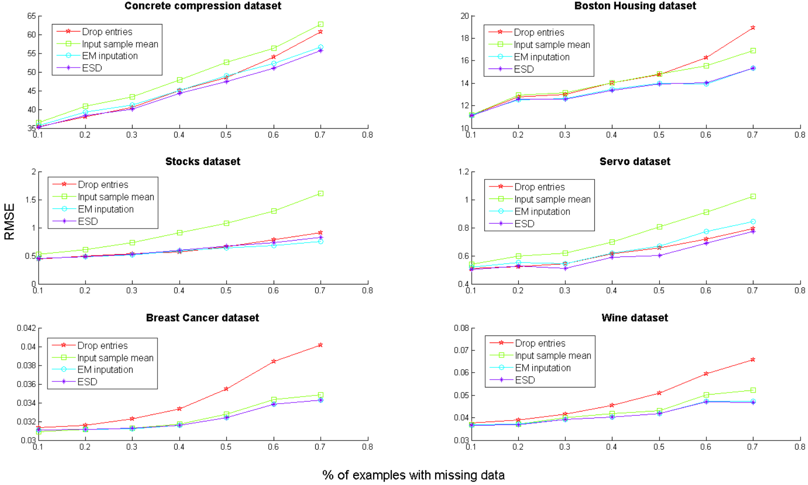


Figure: RMSE for all datasets

Conclusions

- We have introduced a variant of MLM capable of handling missing data

Conclusions

- We have introduced a variant of MLM capable of handling missing data
- The proposed approach achieved the best results when compared to other strategies for missing data

THANK YOU!!!