Impact of data preprocessing methods on the performance of machine learning models

Anna Kozak, Hubert Ruczyński, Maksymilian Tabian

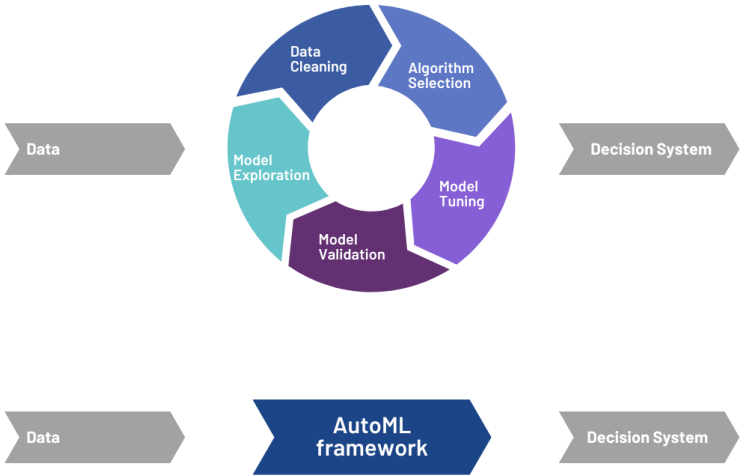
Faculty of Mathematics and Information Science
Warsaw University of Technology

Katowice, 24.05.2025

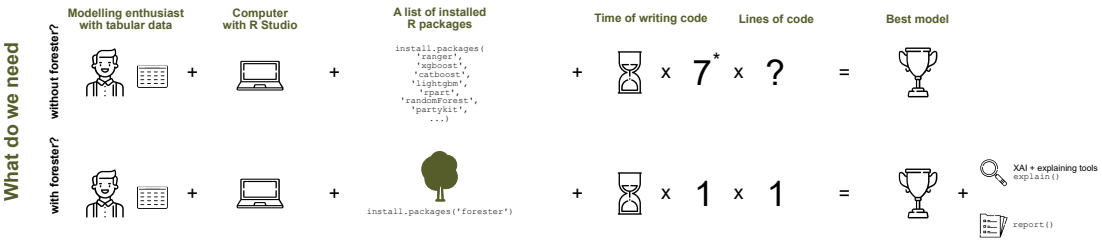
Introduction

Impact of **data preprocessing** methods
on the **performance** of machine learning models.

Motivation – AutoML System



AutoML framework in R - forester[1]



[1] Kozak, A., Ruczyński, H. (2023) *forester: A Novel Approach to Accessible and Interpretable AutoML for Tree-Based Modeling*

Experiment Setup

For binary and multiclass classification tasks, we utilized datasets from the OpenML-CC18 benchmark[2], whereas the regression tasks were sourced from OpenML[3].

- 7 regression
- 10 binary classification
- 8 multiclass classification

Each data frame was split into training, testing, and validation samples, with the proportions 60%, 20%, and 20%, respectively.

We use five tree-based models: decision tree, random forest, XGBoost, LightGBM, and CatBoost.

[2] Bischl B., Casalicchio G., Feurer M., Hutter F., Lang M., Mantovani R., van Rijn J., Vanschoren J. (2019) *OpenML Benchmarking Suites*

[3] Vanschoren J., van Rijn J., Bischl B., Torgo L. (2013) *OpenML: networked science in machine learning*

Custom preprocessing

Heuristic removals

- duplicated columns
- id-like columns
- static columns
- sparse columns
- corrupted rows
- highly correlated columns

Data Imputation

- MICE
- median-other
- median-frequency
- KNN

Feature Selection

- mutual information
- Boruta
- MCFS
- variable importance

We use only the default parameters for each method.

Strategies

We aggregate the removal strategies into the following approaches:

- **Minimal** - it removes the observations that do not have the target value,
- **Medium** - it removes duplicate, id-like, static (threshold $= 0.99$), and sparse (threshold $= 0.3$) columns and corrupted rows with too many missing values (threshold $= 0.3$),
- **Maximal** - it is a medium approach with the removal of highly correlated columns.

To minimize the computational overhead, we decided to omit some presumably costly combinations, which resulted in carrying on with 38 different preprocessing strategies.

Model Training

$$\underset{\text{datasets}}{25} \times \underset{\text{strategies}}{38} \times \underset{\text{configurations}}{105}$$

Model Training

$$\underset{\text{datasets}}{25} \times \underset{\text{strategies}}{38} \times \underset{\text{configurations}}{105}$$

$$105\ \textit{configurations} = 5\ \textit{models} \times (\textit{default} + 20 \times \textit{Random Search})$$

Baselines

The **baseline preprocessing strategy** is a preprocessing strategy which consists of minimal removal, median-other imputation, and lack of feature selection.

The **baseline dataset** (B) for the dataset D is the dataset created from D with the **baseline preprocessing strategy**.

The **baseline model** (m_B) for the dataset D is the best performing model trained on the **baseline dataset**.

For the dataset D we compared the results of the best model of each preprocessing strategy for D ($\theta(m)$) to the performance of it's **baseline model** ($\theta(m_B)$).

Preprocessibility measure

Definition

The preprocessibility measure[a] is

$$P^{+}(D) = \max_{d_i \in D}\Big(\max_{m_j(d_i)}(\theta(m_j(d_i)))\Big) - \max_{m_j(B)}(\theta(m_j(B))), \tag{1}$$

$$P^{-}(D) = \min_{d_i \in D}\Big(\min_{m_j(d_i)}(\theta(m_j(d_i)))\Big) - \min_{m_j(B)}(\theta(m_j(B))), \tag{2}$$

where D is a set preprocessed datasets, $d_i \in D$ is a dataset from D, θ is the performance measurement metric, $m_j(d_i)$ is the model trained on d_i dataset, and B is a baseline dataset for D.

[a] Probst, P., Boulesteix, A., Bischl, B. (2019) *Tunability: Importance of Hyperparameters of Machine Learning Algorithms*

The impact of preprocessing methods

Statistic	All strategies	Is FS Used?		Feature Selection Methods				Removal Strategy			Imputation Method			
		No	Yes	Boruta	MCFS	MI	VI	Minimal	Medium	Maximal	MICE	Median-other	Median-frequency	KNN
Wins [%]	15.5%	12.3%	17.3%	22.5%	8.0%	14.4%	22.7%	10.0%	13.3%	13.0%	29.2%	29.2%	27.5%	58.3%
Ties [%]	56.6%	70.6%	48.5%	53.0%	76.0%	40.4%	36.0%	85.0%	71.4%	55.0%	12.5%	41.6%	40.0%	25.0%
Losses [%]	27.9%	17.1%	34.2%	24.5%	16.0%	45.2%	41.3%	5.0%	15.3%	32.0%	58.3%	29.2%	32.5%	16.7%
Average positive preprocessibility	0.009	0.006	0.009	0.008	0.001	0.008	0.003	0.005	0.005	0.005	0.005	0.004	0.002	0.017
Average negative preprocessibility	-0.048	-0.013	-0.047	-0.013	-0.010	-0.044	-0.021	-0.002	-0.003	-0.011	-0.021	-0.019	-0.016	-0.011

Table: The impact of preprocessing methods on the tree-based models predictive quality.

The impact of preprocessing methods

Statistic	All preprocessing strategies					Best preprocessing strategies			
	Decision tree	Random forest	XGBoost	LightGBM	CatBoost	XGBoost	CatBoost	Random forest	All
Wins [%]	13.7%	18.2%	17.0%	10.7%	22.0%	18.0%	36.0%	36.0%	30.0%
Ties [%]	60.6%	52.5%	55.5%	62.8%	50.5%	64.0%	52.0%	58.0%	58.0%
Loses [%]	25.7%	29.3%	27.5%	26.5%	27.5%	18.0%	12.0%	6.0%	12.0%
Average positive preprocessibility	0.007	0.014	0.010	0.004	0.010	0.006	0.007	0.008	0.007
Average negative preprocessibility	-0.063	-0.039	-0.043	-0.066	-0.070	-0.001	-0.001	-0.001	-0.001
Average maximal score (Accuracy/R^2)	0.752	0.694	0.845	0.795	0.866	0.840	0.862	0.688	0.797

Table: The behaviour of different tree-based models on preprocessing pipelines and the validation of best strategy.

Fuzzy methods of data preprocessing

We consider two types of fuzzy methods of data preprocessing

- data imputation
- feature selection

We selected and implemented one data imputation method[4] and one feature selection method[5].

[4] Nikfalazar, S., Yeh, CH., Bedingfield, S. et al. Missing data imputation using decision trees and fuzzy clustering with iterative learning. Knowl Inf Syst 62, 2419–2437 (2020). https://doi.org/10.1007/s10115-019-01427-1

[5] Rezaee, M. R., Goedhart, B., Lelieveldt, B. P., Reiber, J. H. (1999). Fuzzy feature selection. Pattern Recognition, 32(12), 2011-2019.

Learning Curves[6]

Learning curves can and have been used to address common AutoML problem types.

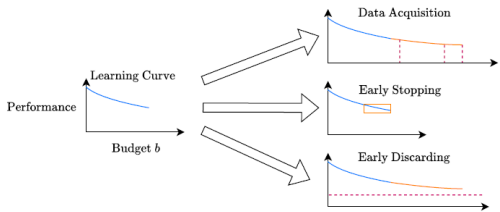
- Algorithm Selection Problem
- Hyperparameter Optimization Problem (HPO)
- Combined Algorithm Selection and Hyperparameter Optimization Problem (CASH)
- Neural Architecture Search (NAS)
- Few Shot Learning

[6] Mohr, F., Rijn, J.N.V. (2022). Learning Curves for Decision Making in Supervised Machine Learning: A Survey. arXiv preprint arXiv:2201.12150.

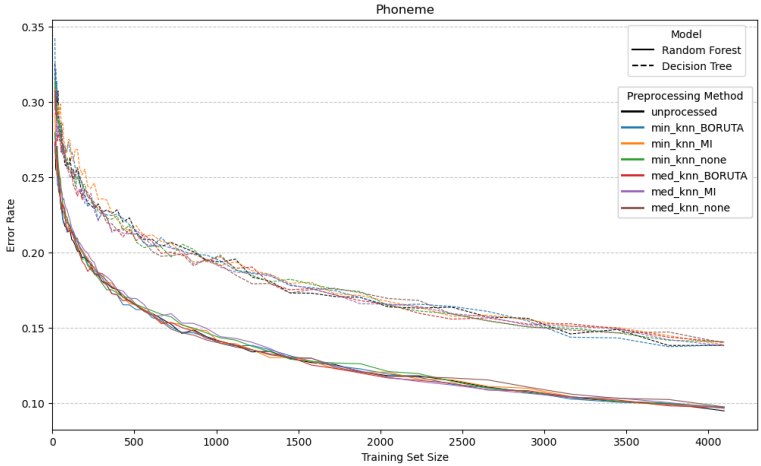
Learning Curves

There are at least three situations in which learning curves aid decision making:

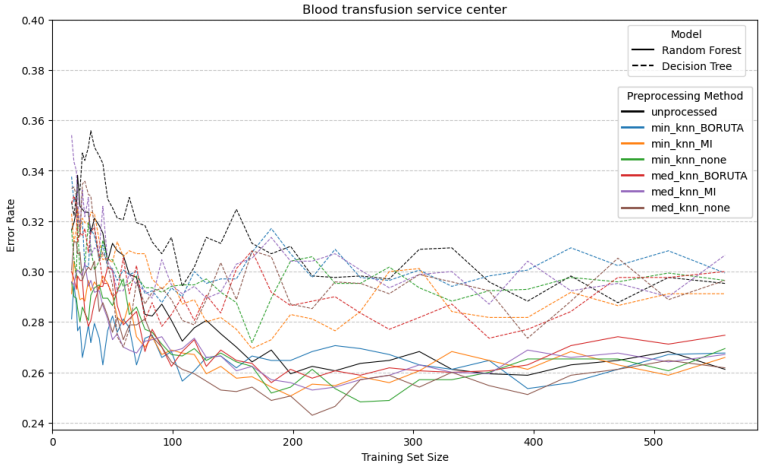
1. **Data Acquisition** The acquisition of how mamy additional labels is (economically) reasonable?
2. **Early Stopping** Stop model training as soon as limit/saturation performance is reached.
3. **Early Discarding** Stop model training as soon as it can be recognized that the limit/saturation performance will not be at least τ.



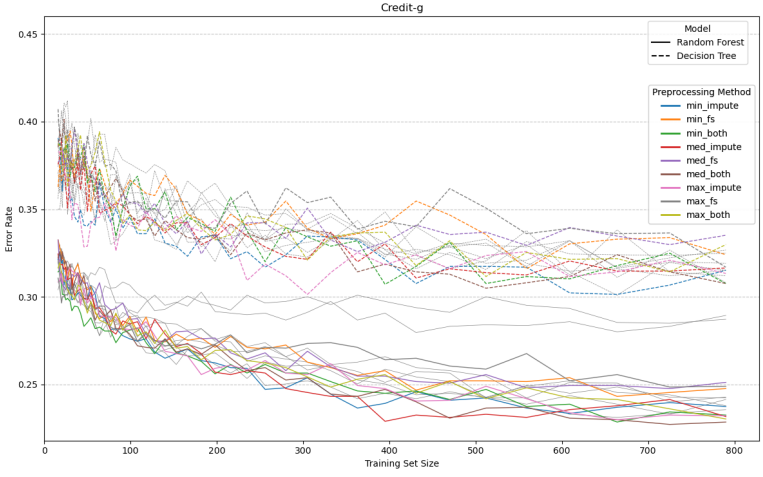
Results for classic data preprocessing methods



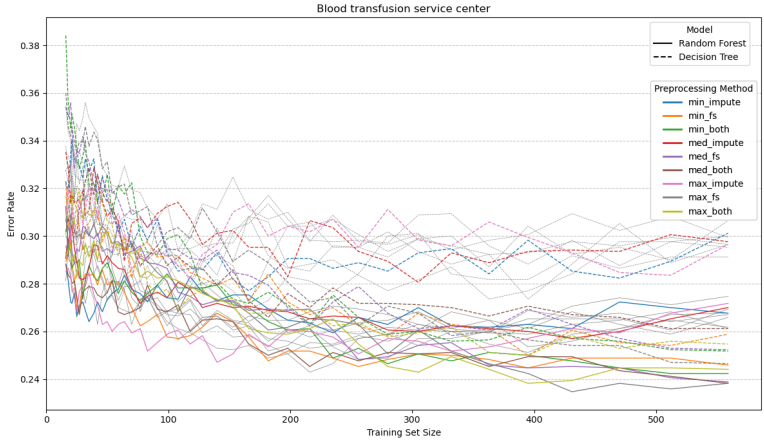
Results for classic data preprocessing methods



Results for fuzzy data preprocessing methods



Results for fuzzy data preprocessing methods



Acknowledgements

Implementation of *forester* framework[7]: **Hubert Ruczyński, Adrianna Grudzień, Patryk Słowakiewicz**.
Experiments related to classical data preprocessing methods: **Hubert Ruczyński**.
Implementation[8] of fuzzy data preprocessing methods: **Antoni Zajko**.
Implementation[9] of learning curves and experiments: **Maksymilian Tabian**.

Faculty of Mathematics and Information Science, Warsaw University of Technology

[7] https://github.com/ModelOriented/forester
[8] https://github.com/azoz01/fuzzy_methods
[9] https://github.com/makstab12/LC-tree-based-models

Thank you!

References


Ruczyński H., Kozak A., *Do Tree-based Models Need Data Preprocessing?*, AutoML Conference 2024 (Workshop Papers), 2024.


Kozak A., Ruczyński H., *forester: A Novel Approach to Accessible and Interpretable AutoML for Tree-Based Modeling*, AutoML Conference 2023 (Workshop Papers), 2023.


Vanschoren J., van Rijn J., Bischl B., Torgo L., *OpenML: networked science in machine learning*, SIGKDD Explorations 15(2), 2013.


Bischl B., Casalicchio G., Feurer M., Hutter F., Lang M., Mantovani R., van Rijn J., Vanschoren J., *OpenML Benchmarking Suites*, 2019.


Probst, P., Boulesteix, A., Bischl, B., *Tunability: Importance of Hyperparameters of Machine Learning Algorithms*, Journal of Machine Learning Research, 2019.


Moiseeva, T., Ledeneva, T., Missing Data Imputation Using Fuzzy System, 4th International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA), 2022.

References

- M. Ramze Rezaee, B. Goedhart, B.P.F. Lelieveldt, J.H.C. Reiber, Fuzzy feature selection, Pattern Recognition, 1999.
- Nikfalazar, S., Yeh, CH., Bedingfield, S. et al. Missing data imputation using decision trees and fuzzy clustering with iterative learning. Knowledge and Information Systems, 2020.
- Rengasamy, D., Mase, J. M., Kumar, A., Rothwell, B., Torres, M. T., Alexander, M. R., Winkler, D. A., Figueredo, G. P., Feature importance in machine learning models: A fuzzy information fusion approach. Neurocomputing, 2022.
- Mohr, F., Rijn, J.N.V. Learning Curves for Decision Making in Supervised Machine Learning: A Survey, 2022.
- Viering, T., Loog, M., The Shape of Learning Curves: A Review, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023.
- Mohr, F., Viering, T.J., Loog, M., van Rijn, J.N., LCDB 1.0: An Extensive Learning Curves Database for Classification Tasks, Machine Learning and Knowledge Discovery in Databases, ECML PKDD 2022.