Untitled

Creator **Olusanmi Hundogan** Created **Mar 25, 2024, 11:02** Last updated **Apr 25, 2024, 06:28**

Files

- experiment_preprocessors.py
- analyze_preprocessors.py

Motivation

The Meta model preprocesses the data before training and running the model. The way the preprocessing is done can vary. This experiment aims to investigate different preprocessing options and their impact on the model. Note, that this experiment does not relate to individual preprocessing procedures on the features or feature type but preprocessing from a macroscopic view. In particular, we investigated the following preprocessing variants:

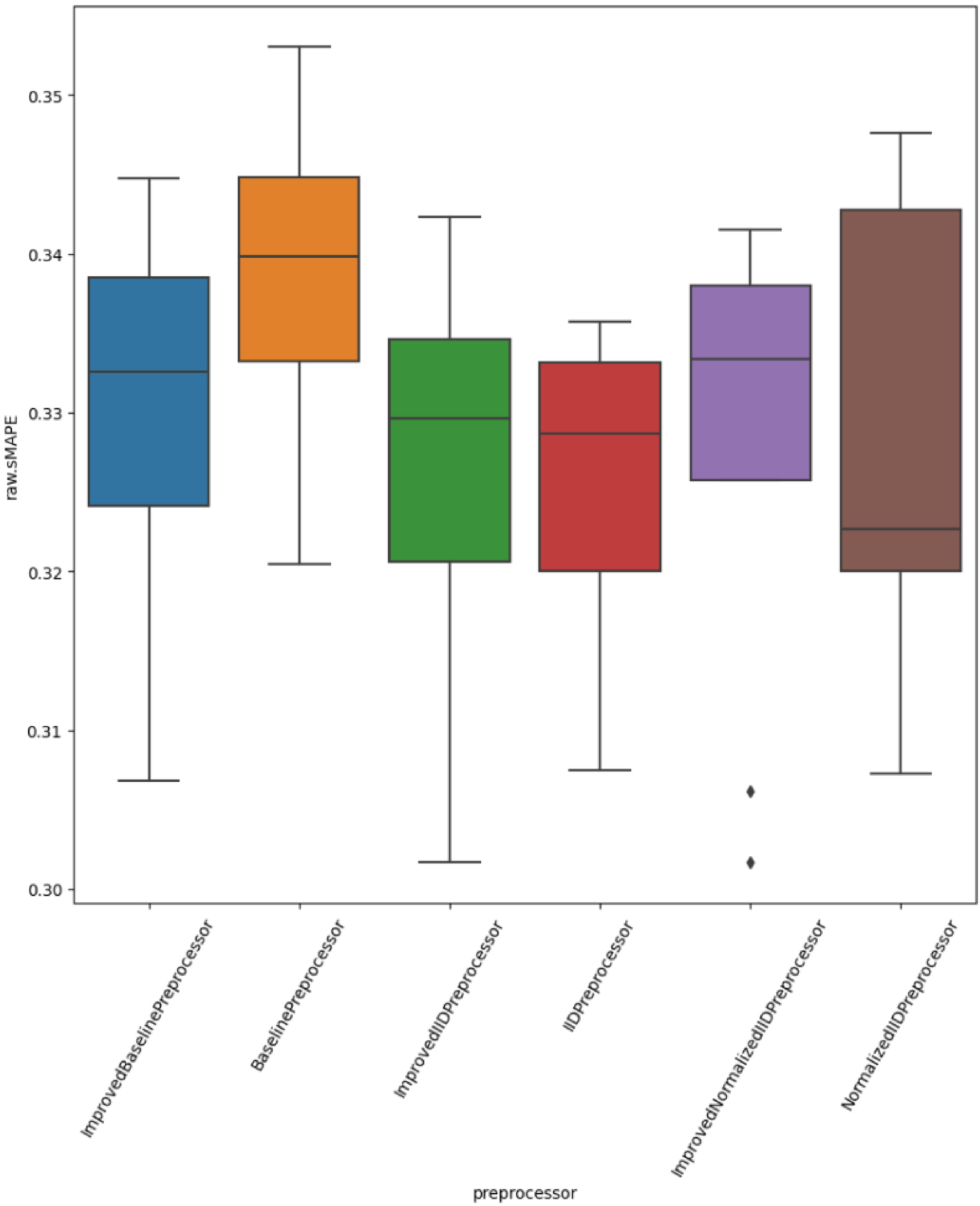
- BaselinePreprocessor: Standard preprocessing where we split features into numeric and categorical, apply feature transformations like imputations, scaling and encoding on each type and recombine the data.
- IIDPreprocessor: Follows the baseline procedure but applies an additional standard scaling on top all features. This will affect both numeric and categorical data.
- NormalizedIIDPreprocessor: Same as IIDPreprocessor but instead of standard scaling it uses min max scaling.
- ImprovedBaselinePreprocessor: Applies the same procedure as BaselinePreprocessor but while the former normalizes only the financial columns, the improved version applies the normalisation on all columns (numeric and categorical-encoded).
- ImprovedIIDPreprocessor: Improved version of the IIDPreprocessor
- ImprovedNormalizedIIDPreprocessor: Improved version of the NormalizedIIDPreprocessor

Design

For this experiment, we train an MMA model for every preprocessor variant and evaluate the sMAPEvalue on scope 1. Each preprocessor uses power transformation to scale the numerical variables. We run the experiment for 10 repetitions for each configuration. We run the experiments without dimensionality reduction.

Results and Insights

The plot below shows all the sMAPEvalues for the different PreprocessingTechniques The results show that the NormalizedIIDPreprocessor to have the lowest median sMAPEvalue.



However, the IIDPReprocessorappears to have lower sMAPEvalues for the 1st, 3rd and last quartiles. However, these differences seem negligibly low. The lowest min. sMAPEvalues are achieved with the improved versions. However, these also seem to have a higher spread in their results. Lastly, the BaselinePreprocessorfails on every quartile compared to the other methods.

preprocessor	mean	std	min	25%	50%	75%	max
BaselinePreprocessor	0.338294	0.010448	0.320462	0.333232	0.339824	0.344834	0.353016
IIDPreprocessor	0.324961	0.010113	0.307447	0.320018	0.328645	0.333125	0.335740
ImprovedBaselinePreprocessor	0.329845	0.011504	0.306810	0.324160	0.332535	0.338512	0.344743
ImprovedIIDPreprocessor	0.327072	0.012475	0.301711	0.320594	0.329614	0.334627	0.342353
ImprovedNormalizedIIDPreprocessor	0.328127	0.013818	0.301660	0.325773	0.333397	0.338028	0.341518
NormalizedIIDPreprocessor	0.327988	0.015025	0.307271	0.320055	0.322652	0.342759	0.347578

Decision

Update 25.04.24

We decide to either the IIDPreprocessorbecause it appears to display the best results while remaining consistent.