Automating the selection of preprocessing techniques for deep neural networks

Student: Marcus September

Supervisor: Francesco Sanna Passino

Department of Mathematics, Imperial College London

marcus.september22@imperial.ac.uk

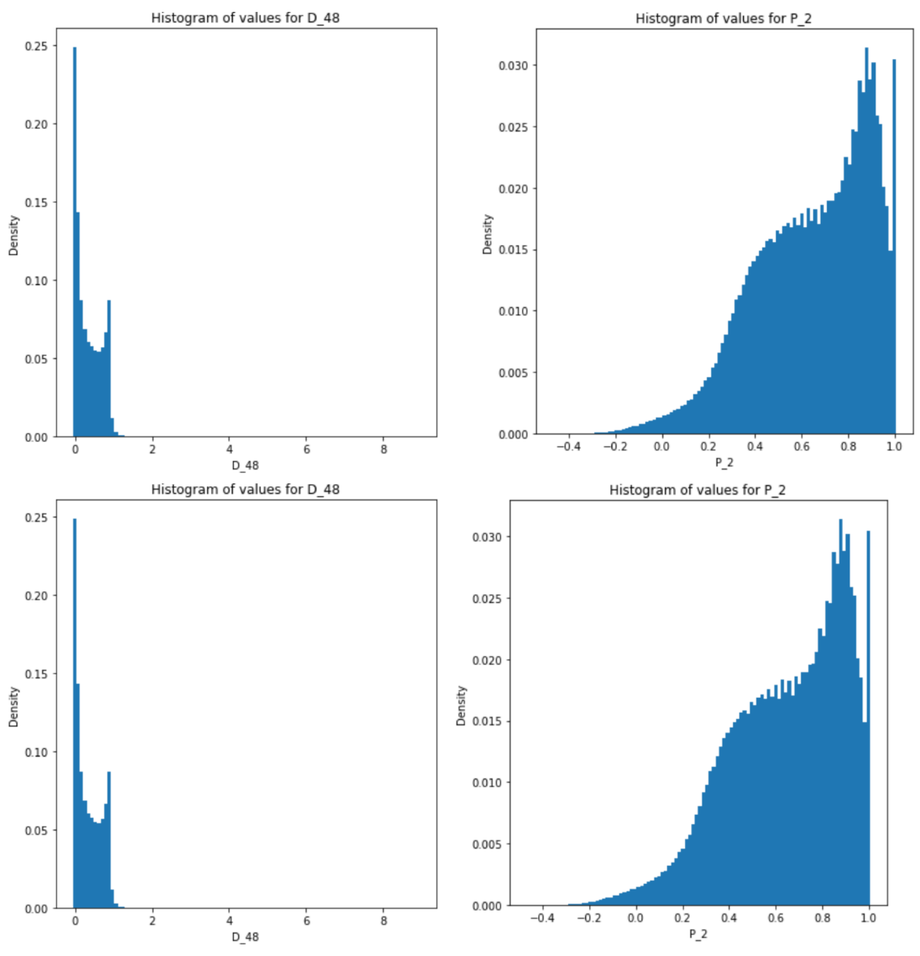
1. Problem and motivation

Deep learning sequence models, such as Recurrent Neural Networks (RNNs) and transformers, are sensitive to input variable distributions. Both training speed and performance can drop significantly for non-normal distributions, such as skewed distributions and those with outliers. Real-world data also usually contains a lot of missing values, which needs to be converted to numeric values for use in deep learning models. Deciding on appropriate preprocessing methods and how to handle missing values is essential in optimising model performance. However, this is a time-consuming process. The project aims to automate this by automatically selecting the preprocessing methods to use for any given sequence dataset, in order to optimise performance and training efficiency.

2. Data

Dataset consists of $N \approx 460000$ customers, each with $P = 188$ aggregated profile features recorded at each of the $T = 13$ sequential statement dates. Together, the dataset forms N instances of a P-dimensional multivariate time-series of length T. For each multivariate time-series, the target label $y \in \{0, 1\}$ indicates whether the customer defaulted on their loan or not. The task is to predict the probability $\mathbb{P}(Y = 1)$. Note that due to privacy concerns, all the dataset features have been anonymized. TODO: say something about anonymised with random noise.

3. Exploratory data analysis



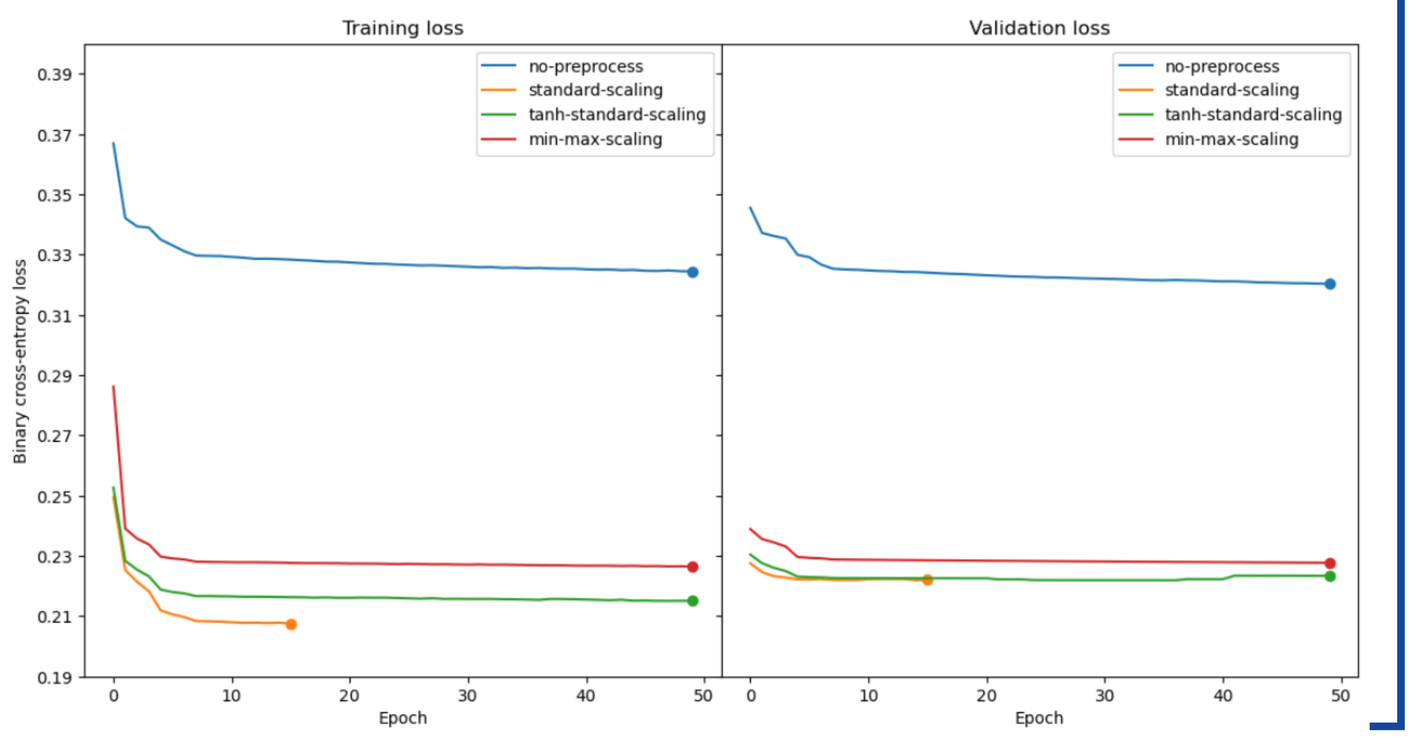
A lot of the variables have very skewed distributions or extreme values. There are also many missing values.

4. Synthetic data

- With the synthetic data generation procedure I have proposed, I can specify arbitrary unnormalized probability density functions (PDFs) for each of the P variables, from which the inverse cumulative density functions, $F_1^{-1}, \ldots, F_P^{-1}$ are inferred. Correlated uniform random variables $U \in [0, 1]^{T \times P}$ with a similar correlation structure as that of a multivariate time-series are then generated, and a response $y \in \{0, 1\}$ is formed from these. These uniform random variables are then transformed by $F_1^{-1}, \ldots, F_P^{-1}$ and returned to the user with the response, as samples from the provided PDFs.
- By being able to fully control the distribution of each variable, it will be easier to experiment and learn *when* each preprocessing method works best, and this insight can be used to form the heuristics used for automatically selection the appropriate techniques.

5. Preliminary result I: Preprocessing on real data

- Applying appropriate preprocessing can improve both the predictive performance and the training efficiency of deep sequence models (see figure 1a).
- Non-normalized data reduces the predictive power of deep sequence models (see figure 1b/table)

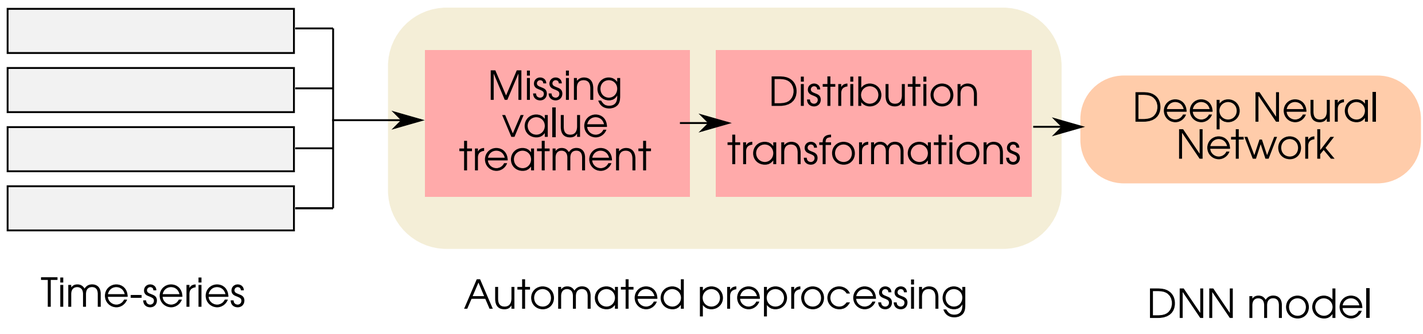


6. Preliminary result II: Using synthetic data

TODO jk: Talk next here about synthetic data results, and not able to learn to undo $F_1^{-1}, \ldots, F_P^{-1}$ transformations.

Dataset	Validation accuracy
Synthetic data with non-normal distributions	95.65%
Synthetic data with uniform distributions	98.65%

7. Automating the preprocessing procedure



To automate the selection of preprocessing techniques, heuristics based on each variable's distribution can be used. Another option is parameterizing the distribution transformations and incorporating it as part of the forward- and back-propagation of the training procedure. This ensures that the optimal transformations are learned automatically.

References

- TODO