

Week ML3 – Data Pre-Treatment & Baseline Models

H. Majeed, S. Bukhari, S. Gohar

November 24, 2025

1. DATA PRE-TREATMENT

1.1 Sampling Frequency and Continuity

The Dogecoin dataset consists of daily open, high, low, close, and volume values. Frequency is regular (intervals of 1 day), so no explicit resampling is needed.

In case there were irregularities, typical remedies would include forward/backward filling, linear interpolation, or resampling to a fixed daily grid to ensure that models such as AR and LSTM operate on evenly spaced timesteps.

1.2 Synchronization of Variables

All features have the same timestamps, so no alignment challenges.

For variables that are asynchronous, several methods for aligning features to a common timeline can be employed, including timestamp reindexing, interpolation, and merging by nearest timestamp.

1.3 Handling Missing Values

There are no missing values in this dataset.

If there were any missing observations, short gaps could be treated with either forward fill or interpolation, while longer runs of consecutive gaps might need moving-window imputation or removal. Care should be taken to avoid distorting volatility patterns.

1.4 Outlier Detection Using STL

STL decomposition helps to separate the trend, seasonality, and residual components. Residual spikes that do not correspond to market behavior can be marked as outliers. These may be winsorized, replaced with trend+seasonal estimates, or removed.

However, in cryptocurrency data, large fluctuations are usually real, so STL helps differentiate between structural changes and anomalies.

2. BASELINE AUTOREGRESSIVE MODEL

The baseline model provides a basis for comparison when evaluating LSTM and other deep learning architectures.

The Dogecoin auto-correlation plots indicate a strong lag-1 relation and slow decay in the ACF, implying that AR(1) or even AR(p) models are good baselines.

Procedure:

- Train on the first 70% of the data.
- Validate using the next 15%.
- Test on the final 15%.
- Evaluate using RMSE, MAE, and direction-accuracy metrics.

While the AR model captures short-term momentum, nonlinear surges in early 2021 are a challenge for it. Therefore, it is a suitable benchmark, since future RNN/LSTM models are expected to outperform it in terms of volatility handling and long-term structure.

3. LITERATURE REVIEW

3.1 Sub-Sequencing Long Time Series

Generally, long time series are divided into shorter, fixed-length sequences so that deep learning models can learn temporal patterns proficiently. The sliding-window approach is one of the most common; a window of length T is moved step-by-step across the dataset to generate multiple overlapping sequences (Hewamalage, Bergmeir & Bandara, 2021).

This method is particularly effective for cryptocurrency data because the local volatility and fast regime changes make long unsegmented sequences unsuitable for training.

3.2 How Seasonality Affects Sub-Sequencing

The concept of seasonality should be used to inform how the sequences are built. If data is periodic- in other words, containing repeating cycles-such as the weekly seasonality seen from our STL decomposition-then the window length should be long enough to capture one full seasonal cycle. This is to capture recurring structure in the window (Bandara, Bergmeir & Smyl, 2020).

While LSTMs are capable of identifying seasonality on their own, performance improves when seasonal patterns are stabilized or exposed by decomposition steps like STL.

3.3 Trends & Seasonality in LSTM Models

With their gated memory structure, LSTM networks are especially suitable for the modelling of nonlinear dynamics, trends, and recurring seasonal behaviour.

However, for highly noisy financial series like Dogecoin, studies show that performance increases when trends and seasonal components are pre-processed, reducing noise and increasing the signal.

Our pipeline follows the recommended practices of using STL for detecting weekly seasonality, removing anomalies, and stabilizing the trend.

3.4 Standardization Methods for LSTM

Since LSTMs are sensitive to the scaling of inputs, data should be normalized beforehand.

Common approaches include:

Min–Max Scaling

- Most widely used for cryptocurrency data
- It preserves relative changes and stabilizes volatile price movements.

Z-Score Standardization

- Suitable when the distributions are roughly stable

Robust Scaling

- Effective for crypto data with extreme outliers

Recent reviews reflect that due to its capability of compressing large fluctuations into a manageable range, Min–Max scaling is generally the best performing method for financial time series (Hewamalage et al., 2021). All standardization must be fitted only on the training set to avoid data leakage.

REFERENCES

Bandara, K., Bergmeir, C. and Smyl, S., 2020. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert systems with applications*, 140, p.112896.

Hewamalage, H., Bergmeir, C. and Bandara, K., 2021. Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37(1), pp.388-427.

Zeng, A., Chen, M., Zhang, L. and Xu, Q., 2023, June. Are transformers effective for time series forecasting?. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 37, No. 9, pp. 11121-11128).