Abstract

Bitcoin has attracted attention from different market participants due to unpredictable price patterns. Sometimes, the price has exhibited big jumps. Bitcoin prices have also had extreme, unexpected crashes. We test the predictive power of a wide range of determinants on bitcoins’ price direction under the continuous transfer entropy approach as a feature selection criterion. Accordingly, the statistically significant assets in the sense of permutation test on the nearest neighbour estimation of local transfer entropy are used as features or explanatory variables in a deep learning classification model to predict the price direction of bitcoin. The proposed variable selection methodology includes the NASDAQ index and Tesla as drivers. Under different scenarios and metrics, the best results are obtained using the significant drivers during the pandemic as validation. In the test, the accuracy increased in the post-pandemic scenario of July 2020 to January 2021 without drivers. In other words, our results indicate that in times of high volatility, Bitcoin seems to self-regulate and does not need additional drivers to indicate the accuracy of the price direction [1].

Data

01/Jan/2017 to 09/Jan/2021 at a daily frequency for a total of n = 1470 observations.

Transfer Entropy

Transfer Entropy (TE) measures the flow of information from system Y over system X in a non-parametric and nonlinear in nature. The idea is to model the time series as a Markovian system and incorporate the temporal dependencies of the destination. The local TE may be either positive or negative (with the source y(t) being either informative or misinformative respectively) for a specific event set \[ \{ x(t), x(t+1), \ldots, x(t+k) \} \] for each destination element X. If there is no deviation \[ \text{dev}(\beta) \] of the generalized Markov property then is measured: \[ p(x_{t+1}|x_t) = p(x_{t+1}|x_{t+k}) \] If there is no deviation \( Y \) has no influence on \( X \).

Local Transfer Entropy

TE metric can be formulated as a global average or expectation value of a local TE at each observation [3]:

\[
T_{Y \rightarrow X}(k, l) = \sum_{i} p(x_{t+1}, x_t, y_{t+1}^{(k)}|y_t^{(l)}) \log \frac{p(x_{t+1}|x_t, y_{t+1}^{(k)}|y_t^{(l)})}{p(x_{t+1}|x_t, y_t^{(l)})}.
\]

The measure is local in the sense it is defined at each time \( n \) for each destination element \( X \) in the system and each causal information source \( Y \) of the local TE can be either positive or negative (with the source \( y(t) \) being either informative or misinformative respectively) for a specific event set \( \{ x(t), x(t+1), \ldots, x(t+k) \} \).

Variable Selection

We apply the local TE from each source to bitcoin using the nearest-neighbor estimation [4]. Here, the Markovian order \( k, l \) and neighbor parameter \( K \) are varying from 1 to 10, for a total of 1000 different estimations for each driver. The permutation testing is used to measure the statistical significative flow of information. Then, the highest TE on the tuple \( \{ k, l, K \} \) of each significative driver is considered further as feature of a deep learning model.

Deep learning classification models

We can think of artificial neural networks (ANNs) as a mathematical model whose operation is inspired by the activity and interactions between neuronal cells due to their electrochemical signals. The main advantages of ANNs are their non-parametric and nonlinear characteristics. The essential ingredients of an ANN are the neurons that receive an input vector \( x \), and through the point product with a vector of weights \( w \), generate an output via the activation function \( g(\cdot) \):

\[
f(x) = g(x \cdot w + b),
\]

where \( \beta \) is a trend to be estimated during the training process. The basic procedure is the following. The first layer of neurons or input layer receives each of the elements of the input vector \( x \) and transmits them to the second (hidden) layer. The next hidden layers calculate their output values or signals and transmit them as an input vector to the next layer until reaching the last layer or output layer, which generates an estimation for an output vector. Particularly, we considered models with several layers also known as deep learning models [5].

Specifications

- Training set: 75% (before pandemic)
- Validation set: 13% (pandemic)
- Test set: 12% (post-pandemic)
- Scenarios:
  - S1: univariate
  - S2: all features
  - S3: significant features
  - S4: local TE
  - S5: significant features + local TE
- Architectures:
  - D1: Deep LSTM
  - D2: Wide LSTM
  - D3: Deep Bidirectional LSTM
  - D4: Wide Bidirectional LSTM
  - D5: CNN

Conclusions

- An attention must be paid to the evidence about theorder \( k = l = 1 \) throws values near zero. Practitioners usually assume this scenario under Gaussian estimations. Then, a precaution must be put to the memory parameters of Markov at least when working with the KSG estimation.
- On the other hand, the forecasting of Bitcoin’s price direction improves in the validation set, but not for all metrics in the test dataset when including significant drivers or local TE as a feature.
- Two methodological contributions to highlight are the use of nontraditional indicators such as market sentiment, as well as a continuous estimation of the local TE as a tool to determine additional drivers in the classification model.
- Finally, the models presented here are easily adaptable to high-frequency data because they are non-parametric and nonlinear in nature.

Acknowledgements

This research was funded Consejo Nacional de Ciencia y Tecnología (CONACYT, Mexico) through fund FOSEC SEP-INVESTIGACION BASICA (Grant No. A1-A-34514).

References