A Time Series Approach to Explainability for Neural Nets with Applications to Risk-Management and Fraud Detection

Marc Wildi[3] and Branka Hadji Misheva[4]

September 13, 2022

[3] ZHAW Zurich University of Applied Sciences, IDP; e-mail: wlmr@zhaw.ch

[4] BFH Bern University of Applied Science; Institute for Applied Data Science and Finance; email: heb1@bfh.ch

Table of contents

1. Forecasting with NN
2. Why do we Need Explainability?
3. Classic XAI Approaches
4. Regression and Explainability
5. NN and Explainability: LPD
6. LPD Applied to Bitcoin
7. Risk Management
8. Summary

Neural Nets: Forecasting

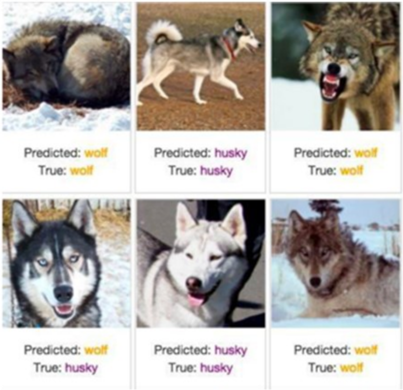
1. Forecast performances: review of **international forecast competitions**
   - M1 (1982), M2 (1993), M3 (2000), NN3 (2007) and NN5 (2009) competitions: **classic linear** approaches **outperform** computationally intensive approaches (including neural nets)
   - M4 (2020) and M5 (2021): **hybrid approaches** based on a mix of ARIMA and neural nets **outperform** (hybrid approach won M4 competition)
2. Accruing interest in **neural nets for forecasting** (in particular economic time series)

Neural Nets: Main Problems/Issues

1. Problem 1: **random nets**, see paper (numerical computations, learning/estimation)
2. Problem 2: **Black-box** or how to relate 'output' to 'input'? **Trust**?
3. Problem 3: **Overfitting** of richly parameterized nets (will be adressed in follow-up paper)

Why Do We Need Explainability?

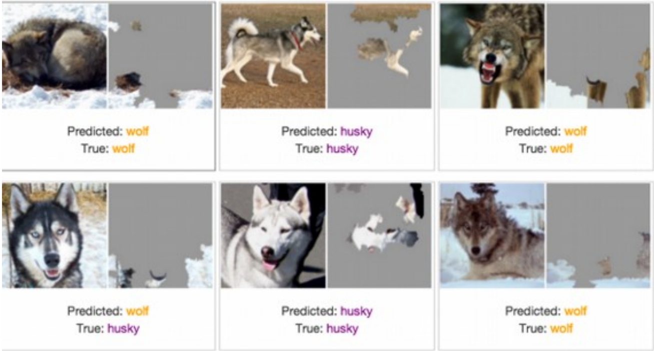
Let's consider the 'Husky vs Wolf' experiment results.



- The classifier makes one mistake!

Why Do We Need Explainability?

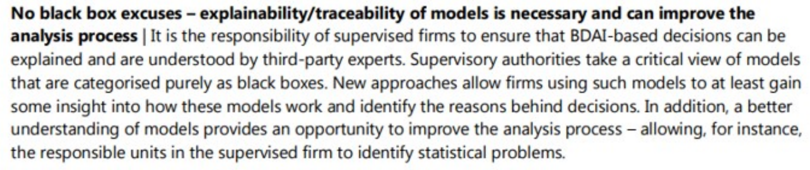
Next, we investigate which features drive the classification.



- The decision is based on whether there are white patches on the image!

Why Do We Need Explainability?

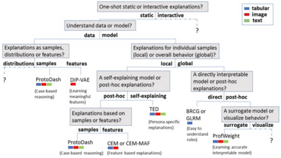
- Verify that **accuracy** is the result of **proper problem representation**
  - The model is capturing **relevant dependencies** between features.
  - This ensures **trust** in the system.
- **Communication**: convince layperson
- **Regulation** demands it.



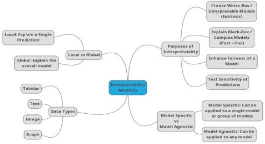
No black box excuses – explainability/traceability of models is necessary and can improve the analysis process | It is the responsibility of supervised firms to ensure that BDAI-based decisions can be explained and are understood by third-party experts. Supervisory authorities take a critical view of models that are categorised purely as black boxes. New approaches allow firms using such models to at least gain some insight into how these models work and identify the reasons behind decisions. In addition, a better understanding of models provides an opportunity to improve the analysis process – allowing, for instance, the responsible units in the supervised firm to identify statistical problems.

Figure: **Extract: Bafin AI and Big Data Report 2020**

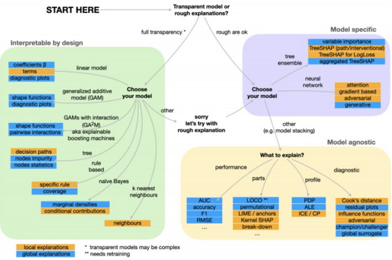
Deploying Explainability: Zoo of XAI models


One-shot static or interactive explanations?
static
interactive
?
tabular
image
text
Understand data or model?
data
model
Explanations as samples, distributions or features?
distributions
samples
features
?
ProtoDash
(Case-based reasoning)
DIP-VAE
(Learning meaningful features)
Explanations for individual samples (local) or overall behavior (global)?
local
global
A self-explaining model or post-hoc explanations?
post-hoc
self-explaining
Explanations based on samples or features?
samples
features
ProtoDash
(Case-based reasoning)
CEM or CEM-MAF
(Feature based explanations)
TED
(Persona-specific explanations)
A directly interpretable model or post-hoc explanations?
direct
post-hoc
BRCG or GLRM
(Easy to understand rules)
A surrogate model or visualize behavior?
surrogate
visualize
ProfWeight
(Learning accurate interpretable model)
?

Arya et al. (2019) proposed taxonomy based on questions about what is explained, how it is explained and at what level


Interpretability Methods
Local vs Global
Local: Explain a Single Prediction
Global: Explain the overall model
Data Types
Tabular
Text
Image
Graph
Purposes of Interpretability
Create White-Box / Interpretable Models (Intrinsic)
Explain Black-Box / Complex Models (Post - Hoc)
Enhance Fairness of a Model
Test Sensitivity of Predictions
Model Specific vs Model Agnostic
Model Specific: Can be applied to a single model or group of models
Model Agnostic: Can be applied to any model

Linardatos et al. (2021) taxonomy mind-map of Machine Learning Interpretability Techniques.



Maksymiuk et al. (2021) model-oriented taxonomy for XAI method

Figure: **Machine Learning Interpretability Techniques**

Deploying Explainability: Utility of Classical XAI Methods for Finance

- Classic approaches are **data-intensive** (sometimes odd/difficult to explain...)
- Solution:
  - Create '**new**' data (simulation) or
  - **Reshuffle** available data
- Creating new data: **contradiction** (don't know 'true' model)
- Re-shuffling: kills **dependencies** (trends, vola-cluster, draw-downs, extreme events, ...)
- **Key limitation**: many classical methods **ignore feature dependence**

Example: BTC Time Series (Time-Orderd Data)

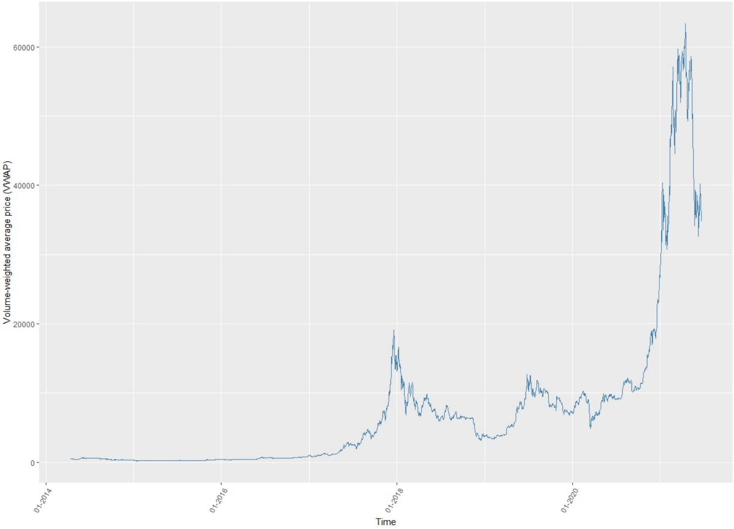


Figure: **BTC Prices (correct ordering)**

Example: Re-Shuffling BTC

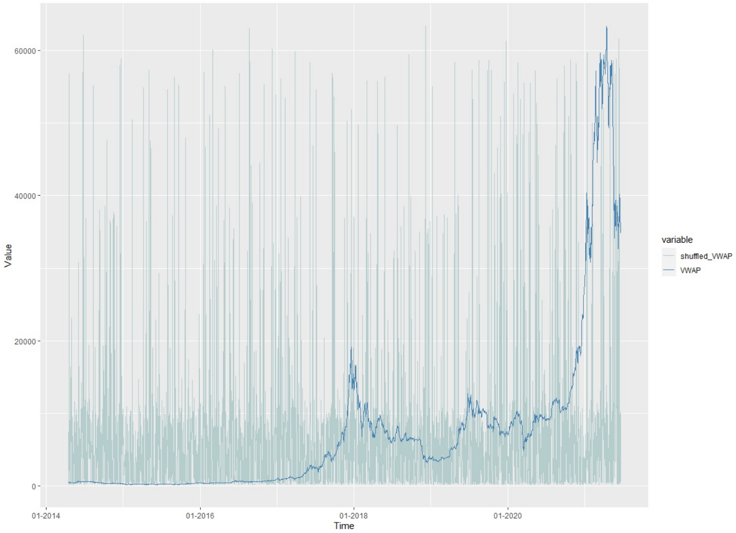


Figure: **BTC Prices (correct ordering vs shuffled values)**

Explainability Example 1: Classic Regression

- Let

$$y_t = 1 + 0.5x_{1,t} + 1.4x_{2,t} + \epsilon_t$$

- **Interpretation**: what is the meaning of the parameters?
- **Communication**: regression is 'well-known'
- **Utility**: change $x_{1,t}, x_{2,t}$ in view of achieving a certain target y_t (macro-economic model); **policy**
- **Validation**: confrontation with common sense/expert knowledge/**experience**

Explainability Example 2: Time Series

- Consider two simple **forecast** rules

$$
\begin{aligned}
\hat{x}_{t+1} &= 0.2x_t + 0.2x_{t-1} + 0.2x_{t-2} + 0.2x_{t-3} + 0.2x_{t-4} \\
\hat{x}_{t+1} &= 0.5x_t + 0.25x_{t-1} + 0.13x_{t-2} + 0.06x_{t-3} + 0.03x_{t-4}
\end{aligned}
$$

- **Interpretation**
  - Remote past is as important as present time
  - Remote past is less relevant than recent data
- **Learn** from model: dynamics of time series
- **Story-telling**

New XAI-Tool

- **Preserve dependence** (no re-shuffling)
- Avoid **inventing** (no simulation)
- Link to **regression** (no exoticism)

X-Function - I

- We propose a family of **Explainability (X-)functions xf(.)** for assigning meaning to the nets response or output over time t = 1, ..., T, where dimensional vector of possibly multiple output neurons.
- By selecting the identity $xf(\mathbf{o_t}) = \mathbf{o_t}$, we can mark preference for the sensitivities or partial derivatives $w_{ijt} := \partial o_{jt}/\partial x_{it}$, $i = 1, ..., n$, $j = 1, ..., n_p$, for each explanatory variable x_{it} of the net.
- In order to complete the 'explanation' derived from the identity one can add a synthetic intercept to each output neuron o_{jt} defined according to:

$$b_{jt} := o_{jt} - \sum_{i=1}^{n} w_{ijt} x_{it} \qquad (1)$$

X-Function - II

- For each output neuron o_{jt}, the resulting derivatives or 'explanations' $b_{jt}, w_{1jt}, ..., w_{njt}$ generate a new data-flow which is referred to as **Linear Parameter Data (LPD)**
- the LPD is a matrix of dimension $T * (n+1)$, irrespective of the complexity of the neural net, with $t-$th row denoted by $\mathbf{LPD}_{jt} := (b_{jt}, w_{1jt}, ..., w_{njt})$.
- The LPD can be interpreted in terms of exact replication of the net by a linear model at each time point t and the *natural* time-ordering of $\mathbf{LPD}_{jt}$ subsequently allows to examine changes of the linear replication as a function of time.
- We are then in a position to assign a meaning to the neural net, at each time point t = 1, ..., T, and to monitor non-linearities of the net or, by extension, possible non-stationarities of the data.
- see paper

X-function - III

- In order to give an intuition as to the sensitivities/explanations we want to obtain let's imagine the following brute approach:
  - We start by training a neural network (NN) on the specified inputs and response and store the results
  - Next, we perturb a selected input slightly
  - We use the trained NN and make the predictions for the changed inputs
  - For each changed variable, we collect the perturbed data and the corresponding NN-output
  - We fit a linear model and obtain the weights
  - We train the net for 100 different random initialization and **we observe the dependency of the LPDs across the different random nets.**

Context

- Derivation of LPD: **formula**, see paper
  - web-address: **https://www.explainableaiforfinance.com/repository-of-papers**
- Generic concept: **X-function**, see paper
  - **QPD** see paper (departures from linearity)

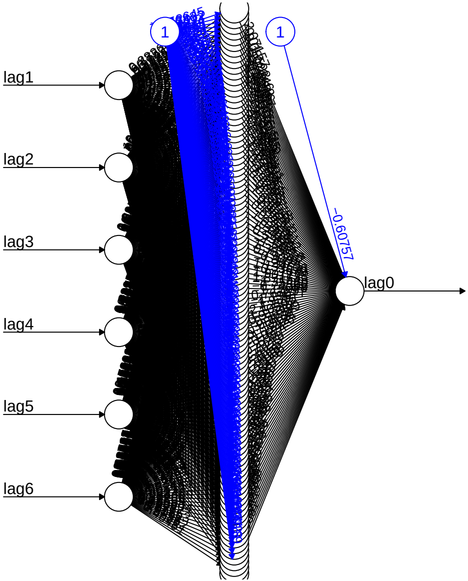
Application of LPD to BTC Cryptocurrency

- **Data**: Bitcoin returns 15-04-2014 to 30-06-2021
- **Model**: simple feedforward net with a single hidden layer and an input layer collecting the last six lagged (daily) returns: the net is then trained to predict next dayâs return based on the MSE-criterion. The number of estimated parameters then amounts to a total of 6 â 100 + 100 = 700 weights and 100 + 1 = 101 biases

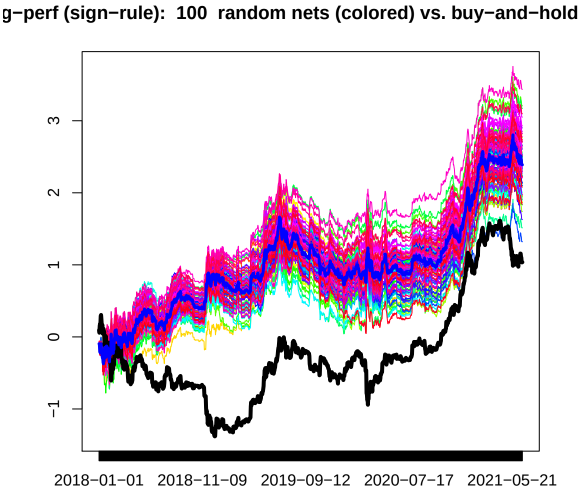
$$B\hat{T}C_{T+1} = NN(BTC_T, BTC_{T-1}, ..., BTC_{T-5})$$

- We optimize the net 100-times, based on different random initializations of its parameters, and we compute trading performances of each random-net based on the simple sign-rule

Net-Architecture



Performance



LPDs and Risk Management

- **Remember**: We observe the dependency of the LPDs across the different random nets.
  - Our initial insights suggest that the time-varying dependency of the data measured by the LPDs is indicative of different states of the market.
  - In particular weak dependency (small absolute LPD) is an indicator of randomness or chaos.
  - We therefore propose a simple rule for managing risks: exit markets at times tagged as chaotic by the LPD.

Market-Exit when |LPD| Weak (Critical Time)

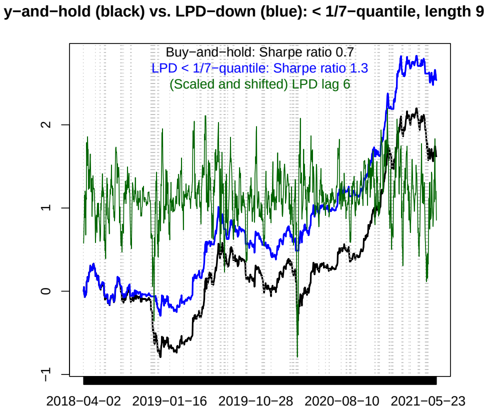


Figure: Buy-and-hold (black) vs. Out-of-sample (mean-) LPD market-exit strategy (blue): exits (shaded in grey) occur if today's out-of-sample mean-LPD (green) drops below the 1/7-quantile.

LPDs and Explainability

- Since the LPD corresponds to the parameters of a (time-dependent) linear replication of the net, synthetic t-statistics could be computed for inferring the relevance of the explanatory variables by computing the ratio of mean-LPD and standard-deviation

$$\mathbf{t}_t := \frac{\overline{\mathbf{LPD}}_t}{\boldsymbol{\sigma}_t}$$

at each time point t, corresponding to (a vector of) synthetic t-statistics, one for each input variable

Summary

- Classic XAI methods are not suited for financial time series data
- **X-functions and tool developed**: exact, fast, intuitive and preserves the natural time ordering of data
- Can be used to explain variable importance though proper statistical testing and can be used for risk management
- Check out our website, paper and code:
  - Website
  - Paper
  - Code