Shapley Value Based Variable Interaction Networks for Data Stream Analysis

Jan Zenisek[1,2(✉)], Sebastian Dorl[1], Dominik Falkner[3], Lukas Gaisberger[1], Stephan Winkler[1,2], and Michael Affenzeller[1,2]

[1] University of Applied Sciences Upper Austria, Softwarepark 11, 4232 Hagenberg, Austria
[2] Institute for Symbolic Artificial Intelligence Johannes Kepler University Linz, Altenberger Straße 69, 4040 Linz, Austria
[3] RISC Software GmbH, Softwarepark 32a, 4232 Hagenberg, Austria
jan.zenisek@fh-hagenberg.at

Abstract. Due to the growing use of machine learning models in many critical domains, ambitions to make the models and their predictions explainable have increased recently significantly as new research interest. In this paper, we present an extension to the machine learning based data mining technique of *variable interaction networks*, to improve their structural stability, which enables more meaningful analysis. To verify the feasibility of our approach and it's capability to provide human-interpretable insights, we discuss the results of experiments with a set of challenging benchmark instances, as well as with real-world data from energy network monitoring.

Keywords: Interpretable machine learning · Shapley value · Data stream analysis · Energy network resilience · Photovoltaic systems

1 Background and Motivation

With the progressing digital transformation of all areas of life, more and more continuous data (i.e. data streams) is being recorded and subsequently evaluated with machine learned models in real-time. Prominent examples for this trend can be found in today's fast changing production industry (e.g. predictive maintenance), social media (e.g. opinion mining), the financial sector (e.g. real-time stock trading), or the energy sector (e.g. blackout prediction), to name just a few. In the research field of machine learning, speeding up training algorithms and improving the accuracy of resulting models represent ongoing and presumably infinite endeavors. However, especially when employed in critical domains, not just accurate, but interpretable models are necessary to enable trustworthy predictions. Making the models themselves, as well as their predictions explainable has been increasingly studied in the past few years [2]. However, recent efforts in producing interpretable machine learning models mostly consider *batch processed data*, whereas analyzing *real-time data streams* explicitly, has not gained

R. Moreno-Díaz et al. (Eds.): EUROCAST 2022, LNCS 13789, pp. 130–138, 2022.
https://doi.org/10.1007/978-3-031-25312-6_15

the same attention yet. Moreover, interpretable machine learning is mostly concerned with a set of dedicated input variables and one prediction target, which does not provide a comprehensive system insight.

In [6] these issues are addressed by using Variable Interaction Networks (VIN) [4] in order to analyze streaming data holistically and improve the understanding of system dynamics (e.g. potential concept drifts). While the results of this work show the applicability of VINs to detect changing system behavior quite accurately, they also depict structural instability of the continuously re-created networks, while analyzing the streaming data. Although this does not hamper the accuracy of change detection too much, as most network alterations are small, each alteration certainly impairs the networks' functionality for system interpretation by domain experts.

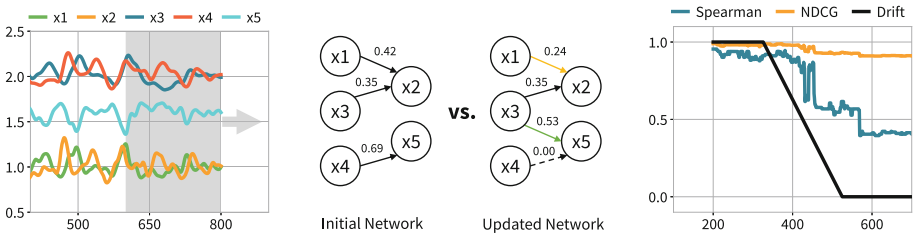
In the following Sect. 2, we describe the conventional approach to model and evaluate variable interaction networks on streaming data. After that, we present an extension to this, with the aim to decrease structural instability to support better interpretability. In Sect. 3 we show the feasibility of our extended approach by testing it on two data sets and we conclude briefly in Sect. 4.

2 Variable Interaction Networks

2.1 Modeling and Evaluation

Variable Interaction Networks (VIN) [4] are directed graphs, in which system variables are represented as nodes and their impact on each other as directed, weighted edges. The algorithm to create such models is as follows: For each independent system variable a model is trained, using the variable as target and all others as input. For this purpose, arbitrary machine learning methods may be employed. In a second step, for each model, the impact of each input variable for the respective target variable is calculated. This calculation is based on the *permutation feature importance* (PFI) [1], for example – the model error increase, which results from removing the information of a certain variable from the data set by shuffling its values. In a final step, the graph is constructed by adding a node for each variable and adding weighted, directed edges based on the calculated impacts. The resulting model structure provides a holistic system depiction as a *clear-box* since it is human-readable. It has proven to be successful, not only to model stable system states, but also to analyze system dynamics when evaluating data streams in a sliding window fashion (see [6] and Fig. 1). To this end, raw data is processed within a sliding window (Fig. 1a) by re-computing the VIN and comparing it to an initial version (Fig. 1b) using the *Normalized Discounted Cumulative Gain (NDCG)* or the *Spearman's rank correlation*, as proposed in [6]. This results in network similarity trend lines (Fig. 1c), which can be compared to the real system drift by using *Pearson's R* correlation coefficient, in case it is known. We define drift limits as $noDrift = 1$ and $fullDrift = 0$, to get a positive scale for correlation scores. Benchmark tests [6] show the effectiveness of VIN based drift detection, however, also report that these networks currently lack of structural stability. Reasons for this are that

feature impact calculation using PFI is non-deterministic and heavily depends on the underlying models' estimation error. This instability compromises the interpretability of VINs and thus, motivated us to look for improvements.



(a) VIN Evaluation **(b)** Network Comparison **(c)** Network Similarity

Fig. 1. Symbolic illustration of the VIN based data stream analysis approach on a set of time series data with manually introduced (i.e. known) drift. The edge design in 1b encodes the following states: black=unchanged, yellow=changed, green=new, dashed=vanished variable impact. (Color figure online)

2.2 Shapley Value Based Networks

To mitigate the issue of network instability, we extend the work in [6] and introduce a new variable impact calculation routine using the *Shapley Value* method [5], to replace the former Permutation Feature Importance (PFI) based one. The Shapley Value (SV) of a variable is the average gain to the mean model prediction, resulting from adding the variable, to all possible coalitions of the remaining variables. It is a solid mathematical concept from coalition game theory and enables local, i.e. observation-wise model interpretation: Variable impacts are evaluated for each data point individually, which has the potential to show effects of changing system behavior instantaneously. For comparability we adapted the calculation as follows: We scale the resulting absolute numbers to unit length within the interval $[0, 1]$, which was also performed for the PFI outcome. Measuring the effect of adding a variable was done in a reversed fashion: calculating the current impact, then removing the variable information by picking a random value from the variable's recordings and finally, re-calculating the impact. To reduce variance, we repeat this process 10 times and average the outcome, as we did for the shuffling routine of PFI. Eventually, we collect and average the observation-wise calculations to get a global mean for each variable impact.

3 Experiments

In the scope of this work and the focus of this section, we tested the effectiveness of the proposed Shapley Value (SV) extension to the variable interaction network

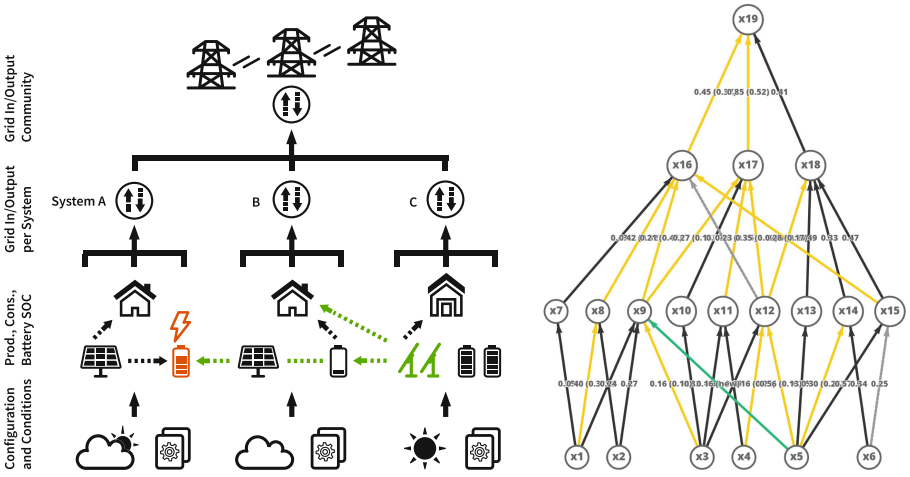
approach compared to Permutation Feature Importance (PFI). Therein, we use different underlying learning algorithms, a varying sliding window size and two problem instances: a synthetically constructed benchmark problem describing dynamically changing communicating vessels over time and a real-world problem from the field of photovoltaic energy production.

3.1 Problem Instances

Benchmark Problem "Communicating Vessels (ComVes)": For this problem we designed a differential equation system to simulate data streams, which drift over time, first introduced and detailed in [6]. The system consists of two vessels, each continuously filled by an inlet, drained by an outlet and connected by a communication path. The system is designed to maintain a stable state, however, by manipulating the equation for the flow rate of the vessel connection, a concept drift can be introduced: a gradually clogging communication path, e.g. representing a maintenance problem.

Real-World Problem "Resilient Energy Networks (ResiNet)": The ResiNet-project is concerned with analyzing energy networks with regard to their resilience. As part of this, we developed prediction models for power production and consumption based on data from ca. 200 households from the region of Upper Austria, all equipped with roof-top mounted photovoltaic modules and battery packs. The measurements include data from 2016–2019 and were further linked to several geographic information and weather data from the Austrian weather forecasting system INCA [3]. To investigate network resilience and to test our extended VIN approach, we designed following what-if scenario: *What if... a small community of 3 systems is sharing its batteries by charging them together for higher network-independence? Can we detect a failing battery pack in such a scenario with our approach?* (cf. illustrated in Fig. 2a). For this purpose, we used the measured real-world system data, but re-calculated battery states and grid input/output differences, to simulate that the systems are connected and sharing their surplus energy produced. For instance, if the consumption of a system can be covered with the current energy production and the system's battery is already filled, surplus energy is shared equally amongst the other community members and only after that, left over energy is passed to the public grid. This way, a small virtual energy community is simulated, which should provide more grid independence. In order to enable more reasonable analysis, we used our domain knowledge and pre-selected input features for the modeling process in the case of this problem instance, instead of alternating all available features (cf. Figure 2b). For each of the experiment runs, we introduced rapid degradation of

the charge/discharge rate of a random battery at a random point of time, using the data stream generation tool from [7]. By this means, we aim to simulate a probable, but not directly observable maintenance problem, which could impede the gained network independence, but potentially remains undetected due to the compensatory behavior of the interconnected energy community.

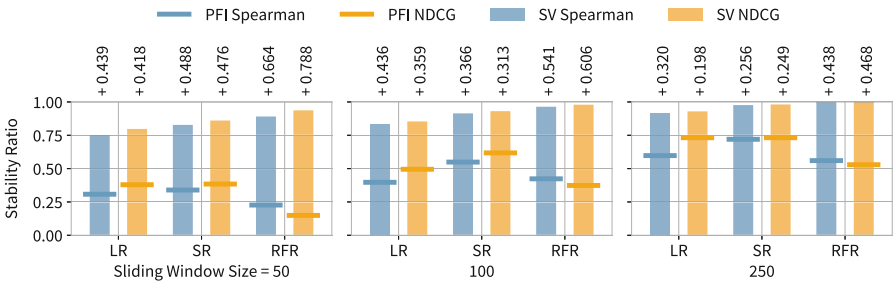


(a) ResiNet schematic **(b)** Knowledge structured VIN

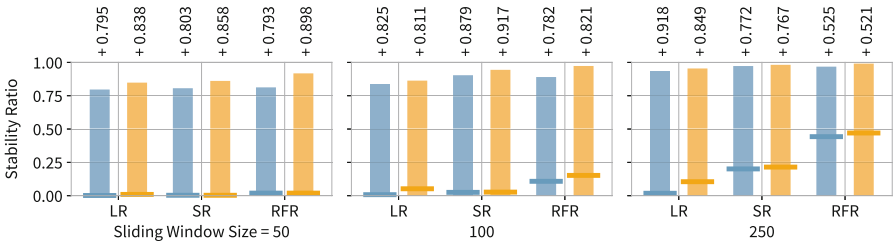
Fig. 2. Depiction of the ResiNet problem instance: In 2b the herein described *what-if*-scenario concerning a simulated, virtual energy community under changing conditions, is illustrated (cf. energy sharing in green, battery fault in red). In 2b the respectively modeled and subsequently evaluated VIN is displayed.

3.2 Results

To compile the foundation of the variable interaction networks (VIN), we trained regression models using multiple linear regression (LR), symbolic regression (SR) and random forest regression (RFR) with the configuration as in [6]. We defined a maximum normalized mean squared error (NMSE) of 0.5 for each model and a minimum variable impact of 0.1 as thresholds to take part within the network creation routine. Further on, we compare different sliding window sizes and both impact calculation methods – Permutation Feature Importance (PFI) and Shapley Values (SV) – for which we provide the calculated differences on each result plot. For the sake of brevity, we elaborate on results in the respective plot's caption and provide a brief discussion at the end of this subsection.



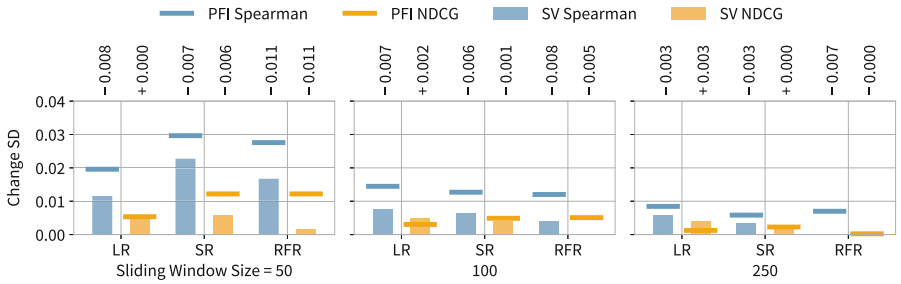
(a) ComVes: NDCG are slightly better than Spearman scores in most cases; best PFI scores with SR, worst with RFR; larger window sizes reveal more stability; SV scores are superior in all cases.



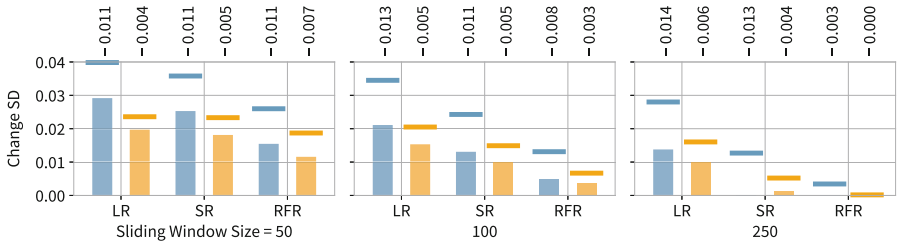
(b) ResiNet: NDCG are slightly better than the Spearman scores in most cases; RFR models work best, especially for PFI scores; SV are superior to PFI scores by far in any case.

Fig. 3. Standard Deviation (SD) of changes, representing the mean magnitude of network changes during sliding window evaluation. The lower the deviation, the better for model interpretability.

We set up two experiment types, each consisting of 10 runs with randomly sampled time series with a length of 1000 consecutive events originating from the described problem instances: one for testing network stability and one for drift detection. All models were trained on data partitions where systems were stable. To evaluate the stability of the created networks over time, we tested on time series data, which was again sampled from stable system states. In Fig. 4 the ratio of sliding window movements without resulting network change is illustrated. In Fig. 3 the magnitude of detected changes is given by reference to their standard deviation (SD) during the runs. To evaluate the drift detection capability of the approach, we tested on unseen data for which a concept drift has been introduced at a random point, after a fixed burn-in phase of 250 events – see details in Fig. 5.



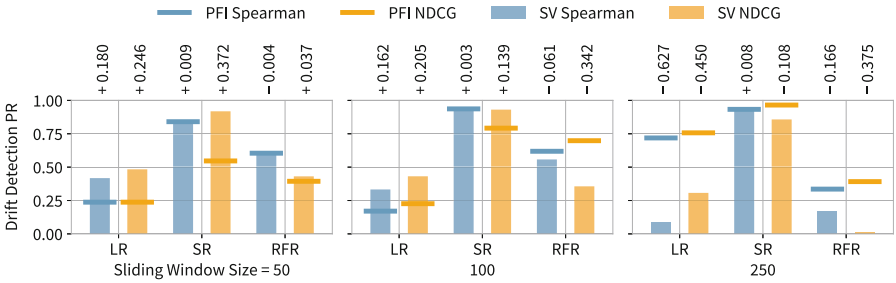
(a) ComVes: NDCG are better than the Spearman scores; larger window sizes reveal lower SDs; SV are better than PFI scores for most combinations of model and window size.



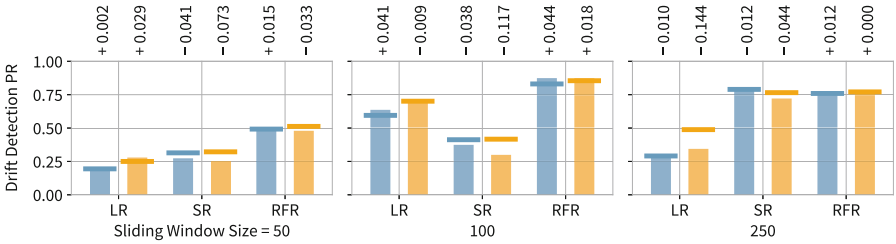
(b) ResiNet: RFR perform best, followed by SR and LR; NDCG are better than the Spearman scores; larger window sizes reveal lower SDs; SV scores are superior, no matter the model nor window size.

Fig. 4. Stability ratio test results, representing the ratio of sliding window movements without network change. Thus, a high ratio is desirable to increase model interpretability.

We want to highlight the superiority of the new SV over PFI based VINs in all test cases regarding network stability ratio when analyzing stable systems (Fig. 4). This also applies for the standard deviation of weight changes, as they are lower for SV based VINs in most cases (Fig. 3). Furthermore, in this analysis we see a pronounced improvement of using the NDCG over the Spearman scores. As shown in Fig. 5 both methods, SV and PFI, generate comparably good results in terms of drift detection performance. To this end, the SR based VINs for the benchmark data and the RFR based VINs for the real-world data perform best, as the high *Pearson R* correlation scores show. In summary, these results suggest that SV based VINs are superior to PFI based ones, since performance on stable systems is noticeably improved without losing the ability to detect system changes.



(a) ComVes: no clear winner between NDCG and Spearman scores; no clear winner between PFI and SV scores; SR models perform best by far at a high level with window sizes 100 and 250.



(b) ResiNet: NDCG/Spearman and PFI/SV with very similar scores; RFR performs best.

Fig. 5. Concept drift detection performance, represented by *Pearson's R* (PR) correlation coefficient of the network similarity and the known drift over time.

4 Conclusion and Outlook

With this work we presented an extension to the variable interaction network modeling and evaluation technique for data stream analysis, giving it more stability and thus, improving it's interpretation potential. Therefore, we propose a customized form of Shapley Values as alternative to the conventional permutation feature importance for computing network edge weights. The effectiveness of this extension has been shown for a benchmark and a real-world problem data set, both dealing with stable and changing system behavior.

Future work may consider other variable impact (i.e. feature importance) estimation measures to further improve the characteristics of variable interaction networks and broaden its application scope. Another promising lead is to investigate the potential of VINs for root-cause analysis, e.g. by analyzing those network paths with the highest change sum within a VIN, which is evaluated on data with concept drifts.

Acknowledgments. The work described in this paper was done within the projects "RESINET", funded by the European Fund for Regional Development (EFRE) and the country of Upper Austria as part of the program "Investing in Growth and Jobs 2014–2020" and "Secure Prescriptive Analytics", funded by the country of Upper of Austria as part of the program "#upperVISION2030".

References

1. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324
2. Carvalho, D.V., Pereira, E.M., Cardoso, J.S.: Machine learning interpretability: a survey on methods and metrics. Electronics **8**(8), 832 (2019). https://doi.org/10.3390/electronics8080832
3. Haiden, T., Kann, A., Pistotnik, G., Stadlbacher, K., Wittmann, C.: Integrated nowcasting through comprehensive analysis (INCA)—system description. ZAMG Rep **61**, 1–60 (2010). https://www.zamg.ac.at/fix/INCA_system.pdf
4. Hooker, G.: Discovering additive structure in black box functions. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge Discovery and Data Mining, pp. 575–580 (2004). https://doi.org/10.1145/1014052.1014122
5. Shapley, L.S.: A value for n-person games. In: Kuhn, H., Tucker, A., (eds.), Contributions to the Theory of Games II vol. 2, no. 28, pp. 307–317 (1953). https://doi.org/10.1515/9781400881970-018
6. Zenisek, J., Kronberger, G., Wolfartsberger, J., Wild, N., Affenzeller, M.: Concept drift detection with variable interaction networks. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds.) EUROCAST 2019. LNCS, vol. 12013, pp. 296–303. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-45093-9_36
7. Zenisek, J., Wolfartsberger, J., Sievi, C., Affenzeller, M.: Streaming synthetic time series for simulated condition monitoring. IFAC-PapersOnLine **51**(11), 643–648 (2018). https://doi.org/10.1016/j.ifacol.2018.08.391