Arrhythmia detection using spiking variable projection neural networks

Péter Kovács[1], Kaveh Samiee[2]

[1] Department of Numerical Analysis, Eötvös Loránd University, Budapest, Hungary
[2] GE Healthcare, Helsinki, Finland

Abstract

Edge-based Machine Learning (ML) has a pivotal role in revolutionizing smart healthcare by introducing a tangible improvement in the secure and discrete medical data analysis. This paper presents a novel neural network architecture by combining Variable Projections (VP) and Spiking Neural Networks (SNN). VPs are nonlinearly parameterized orthogonal projections whose weights have physical meaning, whereas SNNs are biologically plausible neural networks that operate on both spatial and temporal information. In the proposed hybrid topology, VP layer is coupled with spiking layers to encode input space into a compact and interpretable latent feature space. The latent space, encoded into spike trails, enables the subsequent SNNs to be trained with a low bandwidth. The effectiveness of the proposed VPSNN architecture is assessed in binary classification of normal and ventricular ectopic beats (VEBs) in ECG recordings of the PhysioNet MIT-BIH Arrhythmia Database. In our experiments, ECG records are apportioned into balanced training and test sets with approximately 60/40 ratio. Results show that a VPSNN variant detects VEBs with an overall classification accuracy of 97.16%*, with a highly shallow topology consisting of only 242 trainable parameters. The compact topology of VPSNN, makes it a suitable candidate for neuromorphic computing.*

1. Introduction

The large availability of biomedical data allows to develop reliable medical tools for detecting cardiovascular diseases using data-driven machine learning approaches. In fact, state-of-the-art deep learning (DL) methods provide promising results in the classification of various arrhythmia types. On the other hand, in clinical applications, improving classification accuracy alone is usually not enough, as physiological interpretation of the results are also important. Besides the lack of explainability, high computational complexity is another drawback of the top performing DL models.

In this paper, we alleviate the previously mentioned problems by combining variable projections (VP) with spiking neural networks (SNN). VPs are nonlinearly parameterized orthogonal projections whose weights have physical meaning, whereas SNNs are brain-inspired network topologies, which can be implemented with ultra-high speed and ultra-low energy consumption on neuromorphic devices. In order to combine the advantages of VPs and SNNs, we design a hybrid neural network model called VPSNN. This architecture is adapted specifically to ECG data analysis such that the first VP layer servers as an automatic feature extractor and spike encoder whose weights represent the positions and the widths of the clinically relevant ECG waveforms (P-QRS-T). The VP layer constructs a clinically interpretable latent feature space and is trained together with the subsequent SNN layers, which are responsible for the feature analysis and the actual classification of the data.

As a case study, we consider the classification of normal and ventricular ectopic beats (VEBs) in real ECG recordings of the PhysioNet MIT-BIH Arrhythmia Database. Our experiments show that the proposed VPSNN architecture can be effectively used for detecting VEBs with an overall classification accuracy of 97.16%, comparable to the state-of-the-art. Moreover, due to highly compact topology of VPSNN, it offers a low computational cost inference ability suitable for edge computing in clinical applications.

2. Methodology

2.1. Variable Projection Neural Networks (VPNNs)

VPs are parameterized orthogonal transformations introduced by Golub and Pereyra [1] to solve separable nonlinear least squares problems:

$$\min_{\theta} r_2(\theta) := \min_{\theta} \left\| x - \Phi(\theta)\Phi^{+}(\theta)x \right\|_2^2, \qquad (1)$$

where $x \in \mathbb{R}^N$ is the data to be fitted, and $\Phi^{+}(\theta)$ stands for the Moore–Penrose pseudoinverse of the matrix $\Phi(\theta) \in \mathbb{R}^{N\times n}$. For a given parameter $\theta \in \mathbb{R}^m$, the matrix product $\Phi(\theta)\Phi^{+}(\theta)$ represents the orthogonal projector on the

linear space spanned by the columns of $\Phi(\theta)$. The entries of the matrix $\{\Phi(\theta)\}_{i,j} = \varphi_j(t_i;\theta)$ can be defined by a set of functions $\{\varphi_j(t;\theta) \,|\, j = 1,\ldots,n\}$ sampled at a sequence of points $t_1, t_2, \ldots, t_N$. The functions $\varphi_j(\cdot;\theta)$'s are chosen a priori to the problem in question in such a way that the parameter θ usually represents physical quantities, e.g., attenuation coefficients, calibration parameters, dominant frequencies, etc. Therefore, VPs provide an interpretable representation of the input x, which motivated the construction of VPNNs [2].

VPNN is a type of model-driven neural network, where the design of the NN architecture resembles the solution to well understood mathematical problems. In fact, the VPNN integrates a separable non-linear least squares problem into its first layer that performs a VP operator on the input data. The VP layer has two modes of operations, its output could be either the projected signal $\hat{x} = \Phi(\theta)c$ or the coefficients of the projection $c = \Phi^{+}(\theta)x$. Note that both cases are differentiable [1], thus backpropagation can be used for training [2]. In this work, all the VP layers forward the coefficient vector c of the orthogonal projection of the input x with respect to a set of parameterized Hermite functions $\varphi(\cdot;\theta)$ where $\theta = (\tau, \lambda)$. In this case, the trainable weights of the VP layers are the translation τ and the dilation λ of the basis functions [2]. Hermite functions are proved to be well suited to model spike like waveforms, such as the QRS complexes [3]. Consequently, the first VP layer in the proposed network architectures extracts features automatically with interpretable layer parameters τ and λ, which are proportional to the location and the width of the fitted QRS complexes, respectively.

2.2. Spiking Neural Networks (SNNs)

The family of SNNs is referred to as the third-generation networks [4], which recently received much attention from the AI community. The basic computational unit of these networks are the so-called spiking neurons, which turned to be a biologically more accurate neuron model compared to their artificial counterparts. In fact, the activation functions of the SNNs are extended by a temporal dimension to encode the timing of the propagated spikes.

There are several ways to describe neuronal dynamics, one of the most commonly used is the so-called leaky integrate-and-fire (LIF) model. This involves a linear differential equation, whose iterative solution can be utilized to define the output of the LIF neuron [5]. Namely, the membrane potential $u_{t,n}$ of the nth layer evolves according to:

$$u_{t,n} = \tau u_{t-1,n} \cdot (1 - o_{t-1,n}) + \sum_j w^j o^j_{t,n-1}, \qquad (2)$$

$$o_{t,n} = H(u_{t,n} - V_{th}), \qquad (3)$$

where τ is a fixed decay factor, V_{th} is a predefined threshold, $o_{t,n}$ stands for the binary output, H denotes the Heaviside step functions, and the sum in Eq. (2) aggregates the spike outputs received from the LIF neurons of the previous layer. When the membrane potential exceeds the threshold V_{th}, a binary spike is generated, which is followed by a soft reset of the membrane potential, i.e. $u_{t,n}$ is either reduced by τ or reset to zero. Note that the output of the LIF neuron is fed back to its input, thus it can be interpreted as a recurrent neural network (RNN) whose hidden state is the membrane potential. Although, the spiking activity is non-differentiable due to the Heaviside step function in Eq. (3), the gradient can be approximated as follows:

$$\frac{\partial o_{t,n}}{\partial u_{t,n}} = \frac{1}{a} \operatorname{sign}\left(|u_{t,n} - V_{th}| < \frac{a}{2}\right), \qquad (4)$$

where a is a user-defined constant that controls the peak width and ensures that the integral of the gradient function is equal to one. In [5], it was shown that Eq. (4) is a proper approximation to the true gradient which enables learning with backpropagation.

2.3. VPSNN variations

In contrast to Artificial Neural Networks (ANNs), spiking neurons are communicating with binary spike trains that supports efficient implementations on target hardware. In fact, SNN can run faster on neuromorphic devices [6], since the inference require fewer floating point computations. Power efficiency is another advantage of SNNs that originates from the sparse nature of the transmitted binary spike trains [7]. This motivated us to replace the fully connected ANN block in our previous work [2], and combine SNNs with VP operators. The latter also serves as a trainable feature extractor that reduces the dimension of the input data.

The SNNs process data in an event-based manner, where the information is represented via occurrences of spikes. However, real-world measurements are usually not encoded and stored by spike trains, thus a conversion is needed. There are various ways to convert input data into spikes [7] which we choose the simplest one that is direct input encoding. In this case, the static data are interpreted as constant currents, i.e., the same features are passed to the SNN at every time step. In our framework, this can be implemented in two different ways. If we consider n_{steps} number of time steps and repeat the input accordingly, we can apply either a single or multiple VP operators to the repeated input. The corresponding latent feature spaces are essentially different, which is demonstrated by Fig. 1 a)-b). Besides the structure of the latent feature space, we experimented with various other configurations of the VP and the fully connected spiking layers:

• *Basic VPSNN*: a single VP layer whose output is replicated n_{steps} number of times, and then forwarded to subsequent fully connected SNN layers with LIF activations, and with a softmax activation at the end. The basic architecture is depicted by Fig. 1 a), where the optinal scaling and convolutional layers are turned off.
• *Multiple (M)VPSNN*: multiple VP layers are connected in parallel, whose outputs are forwarded to the subsequent layers. Fig. 1 b) displays this architecture with inactive scaling and convolutional layers.
• *Convolutional VP(C)SNN*: same as the basic VPSNN, but the convolutional layer prior to the SNNs is turned on.
• *Temporal VP(T)SNN*: same as the basic VPSNN, but the temporal scaling layer after the VP layer(s) is turned on.

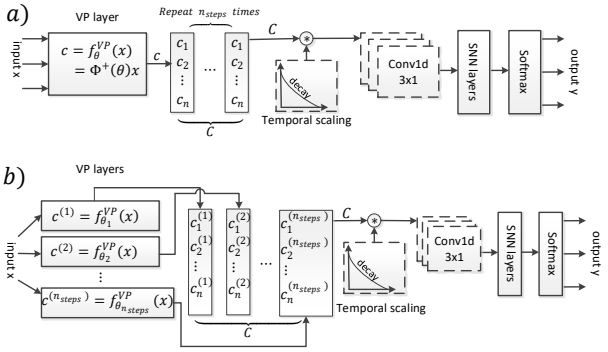

Figure 1. VPSNN variations: a) single VP layer with replicated output channels, b) multiple VP layers with concatenated output channels.

3. Experimental results

3.1. Dataset

PhysioNet MIT-BIH Arrhythmia Database [8] consists of 48 two-channel ambulatory ECG recordings. Each record is with around 30 minutes length and was digitized with sampling frequency of 360 Hz per channel. Normal heartbeats are heavily overrepresented in the database which poses a challenge for training classifiers. To mitigate this problem, we selected a balanced subset of the original dataset. More precisely, after preprocessing and heartbeat segmentation [9], we split the ECG beats into training (DS1) and test (DS2) sets according to [10]. Heartbeat signals of these sets come from different patients, thus there is no data leakage between the training and test phases. Then, we selected all the VEBs from DS1 and DS2 along with an equal number of normal heartbeat signals as well. This resulted in 4260 plus 4260 heartbeat signals for training, and 3220 plus 3220 signals for testing. Only the QRS complexes from lead I, i.e., 100 samples in the vicinity of the R peaks were inputted to the NNs. All neural networks were trained and tested on the QRS complexes of DS1 and DS2, respectively. Therefore, we expected to have a fair comparison of all the tested NN architectures.

3.2. Classification performances

A comparative study is performed in order to quantify classification performances of four VPSNN variants in contrast to five competitor classification models. These classifiers are namely, a conventional fully connected ANN, a fully connected SNN, a conventional Convolutional Neural Network (CNN), a Spiking CNN (SCNN) and a non-spiking VP neural network (VPNN). All the classifier models are trained and tested using same sets used for VPSNN variants. Additionally, topology of each competitor model, is individually tuned in order to achieve its highest performance. Table 1 summarizes the overall classification performance obtained by each model on test set. As it can be seen, VPTSNN outperforms other classification models with an overall accuracy of 97.16% and an overall specificity rate of 99.60%. In terms of sensitivity rate, SNN model surpasses other classifiers with an overall sensitivity rate of 97.45%. VPTSNN obtains sensitivity rate of 94.72% which indicates that more VEB beats are wrongly detected as normal by VPTSNN in comparison with SNN. On the other hand, higher specificity rate of VPTSNN indicates that fewer normal beats might be detected as VEB compared to other classifiers.

Table 1. Overall Classification performances of different NN architectures.

Architecture	$Acc\%$	$Spec\%$	$Sens\%$
ANN	94.32%	95.53%	93.11%
CNN	95.92%	96.09%	95.75%
SNN	95.59%	93.73%	**97.45**$\%$
SCNN	95.42%	95.31%	95.53%
VPNN	96.65%	96.83%	96.61%
VPSNN	96.61%	99.10%	94.13%
VPTSNN	**97.16%**	**99.60%**	94.72%
MVPSNN	94.55%	96.52%	92.58%
VPCSNN	95.61%	98.07%	93.14%

Table 2 lists the total number of parameters of each classifier model. Moreover, inference cost of each model is defined in terms of total memory size of that is required to be allocated in a target device for deployment of the model.

As it can be seen in Table 2, the memory allocation sizes required by VPNN and VPSNN/VPTSNN for inference are 0.7 kilobytes (KB) and 5.3 KB, respectively. Moreover, VPNN ans VPCSNN can achieve fast forward propagation pass due to few number of mathematical op-

Table 2. Total number of parameters and inference memory size of different NN architectures.

Architecture	# of parameters	memory size
ANN	59010	234 KB
CNN	212610	967 KB
SNN	58880	231 KB
SCNN	376704	1.5 MB
VPNN	39	**0.7 KB**
VPSNN	242	5.3 KB
VPTSNN	242	5.3 KB
MVPSNN	72	2.2 KB
VPCSNN	**26**	1.1 KB

erators in their architecture. In our experiments, VPCSNN is constructed by a sole convolutional layer and a single SNN layer resulting in total 26 trainable parameters. However, other competitor models require wider and deeper topology in order to reach their highest performance which results in significantly larger memory allocation size. Thanks to the highly compact discriminative latent feature space obtained by the VP layer, stacking of NN and SNN layers to VP layer results in a shallower and thinner topology. Thus, there is a tangible reduction in total number of parameters in VP-based architectures compared to other conventional deep and/or wide models. For instance, both VPSNN and VPTSNN have only 242 parameters which 240 of them are trainable. The shallow and thin network topology obtained by VP-based models, can minimize the inference latency and computational cost. Therefore, they are easier to be implemented on neuromorphic device and can be a suitable candidate for edge-computing where low power consumption and low bandwidth are the main constraints.

4. Future work

In this paper we presented a novel spiking neural network variant with a thin and shallow architecture. Namely, VPSNN which is constructed by stacking SNN layers after VP layer. Experimental results, obtained on binary classification task of VEB beats, demonstrated the effectiveness of VPSNN as a lightweight solution for ML assisted edge computations. As a future work, we are planning for Field Programmable Gate Array (FPGA) implementation of VPSNN. Moreover, we are investigating the application of VPSNN in other domains such as tire sensor signal classification [11], and visually evoked potential classification [12].

Acknowledgments

This project was supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

References

[1] Golub GH, Pereyra V. The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. SIAM Journal on Numerical Analysis SINUM 1973;10:413–432.

[2] Kovács P, G. B, Huber C, Huemer M. VPNet: Variable Projection Networks. International Journal of Neural Systems IJNS 2021;32:2150054:1–19.

[3] Sörnmo L, Börjesson PL, Nygards ME, Pahlm O. A method for evaluation of QRS shape features using a mathematical model for the ECG. IEEE Transactions on Biomedical Engineering 1981;28:713–717.

[4] Wolfgang M. Networks of spiking neurons: The third generation of neural network models. Neural Networks 1997; 10(9):1659–1671.

[5] Wu Y, Deng L, Li G, Zhu J, Shi L. Spatio-temporal backpropagation for training high-performance spiking neural networks. Frontiers in Neuroscience 2018;12:331–342.

[6] Kamata H, Mukuta Y, Harada T. Fully spiking variational autoencoder. Proceedings of the 36th AAAI Conference on Artificial Intelligence 2022;1–15.

[7] Eshraghian JK, Ward M, Neftci E, Wang X, Lenz G, Dwivedi G, Bennamoun M, Seok Jeong D, Lu WD. Training spiking neural networks using lessons from deep learning. arXiv 2021;2109.12894:1–44.

[8] Goldberger AL, et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation 6 2000;101(23):e215–e220. DOI: 10.1161/01.CIR.101.23.e215.

[9] Bognár G, Fridli S. Heartbeat classification of ECG signals using rational function systems. In Moreno-Díaz et al. R (ed.), Computer Aided Systems Theory–EUROCAST 2017: Part II, volume 10672 of LNCS. Cham: Springer, 2018; 187–195. DOI: 10.1007/978-3-319-74727-9_22.

[10] de Chazal P, O'Dwyer M, Reilly RB. Automatic classification of heartbeats using ECG morphology and heartbeat interval features. IEEE Trans Biomed Eng 7 2004; 51(7):1196–1206. DOI: 10.1109/TBME.2004.827359.

[11] Dózsa T, Radó J, Volk J, Kisari A, Soumelidis A, Kovács P. Road abnormality detection using piezoresistive force sensors and adaptive signal models. IEEE Transactions on Instrumentation and Measurement 2022;71:1–11.

[12] Dózsa T, Böck C, Bognár G, Meier J, Kovács P. Color classification of visually evoked potentials by means of Hermite-functions. In Proc. of the 55st Annual Asilomar Conf. on Signals, Systems, and Computers. 2021; 251–255.

Address for correspondence:

Péter Kovács
Pázmány Péter sétány 1/C,
1117 Budapest, Hungary,
kovika@inf.elte.hu