



HAL
open science

An Inception-Based Architecture for Haemodialysis Time Series Classification

Giorgio Leonardi, S. Montani, Manuel Striani

► **To cite this version:**

Giorgio Leonardi, S. Montani, Manuel Striani. An Inception-Based Architecture for Haemodialysis Time Series Classification. 17th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), Jun 2021, Hersonissos, Crete, Greece. pp.194-203, 10.1007/978-3-030-79157-5_17. hal-03789016

HAL Id: hal-03789016

<https://inria.hal.science/hal-03789016v1>

Submitted on 27 Sep 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

An Inception-based Architecture for Haemodialysis Time Series Classification

Giorgio Leonardi, Stefania Montani, Manuel Striani

DISIT, Computer Science Institute, Università del Piemonte Orientale, Alessandria, Italy

Abstract. Classifying haemodialysis sessions, on the basis of the evolution of specific clinical variables over time, allows the physician to identify patients that are being treated inefficiently, and that may need additional monitoring or corrective interventions. In this paper, we propose a deep learning approach to clinical time series classification, in the haemodialysis domain. Specifically, grounding on our previous experience in adopting convolutional neural networks on haemodialysis time series, we have defined an inception-based architecture, able to exploit kernels of different sizes in parallel. The proposed architecture has outperformed the results obtained by resorting both to a more standard convolutional neural network, and to the state of the art approach ROCKET, since we reached higher accuracy values, coupled with a good Matthews Correlation Coefficient.

1 Introduction

End Stage Renal Disease (ESRD) is a severe chronic condition, which requires haemodialysis treatment. The efficacy of haemodialysis can be assessed on the basis of a few monitoring variables [2], regularly sampled during the treatment sessions (and thus recorded as time series). Among them, the behavior of the Haematic Volume (HV) is particularly important, because of its correlation to the water reduction rate and to the presence of cardiovascular alterations [17, 10]. Classifying haemodialysis sessions on the basis of HV evolution over time allows the physician to identify patients that are being treated inefficiently, and/or that may need additional monitoring or corrective intervention because they are experiencing problems.

In the past [8, 14], we proposed to afford this issue resorting to deep learning [13]. Specifically, we tested classical convolutional architectures, which are particularly suitable for time series data, due to their ability to model local dependencies that may exist between adjacent data points.

In this paper, we propose a more advanced convolutional solution, based on the exploitation of multiple instances of the so-called *inception* module [16].

In the following, we illustrate the network details and showcase our experiments, where the inception solution has outperformed both a standard convolutional neural network, and the state of the art tool ROCKET [4]: indeed, our

approach has reached higher accuracy values, coupled with a good Matthews Correlation Coefficient.

The paper is organized as follows: in section 2 we present background and related work; section 3 illustrates the proposed deep learning architecture; section 4 provides experimental results. Section 5 is devoted to discussion and conclusions.

2 Background and related work

Deep learning architectures are able to stack multiple layers of operations, in order to create a hierarchy of increasingly more abstract *deep* features [13]. These techniques have achieved a great success in computer vision, and also their adoption for time series data classification is gaining increasing attention in several domains [11, 18, 20], including health care (see, e.g., [6, 15, 8]).

In the following, we will introduce some background information about different deep learning architectures, as well as some related works dealing with their application to time series classification.

2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) operate by exploiting multiple convolution operators. A convolution is an operation which takes a filter and multiplies it over the entire area of the input. Convolution layers are followed by pooling layers, meant to further reduce dimensionality.

The convolution+pooling modules can be stacked in the network, providing progressively deeper architectures. The output of the final pooling layer is then typically flattened, and provided as an input to a fully connected network, outputting the class.

Composed of sparse connections with tied weights, CNNs have significantly fewer parameters than a fully connected network of similar size [1].

One-dimensional CNNs are particularly suitable for time series data, due to their ability to model local dependencies that may exist between adjacent data points [19], and to capture how the input evolves over time [12].

CNNs for time series classification have been proposed, e.g., in [3, 5, 15], and are the most popular deep learning approach in physiological signals classification, as testified in a recent survey [6]. We also obtained encouraging results in medical time series classification resorting to CNN in our previous work [8, 14].

2.2 Inception Networks

The inception module, proposed in [16] as a building block for the GoogLeNet architecture, uses kernels of varied size in a convolution layer to capture features at different levels of abstraction. Since salient parts in images and sequences can have extremely large variation in size, choosing the right kernel for the convolution operation becomes difficult: large kernels are suited for globally distributed information, while small kernels work well with locally distributed information.

The inception module processes information at different scales, by exploiting branches with different kernels in parallel, and then aggregates them to efficiently extract relevant features. The network essentially gets a bit wider rather than deeper. Inception mitigates the problem of correctly setting the kernel size, since it simultaneously applies many convolutions with different kernel sizes to the same input.

This approach has been successfully applied to the time series domain by the tool InceptionTime [7]. InceptionTime is however computationally expensive; as a further evolution, therefore, the tool ROCKET [4] has been proposed, which uses random convolutional kernels, and achieves satisfactory accuracy results while reducing computation time.

3 Inception-based classification

Our implementation of inception-based classification, depicted in figure 1, exploits a basic module (see the blue box in the figure) presenting three convolutions with kernels of sizes 1, 3, and 5, respectively, in parallel also to a 3 max-pooling path. The outputs of the parallel branches are then concatenated.

Three basic inception modules built as explained above are then cascaded in our proposed architecture, and passed through a final sigmoid layer for classification. Every convolution branch is composed by 16 kernels and is activated by the Rectified Linear Unit function. Parameters were set experimentally.

We have also implemented a simpler architecture for comparison. This architecture, depicted in figure 2, is a more standard CNN with 3 one-dimensional convolution layers, with a kernel size of 1, 3 and 5 respectively, Rectified Linear Unit activation function, and 16, 32 and 64 kernels respectively. Each convolution layer is followed by a max pooling layer with a pool size of 2. The output of the last pooling layer is provided to a layer using the sigmoid activation function to output a final node, representing the predicted class. The parameters values were set experimentally in this case as well.

4 Results

Our input HV time series were recordings of 240 samples on average, with a sampling time of 1 minute. We truncated longer series, and added zeros to extend shorter series. We worked with a dataset of 5914 time series, belonging to 75 different patients (72 series per patient on average, varying from 1 to 280).

Our classification was a binary one, where positive cases represent time series whose plot suggests an insufficient reduction of water and metabolites from the patient's blood, while negative cases are related to non-problematic haemodialysis sessions. Examples of a positive and negative case are shown in figure 3.

We performed the labeling process in two steps: first, each time series was de-noised through wavelet transform and its gradient was calculated over time to apply a first temporary label; then, the labeled time series were validated by

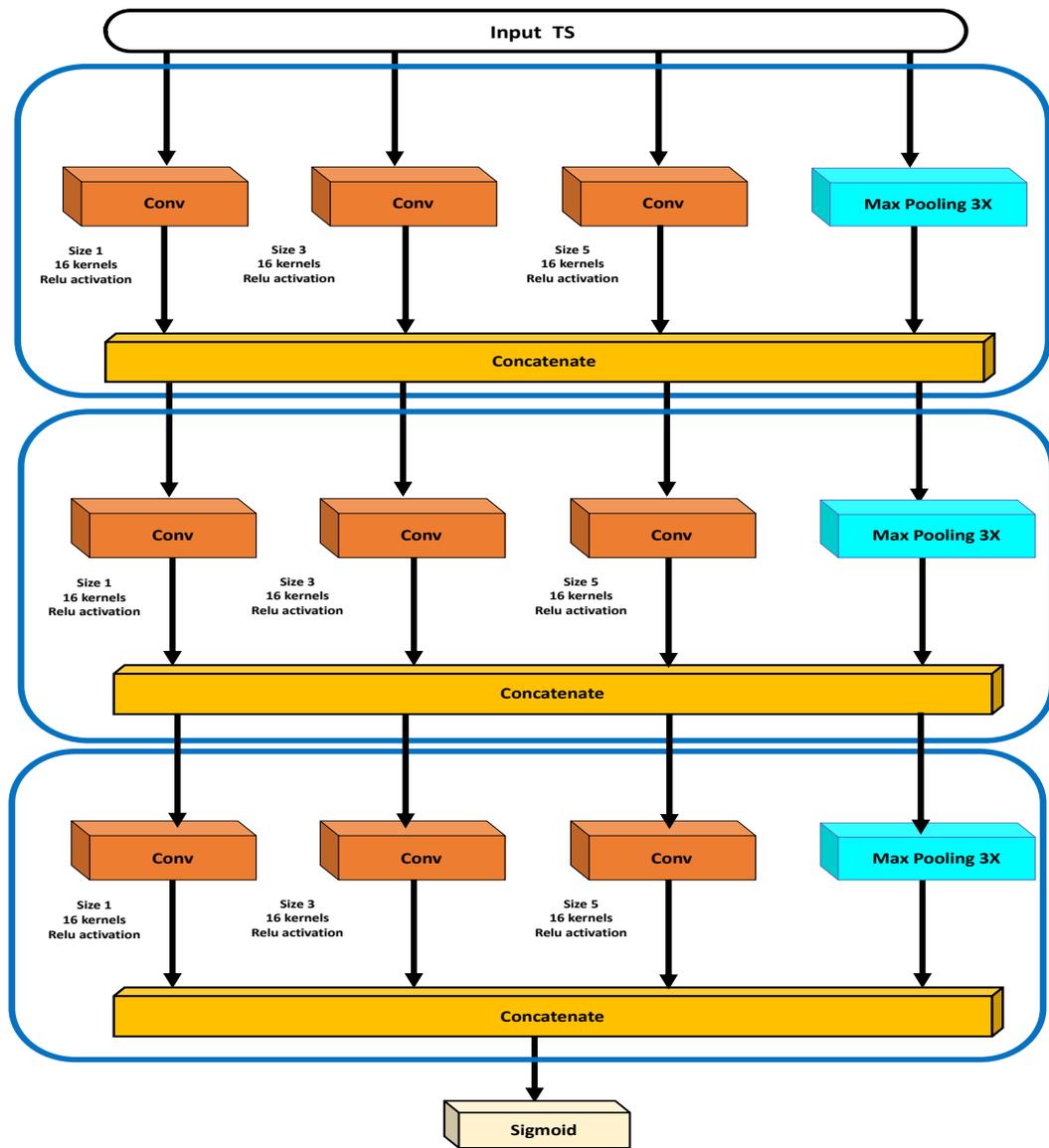


Fig. 1. Inception-based classification architecture

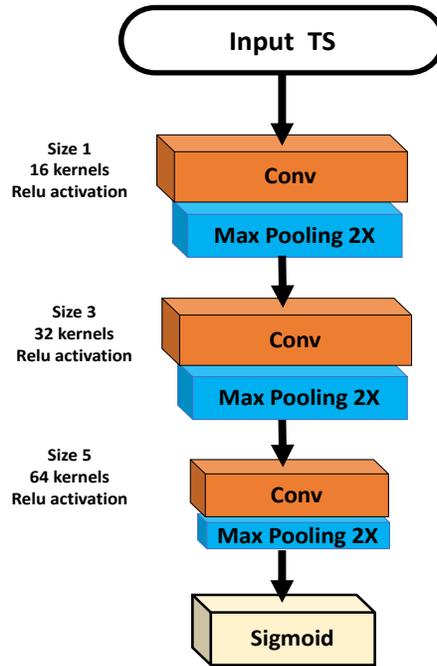


Fig. 2. Standard CNN architecture, implemented for comparison

medical experts to confirm or to correct the automatically assigned labels on the basis of domain knowledge. At the end of the process, 4048 negative cases and 1866 positive cases were made available.

For our experiments, we conducted a 10-fold cross validation.

Experiments were conducted by resorting to the TensorFlow tool¹.

We exploited a machine with the following characteristics: Operating System: Windows Server 2012 R2; Processor: Intel Xeon E3-12xx v2 (Ivy Bridge, IBRS) 2.70 GHz (2 processors); Installed memory (RAM): 8.00 GB; System Type: 64-bit Operating System, 64-based processor; Hard disk memory: 40 GB.

Working at 100 epochs provided a good compromise between quality of results and computation time, as shown in figures 4 and 5, which compare the evolution of the loss values over the number of epochs for the standard CNN and for the inception-based architecture, respectively.

The CNN approach provided an average accuracy of 76%. The complete results are shown in table 1, which also reports on precision, recall, F1-score, Matthews Correlation Coefficient (MCC, a parameter which is particularly suitable to assess the quality of classification when dealing with unbalanced classes, and should be ideally close to 1), and K-statistics. The validation results are provided for each class and as the weighted average by class cardinality, accord-

¹ <https://www.tensorflow.org/>

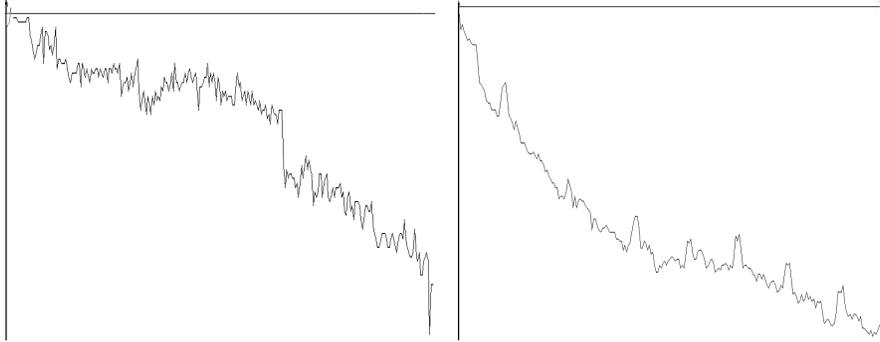


Fig. 3. Example of a positive case (left) and of a negative case (right)

ing to the unbalanced distribution of positive and negative cases. Class 0 refers to the positive cases, while class 1 refers to the negative ones. MCC, K-statistics and accuracy are not related to a single class, therefore we provide them only as overall results.

Table 1. Results obtained by the CNN-based classifier

Class	Precision	Recall	F1-score	MCC	K-Stat	Acc.
0 (positive)	0.65	0.52	0.58			
1 (negative)	0.79	0.87	0.83			
Weighted ave.	0.74	0.76	0.74	0.42	0.41	0.76

Inception-based classification (figure 1) performed very well, as shown in table 2. In particular, accuracy reached 88%, coupled with a MCC of 0.73, much higher than the 0.42 obtained by the standard CNN architecture.

Table 2. Results obtained by the inception-based classifier

Class	Precision	Recall	F1-score	MCC	K-Stat	Acc.
0 (positive)	0.84	0.78	0.81			
1 (negative)	0.90	0.93	0.91			
Weighted ave.	0.88	0.88	0.88	0.73	0.62	0.88

We also made a comparison to the state of the art literature approach ROCKET [4], presented in section 2.2². As shown in table 3, ROCKET re-

² We exploited the implementation provided at <https://github.com/angus924/rocket> (last accessed on March 23rd 2021). Note that in this case the number of epochs cannot be set.

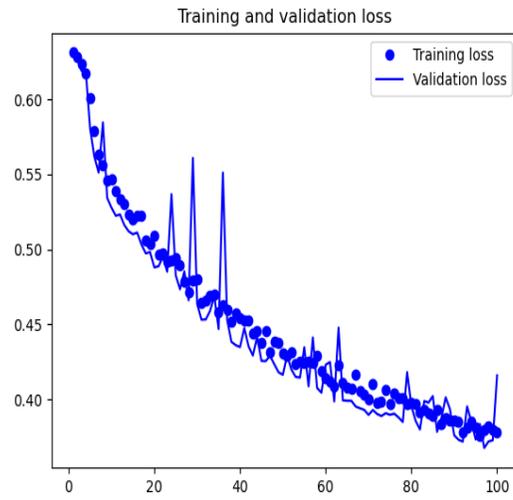


Fig. 4. Evolution of the loss values over the number of epochs for the standard CNN architecture

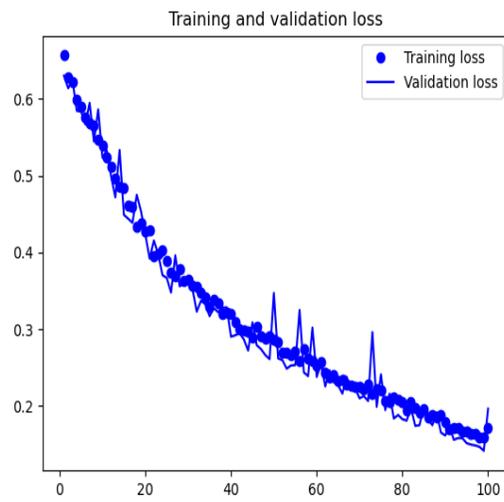


Fig. 5. Evolution of the loss values over the number of epochs for the inception-based architecture

sults were better than the ones of the basic CNN classifier, but poorer than the ones of our inception-based architecture.

Table 3. Results obtained by the ROCKET classifier

Class	Precision	Recall	F1-score	MCC	K-Stat	Acc.
0 (positive)	0.60	0.73	0.66			
1 (negative)	0.90	0.83	0.86			
Weighted ave.	0.80	0.80	0.80	0.53	0.53	0.80

As a final experiment, we tested a patient-based cross-validation (leave-one-patient-out). Each one of the three models tested above (namely, CNN-based architecture, inception-based architecture and ROCKET) was learned on all records from all patients except one, and tested on records from that single patient, and this procedure was repeated for all patients. This technique rules out overly optimistic results, that may be obtained when the dataset contains many records from a single patient, and the test is executed on some record of the same patient. The approach also allows one to mitigate the risk of overfitting [9].

The results obtained by the inception-based architecture (table 4) and by ROCKET (table 5) were (not surprisingly) worse than the corresponding ones already presented in table 2 and table 3, respectively, but still encouraging. Analogous considerations hold for the results of the CNN-based architecture (not reported due to lack of space).

Moreover, our proposed inception-based architecture outperformed the other tools in this experiment as well.

These findings support the hypothesis that our proposed architecture is actually performing well, and that high accuracy values should not be interpreted in the light of overfitting problems.

Table 4. Results obtained by the inception-based classifier with a leave-one-patient-out cross validation

Class	Precision	Recall	F1-score	MCC	K-Stat	Accuracy
0 (positive)	0.66	0.68	0.67			
1 (negative)	0.86	0.85	0.85			
Weighted average	0.79	0.79	0.79	0.52	0.52	0.79

5 Conclusions

HV time series classification can help physicians in identifying haemodialysis treatment inefficiency, allowing for early interventions that can lead to an overall optimization of patient care.

Table 5. Results obtained by the ROCKET classifier with a leave-one-patient-out cross validation

Class	Precision	Recall	F1-score	MCC	K-Stat	Accuracy
0 (positive)	0.50	0.70	0.58			
1 (negative)	0.90	0.80	0.85			
Weighted average	0.77	0.77	0.76	0.44	0.43	0.77

In this paper, we have proposed a deep learning approach for HV classification, based on the exploitation of multiple inception modules, defining the architecture illustrated in figure 1. This solution reached an 88% accuracy in cross validation, coupled with a high MCC (0.73), outperforming the more standard CNN architecture we implemented for comparison, and even the state of the art tool ROCKET [4].

In the future, we plan to conduct additional experiments, by considering other haemodialysis time series variables as well.

Moreover, since deep learning methods operate as black boxes, and it can be difficult to justify misclassification, we will also consider the issue of explainability. To this end, we will investigate whether it is possible to adapt a knowledge-based strategy, along the lines we already tracked in [14].

References

1. Alom, M.Z., Taha, T.M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M.S., Hasan, M., Van Essen, B.C., Awwal, A.A.S., Asari, V.K.: A state-of-the-art survey on deep learning theory and architectures. *Electronics* **8**(3) (2019)
2. Bellazzi, R., Larizza, C., Magni, P., Bellazzi, R.: Temporal data mining for the quality assessment of a hemodialysis service. *Artificial Intelligence in Medicine* **34**, 25–39 (2005)
3. Cui, Z., Chen, W., Chen, Y.: Multi-scale convolutional neural networks for time series classification. *CoRR* **abs/1603.06995** (2016)
4. Dempster, A., Petitjean, F., Webb, G.I.: Rocket: Exceptionally fast and accurate time series classification using random convolutional kernels (2019)
5. Fan, X., Yao, Q., Cai, Y., Miao, F., Sun, F., Li, Y.: Multiscaled fusion of deep convolutional neural networks for screening atrial fibrillation from single lead short ECG recordings. *IEEE J. Biomedical and Health Informatics* **22**(6), 1744–1753 (2018)
6. Faust, O., Hagiwara, Y., Hong, T.J., Lih, O.S., Acharya, U.R.: Deep learning for healthcare applications based on physiological signals: A review. *Computer Methods and Programs in Biomedicine* **161**, 1 – 13 (2018)
7. Fawaz, H.I., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D.F., Weber, J., Webb, G.I., L-Idoumghar, Muller, P., Petitjean, F.: Inceptiontime: Finding alexnet for time series classification (2019)
8. G. Leonardi, S. Montani, M.S.: Deep learning for haemodialysis time series classification. In: Proc. KR4HC/ProHealth and TEAAM workshops, LNCS (to appear). Springer (2019)
9. Gao, Y., Zhang, X., Wang, S., Zoub, G.: Model averaging based on leave-subject-out cross-validation. *Journal of Econometrics* **192**, 139–151 (2016)

10. Krepel, H.P., Nette, R.W., Akcahuseyin, E., Weimar, W., Zietse, R.: Variability of relative blood volume during hemodialysis. *Nephrology, dialysis, transplantation : official publication of the European Dialysis and Transplant Association - European Renal Association* **15**, 673–9 (05 2000). <https://doi.org/10.1093/ndt/15.5.673>
11. Långkvist, M., Karlsson, L., Loutfi, A.: A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters* **42**, 11–24 (2014)
12. Lea, C., Vidal, R., Reiter, A., Hager, G.D.: Temporal convolutional networks: A unified approach to action segmentation. In: Hua, G., Jégou, H. (eds.) *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III. Lecture Notes in Computer Science*, vol. 9915, pp. 47–54 (2016)
13. LeCun, Y., Bengio, Y., Hinton, G.E.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
14. Leonardi, G., Montani, S., Striani, M.: Deep feature extraction for representing and classifying time series cases: towards an interpretable approach in haemodialysis. In: *Proceedings of the 33rd International Florida Artificial Intelligence Research Society Conference, FLAIRS 2020, Miami, Florida. AAAI Press (2020)*
15. Sani, S., Wiratunga, N., Massie, S., Cooper, K.: KNN sampling for personalised human activity recognition. In: Aha, D.W., Lieber, J. (eds.) *Case-Based Reasoning Research and Development - 25th International Conference, ICCBR 2017, Trondheim, Norway, June 26-28, 2017, Proceedings. Lecture Notes in Computer Science*, vol. 10339, pp. 330–344. Springer (2017)
16. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015. pp. 1–9. IEEE Computer Society (2015)*
17. Titapiccolo, J.I., Ferrario, M., Garzotto, F., Cruz, D., Moissl, U., Tetta, C., Ronco, C., Signorini, M., Cerutti, S.: Relative blood volume monitoring during hemodialysis in end stage renal disease patients. *Conference proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference* **2010**, 5282–5 (08 2010). <https://doi.org/10.1109/IEMBS.2010.5626307>
18. Wang, H., Sun, W., Liu, P.X.: Adaptive intelligent control of nonaffine nonlinear time-delay systems with dynamic uncertainties. *IEEE Trans. Systems, Man, and Cybernetics: Systems* **47**(7), 1474–1485 (2017)
19. Wang, Z., Yan, W., Oates, T.: Time series classification from scratch with deep neural networks: A strong baseline. In: *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017. pp. 1578–1585. IEEE (2017)*
20. Zhao, X., Shi, P., Zheng, X., Zhang, J.: Intelligent tracking control for a class of uncertain high-order nonlinear systems. *IEEE Trans. Neural Netw. Learning Syst.* **27**(9), 1976–1982 (2016)