Paulo Roberto DE MOURA JÚNIOR, William LIAW

Analyzing and Extending Time Series Kernels based on Nonlinear Vector Autoregressive Delayed Embeddings

This report is based on the paper (FELICE; GOULERMAS; GUSEV, 2023), published at NeurIPS 2023.

Advisor: Prof. Florence D'Alché

Palaiseau, France
2024

ABSTRACT

In this work, we study a Nonlinear Vector Autoregressive (NVAR) kernel for time-series and its use in the supervised classification framework, as proposed at NeurIPS 2023 (FELICE; GOULERMAS; GUSEV, 2023). The NVAR kernel represents a novel approach to time-series kernel design, combining principles from reservoir computing (RC) with delay embedding techniques rooted in dynamical systems theory. This integration enables efficient and interpretable representation of both univariate and multivariate time-series data, addressing challenges often encountered with traditional RC-based models.

Our study involves implementing and evaluating the NVAR kernel on three distinct datasets: Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) and Berlin Database of Emotional Speech (Emo-DB) for emotion classification task and finally Human Activity Recognition Using Smartphones (HAR) for human activity recognition task. Performance is assessed through accuracy, precision, recall, and F1-score, with additional analyses on computational efficiency and robustness to class imbalances. Results indicate that while the NVAR kernel performs well with high-arousal emotions and distinct physical activities, it faces challenges with nuanced emotional states and static activities. These findings suggest potential areas for enhancement, particularly in adapting the kernel for complex, high-dimensional, and imbalanced datasets. Finally, the NVAR kernel proves to be a viable approach for time-efficient pipelines, as it avoids the extensive training and inference required by neural networks, making it suitable for problems where a lower accuracy is acceptable.

Keywords: NVAR kernel, time-series classification, reservoir computing, delay embeddings, human activity recognition, emotion recognition.

CONTENTS

1 INTRODUCTION

In recent years, kernel methods have become a cornerstone of machine learning, especially for tasks involving non-linear data structures. Kernels enable linear algorithms to operate in transformed feature spaces, allowing for the efficient handling of complex relationships within data. Time-series data, with its inherent sequential dependencies, is a particularly challenging domain for machine learning, demanding specialized methods to capture temporal patterns and underlying dynamics, and arguably one of the most important data types in the moders era (HAMILTON, 1994; STROGATZ, 2018; ZHANG, 2017; ZEROUAL et al., 2020). As traditional kernel approaches often struggle with the complexity of time-series data, recent research has explored innovative kernel designs tailored specifically for this data type.

The selected paper, “Time Series Kernels based on Nonlinear Vector AutoRegressive Delay Embeddings” (FELICE; GOULERMAS; GUSEV, 2023), addresses a major challenge in time-series kernel design by introducing a new kernel based on Nonlinear Vector AutoRegressive (NVAR) delay embeddings. The proposed NVAR kernel draws on the principles of reservoir computing (RC), adapting them into a more interpretable and computationally efficient framework (BOLLT, 2021). Unlike standard RC-based kernels, which rely on recurrent structures and complex hyperparameter tuning, the NVAR kernel leverages non-recursive embeddings. This approach reduces the dependency on recurrent hyperparameters, making it better suited for classification tasks with small datasets or time-efficient pipelines, where deep learning techniques may not be feasible.

This project aims to conduct a comprehensive analysis of the NVAR kernel’s effectiveness and performance for classification tasks. By comparing the NVAR kernel performance with benchmarks for new selected datasets, we seek to evaluate its advantages and limitations in terms of both accuracy and computational efficiency.

This report is structured as follows: First, we present an in-depth analysis of the NVAR kernel and its unique contributions to time-series classification. We then describe the methodology, experimental setup, followed by the results of testing the ker-

nel for the classification task within distinct datasets and a critical evaluation of the method's strengths and weaknesses. Finally, we discuss potential future directions for kernel design in time-series analysis and conclude with key insights drawn from this study.

2 ANALYSIS

2.1 Novelty and Contributions

The paper introduces a novel approach to time-series kernel design using NVAR. This method stands out for its integration of kernel design techniques, RC principles and the NVAR framework, a combination that enhances both interpretability and efficiency in time-series data analysis. Traditional RC-based kernels rely heavily on recurrent structures that are complex and sensitive to hyperparameter tuning, often demanding high computational resources to achieve optimal performance. In contrast, the NVAR kernel circumvents the need for recurrence by structuring embeddings as non-recursive transformations of the input data: time-delays and nonlinear functionals, such as products (GAUTHIER et al., 2021).

This approach not only simplifies the model architecture but also reduces the dependency on interpretatively opaque hyperparameters, making it easier to apply and understand. Thus, it represents a significant advancement in kernel design by developing an NVAR-based kernel suitable for both univariate (UTS) and multivariate time-series (MTS) data, with parameter settings guided by simple heuristics. Experiments across diverse datasets show that this NVAR kernel achieves accuracy comparable to the state-of-the-art (SOTA) kernels for time-series classification while offering a substantial improvement in computational efficiency. This balance between accuracy and speed highlights its practicality for real-world applications, especially where computational resources are limited.

From the perspective of RC, the NVAR kernel introduces a non-recursive model that circumvents the challenges of hyperparameter optimization typically faced with RC-based kernels. By forgoing recurrence, the NVAR kernel remains interpretable and simplifies the model architecture without sacrificing the quality of representation. This change not only enhances interpretability but also positions the NVAR kernel as a more accessible tool for practitioners who might otherwise face the complexities and resource demands of RC.

The paper also expands the use of NVAR beyond its traditional role in forecast-

ing chaotic, noise-free systems, demonstrating its applicability to real-world time-series data. By connecting the method to foundational principles in dynamical systems theory, including Takens' theorem (TAKENS, 1981) and state-space reconstruction, the paper provides a theoretical basis for understanding the embeddings used in the NVAR kernel. This connection to established theory underscores the method's robustness and supports its potential as a versatile tool in machine learning for tasks requiring sophisticated temporal representations.

2.2 Methodology

The methodology of the NVAR kernel is structured around transforming time-series data using a series of lagged embeddings and nonlinear transformations, creating a high-dimensional representation that effectively captures the underlying temporal dynamics. Specifically, the kernel leverages delay embeddings, a technique rooted in dynamical systems theory, to form feature vectors that encapsulate the past states of the time-series data. These feature vectors are then mapped to a high-dimensional space, where a similarity measure is calculated to define the kernel.

This embedding process follows Takens' theorem, which suggests that a time-delayed version of a series can reveal its latent dynamics. In this framework, each time series is enriched with delayed copies and nonlinear combinations of the input, allowing the NVAR kernel to uncover complex patterns and dependencies that traditional kernels might overlook. The paper further simplifies the hyperparameter space by employing a heuristic-based approach for setting the lag length and polynomial order, avoiding the computational burden typically associated with exhaustive tuning.

The NVAR kernel also builds on the kernel trick, which allows the computation of inner products in a transformed feature space without explicitly computing the transformation. By integrating the NVAR embedding structure into the kernel trick, the paper effectively combines the advantages of kernel-based methods with those of RC, resulting in a versatile and efficient approach to time-series analysis.

2.3 Comparison to State-of-the-Art

The proposed NVAR kernel distinguishes itself from existing time-series similarity measures and kernel methods, particularly RC-based kernels, such as Echo State Networks (ESNs) (JAEGER, 2001), that rely on recurrent structures that introduce complexity in hyperparameter optimization and high sensitivity to initial conditions. These recurrent models often struggle with interpretability, as their performance heavily depends on tuning multiple hyperparameters related to the reservoir's size, connectivity, and spectral properties.

In comparison, the NVAR kernel uses a non-recursive structure and reduces the number of hyperparameters that require fine-tuning. Unlike elastic measures, which directly measure similarity in the input space and can overlook deeper temporal dependencies, the NVAR kernel captures underlying dynamics through delay embeddings, improving its ability to handle time distortions and shifts. Additionally, while model-based kernels such as the Time Cluster Kernel (TCK) (MIKALSEN et al., 2018) can achieve high accuracy, they tend to be computationally intensive and may not scale well with larger datasets.

2.4 Strengths

The strengths of the NVAR kernel are evident in its computational efficiency, execution time, scalability, and suitability for small datasets. The non-recursive nature of the kernel design eliminates the iterative computations required in recurrent models, allowing it to scale linearly with the number of time series and reducing overall computational cost, as presented by the low median time per dataset, presented on the Table 1 of the original paper. This efficiency makes the NVAR kernel particularly valuable in contexts where fast processing is essential, such as real-time monitoring or applications with limited computational resources.

Moreover, the simplicity of the NVAR kernel's hyperparameters contributes to its scalability and ease of use. By relying on a heuristic for setting lag and polynomial order, the kernel can be quickly adapted to a variety of datasets without extensive optimization, a feature that is especially useful when working with small datasets where

overfitting risks are higher. The interpretability of the NVAR approach also adds to its strengths, as the non-recursive embeddings make it easier to understand and analyze the extracted temporal features compared to traditional RC methods.

2.5 Weaknesses

Despite its advantages, the NVAR kernel has some limitations. One potential drawback is its reliance on heuristic-based hyperparameter settings, which may not always yield optimal results, particularly for highly complex datasets or data with irregular temporal patterns. While the heuristics provide a practical solution, they may oversimplify the selection of critical parameters such as lag size, leading to suboptimal embeddings in certain cases.

Another limitation lies in the method's performance with high-dimensional datasets. As the number of dimensions in the input data increases, the kernel may face challenges due to the curse of dimensionality, potentially requiring adjustments to its feature selection strategy. Additionally, the NVAR kernel's dependence on delay embeddings may restrict its applicability in situations where the time-series data lacks well-defined temporal dependencies or exhibits chaotic behavior that is difficult to model with fixed embeddings.

3 METHODOLOGY

3.1 Datasets

Selecting appropriate datasets is pivotal in the development, training, and evaluation of recognition models. In this study, we employ three widely recognized datasets: the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS), the Berlin Database of Emotional Speech (Emo-DB), and the Human Activity Recognition Using Smartphones (HAR) dataset. While RAVDESS and Emo-DB are commonly used in speech emotion classification—featuring emotionally expressive performances by professional actors in a UTS—the HAR dataset represents a MTS dataset collected from wearable sensors for activity recognition. The following subsections provide detailed descriptions of each dataset, highlighting their key features, structures, and the rationale for their use in this study.

3.1.1 Ryerson Audio-Visual Database of Emotional Speech and Song

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) (LIVINGSTONE; RUSSO, 2018) is a widely recognized benchmark dataset utilized extensively in audiovisual emotion classification research (ANUSHA et al., 2021; VIMAL et al., 2021; ABDULLAH; AHMAD; HAN, 2020). The dataset consists of short audio and video recordings that feature both spoken and sung performances, enacted by a cohort of 24 actors (12 male and 12 female). Each recording is labeled with one of the following emotion categories: *angry*, *calm*, *disgust*, *fearful*, *happy*, *neutral*, *sad*, and *surprised*.

To promote consistency and reproducibility, each actor delivers two predefined phrases in English: “Kids are talking by the door” and “Dogs are sitting by the door.” Apart from the neutral category, all emotions are expressed at two distinct intensity levels (normal and strong), with each instance repeated twice. These structured variations in emotional intensity, repetition, and diversity of vocal expressions make RAVDESS an invaluable asset for the development and validation of emotion recognition models in a wide range of applications.

For the audio-only subset of the dataset which we employ for further analysis in the present work, there are a total of 1440 speech recordings and 1012 song recordings. It is worth noting that the singing subset is slightly smaller, as one actor's data is missing, and the emotions *sad* and *surprised* are not included for singing performances.

Despite its favorable reception within the academic community, as demonstrated by its widespread adoption, evidence suggests that the application of RAVDESS in real-world scenarios may lead to underwhelming results (CHURAEV; SAVCHENKO, 2021). One possible explanation for this discrepancy is the issue of data leakage. Specifically, an overlap of similar samples between the training and validation sets may result in unintended information sharing, thereby artificially inflating performance metrics. This overestimation does not accurately reflect the generalizability and practical effectiveness of models when deployed in real-world environments.

3.1.2 Berlin Database of Emotional Speech

The Berlin Database of Emotional Speech (Emo-DB) (BURKHARDT et al., 2005), akin to RAVDESS, is a well-regarded dataset for speech emotion classification tasks (SINITH et al., 2015; KOTTI; KOTROPOULOS, 2008; YING; ZHANG, 2010). It comprises short spoken audio recordings performed by 10 professional actors (5 male and 5 female), each enacting various grammatical phrases in German, as detailed in Table 1. Each recording is annotated with one of the following emotion categories: *anger*, *anxiety/fear*, *boredom*, *disgust*, *happiness*, *neutral*, and *sadness*.

To ensure the quality and reliability of the dataset, these samples underwent evaluation by a significant number of listeners, who assessed the naturalness of the emotional expressions. In total, the dataset comprises 535 speech files.

Table 1: Grammatical phrases in the Emo-DB dataset

German	English
Der Lappen liegt auf dem Eisschrank.	The cloth is on the refrigerator.
Das will sie am Mittwoch abgeben.	She will deliver it on Wednesday.

Continued on the next page

German	English
Heute abend könnte ich es ihm sagen.	Tonight I could tell him.
Das schwarze Stück Papier befindet sich da oben neben dem Holzstück.	The black sheet of paper is located up there next to the piece of wood.
In sieben Stunden wird es soweit sein.	In seven hours it will be time.
Was sind denn das für Tüten, die da unter dem Tisch stehen?	What about the bags that are under the table?
Sie haben es gerade hochgetragen und jetzt gehen sie wieder runter.	They just carried it upstairs and now they are going back down.
An den Wochenenden bin ich jetzt immer nach Hause gefahren und habe Agnes besucht.	On weekends, I now always went home and visited Agnes.
Ich will das eben wegbringen und dann mit Karl was trinken gehen.	I will just take this away and then go have a drink with Karl.
Die wird auf dem Platz sein, wo wir sie immer hinlegen.	It will be in the place where we always put it.

Source: Own authorship

3.1.3 Human Activity Recognition Using Smartphones

The Human Activity Recognition Using Smartphones (HAR) dataset (REYES-ORTIZ et al., 2013) represents a distinct category of data compared to the speech-focused RAVDESS and Emo-DB datasets. While the latter are centered around audio signals for emotion recognition, the HAR dataset consists of MTS data collected from wearable sensors, specifically accelerometers and gyroscopes embedded in smartphones. This fundamental difference not only sets it apart in terms of data modality but also introduces unique challenges and considerations in the modeling process.

The dataset was constructed using data from 30 volunteers aged between 19 and 48 years. Each participant performed six predefined physical activities while carrying a waist-mounted Samsung Galaxy S II smartphone. The activities included: *walking*, *walking upstairs*, *walking downstairs*, *sitting*, *standing*, and *laying*.

Sensor signals were recorded at a constant sampling rate of 50 Hz, capturing 3-axial linear acceleration and 3-axial angular velocity. The raw sensor data underwent preprocessing steps, including noise filtering and normalization. Subsequently, the data were segmented into fixed-width sliding windows of 2.56 seconds (equivalent to 128 readings per window) with a 50% overlap between consecutive windows. This segmentation resulted in a rich set of time series samples that capture the dynamic patterns associated with each physical activity.

The MTS nature of the HAR dataset introduces complexities not present in UTS data like audio signals. Modeling such data requires capturing not only temporal dependencies but also the interrelationships between different sensor modalities. This necessitates advanced techniques capable of handling high-dimensional inputs and learning intricate patterns across multiple variables.

In this study, the HAR dataset serves as a means to evaluate the adaptability and robustness of the NVAR kernel when applied to MTS data. For consistency and to focus on the raw data's representational capacity, we utilize the HAR dataset without additional feature engineering.

3.2 Data Preprocessing

Preprocessing is a critical step to ensure that the audio data from the RAVDESS and Emo-DB datasets are in a suitable format for analysis and modeling. The preprocessing pipeline was implemented using the `librosa` library in Python (MCFEE et al., 2015), and it consisted of the following steps:

3.2.1 Audio Loading and Resampling

All audio files were loaded in mono format at a standardized sampling rate of 8 kHz using the `librosa.load` function and cropped to a maximum duration of 3 seconds. This resampling helps to reduce computational complexity and ensures consistency across all audio samples.

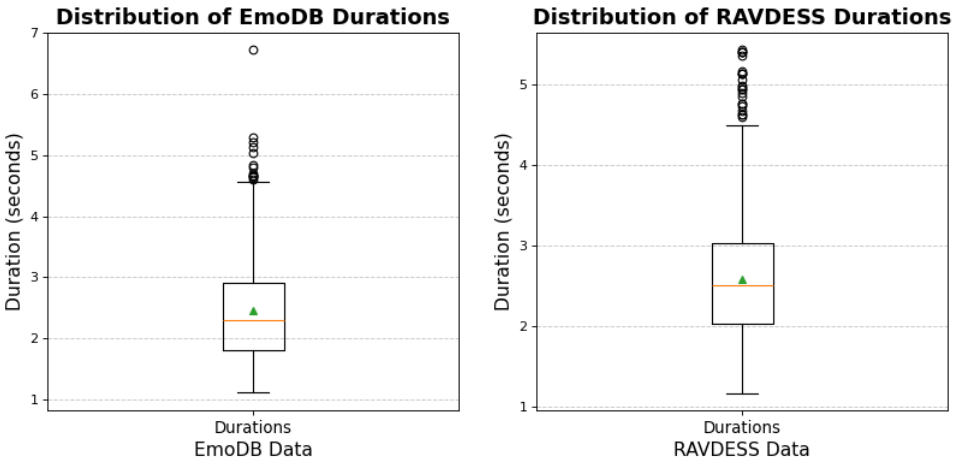

Figure 1 - Durations Box Plot

3.2.2 Vocal Separation

To isolate the vocal components and reduce background noise, an adaptive filtering technique was applied. This process began with computing the short-time Fourier transform (STFT) of the audio signal to obtain its magnitude and phase components. Non-negative matrix factorization (NMF) filtering was then used to separate the vocal content from other elements. A soft mask was subsequently applied to enhance the separation between vocal and non-vocal elements, and, finally, the time-domain signal was reconstructed using the inverse STFT.

3.2.3 Silence Removal

To focus on the significant parts of the speech and eliminate silence or low-amplitude sections, the audio signals were trimmed based on an amplitude threshold. This was done using the `librosa.effects.split` function, which identifies intervals where the signal is above a certain decibel level (`top_db` parameter). The segments were then concatenated to form the final processed signal.

3.2.4 Padding and Alignment

For samples shorter than the desired duration, zero-padding was applied to align all audio samples to a uniform length. This ensures that the input data has consistent

dimensions, which is essential for batch processing in machine learning models.

3.2.5 Final Preprocessing Pipeline

The entire preprocessing routine was encapsulated in a function that loads the audio file, applies vocal separation, removes silence, and pads the signal as needed.

3.3 Evaluation Method and Metrics

We employed the proposed NVAR kernel for classifying both UTS and MTS, using a Support Vector Machine (SVM) classifier with 10-fold cross-validation to fine-tune the hyperparameter C. The performance was evaluated using either accuracy or weighted F1-score as the scoring metric for cross-validation. Each classification experiment was repeated over 10 iterations per dataset, with the cross-validation data split randomized in each iteration to ensure robustness. This evaluation pipeline, as suggested by the original authors, was applied to our selected UTS datasets (RAVDESS and Emo-DB) and the MTS dataset (HAR). All experiments were conducted in our personal computers with limited hardware.

To assess classification performance on both training and testing sets, we computed accuracy, weighted precision, weighted recall, and weighted F1-score. These metrics allowed us to account for class imbalances within the datasets, ensuring a balanced evaluation across all classes. Accuracy was included to facilitate comparisons with existing methods in the literature, while precision, recall, and F1-score provided a comprehensive view of the classifier's effectiveness across imbalanced classes. Additionally, we record the total running time for training and testing each dataset to evaluate the computational efficiency of the proposed method.

To further examine the impact of class imbalance on classification performance, we computed the confusion matrix and analyzed the class distribution for the testing sets. This analysis helped identify any biases arising from imbalanced datasets and provided insights into the classifier's strengths and weaknesses in distinguishing between classes.

4 RESULTS

This section presents the results of our classification experiments, with performance visualized through tables and confusion matrices for the three distinct datasets: RAVDESS, Emo-DB, and HAR. We also analyze the computational efficiency of the present method in terms of running time.

In our experiments, fine-tuning the hyperparameter C through cross-validation using either accuracy or weighted F1-score produced identical results. Consequently, all further visualizations and analyses were reached using accuracy as the primary scoring metric, exactly as it was done in the original work.

4.1 RAVDESS

The clasification results for RAVDESS dataset can be seen in Table 2. The total running time of the pipeline (training and testing) was about 15 minutes.

Table 2 - Classification Results (%) over 10 random iterations - RAVDESS dataset

		Accuracy	Precision	Recall	F1
train	Mean	39.8966	39.6893	39.8966	39.2788
	Std	0.3860	0.3254	0.3860	0.3383
test	Top	41.752	41.540	39.482	-

For the RAVDESS dataset, which involves the classification of emotional states, the confusion matrix and class distribution for the testing set are presented in Figures 2 and 3, respectively.

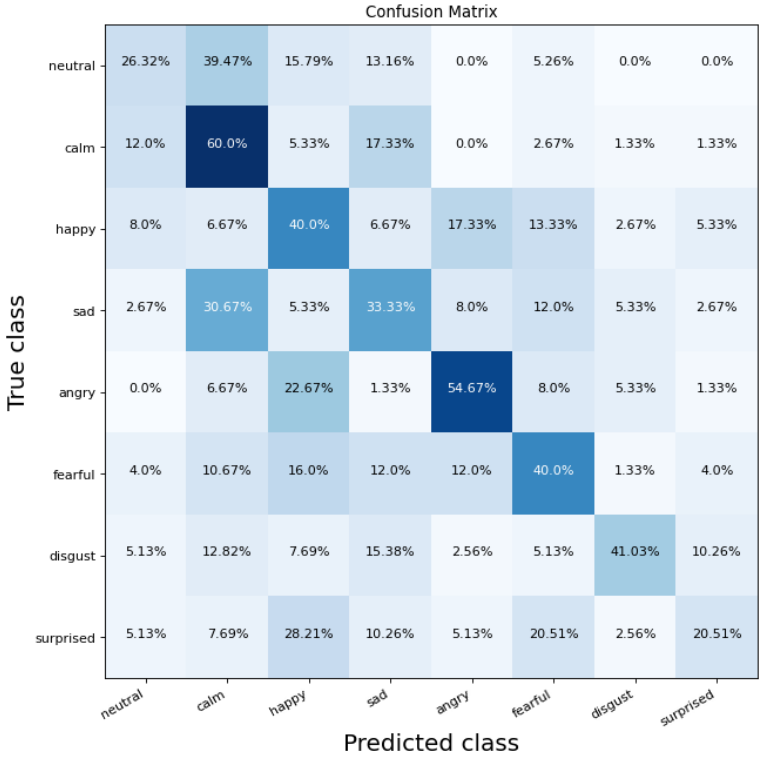

Figure 2 - Confusion matrix for RAVDESS dataset

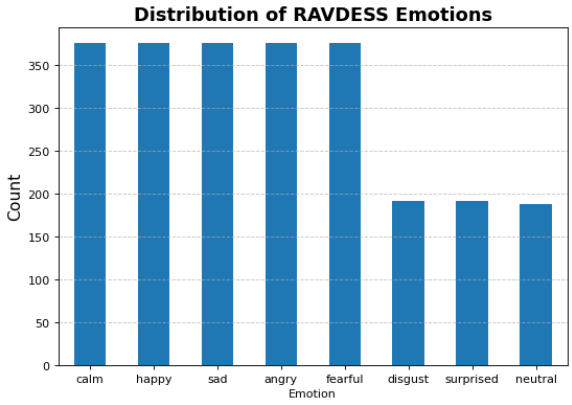

Figure 3 - Distribution of classes for RAVDESS dataset

The classification results for the RAVDESS dataset reveal varying levels of accuracy across different emotional classes. The emotions *Calm* and *Neutral* demonstrate moderate recognition accuracy, with *Calm* achieving 60.0% accuracy and *Neutral* showing 26.32%. However, *Neutral* is frequently misclassified as *Calm* (39.47%) or other emotions, highlighting difficulties in distinguishing subtle emotional nuances. The emotion *Angry* is relatively well-recognized, achieving 54.67% accuracy, though it is occasionally confused with other high-arousal emotions such as *Fearful* and *Dis-*

gust. Conversely, emotions like *Surprised* and *Sad* exhibit lower recognition accuracy, with *Surprised* correctly identified only 20.51% of the time, often being confused with *Happy* and *Fearful*. These findings suggest that the classifier struggles to accurately differentiate between emotions that exhibit subtle or overlapping features.

The results suggest that while the classifier is reasonably effective at identifying distinct high-arousal emotions like *Angry*, it struggles with emotions that have subtle or overlapping features, such as *Surprised* and *Fearful*, *Calm*, and *Neutral*. Moreover, we expect in a biased scenario that the model would try to assign the most frequent labels for the least populated classes (*Neutral, Disgust, Surprised*) but in general this was not observed here.

4.2 Emo-DB

The classification results for Emo-DB dataset can be seen in Table 3. The total running time of the pipeline (training and testing) was about 3 minutes.

Table 3 - Classification results (%) over 10 random iterations - Emo-DB

		Accuracy	Precision	Recall	F1
train	Mean	49.6901	48.2375	49.6901	47.9361
	Std	0.8321	0.6664	0.8321	0.7483
test	Top	53.271	51.690	50.782	-

The Emo-DB dataset's confusion matrix and class distribution for the testing set are shown in Figures 4 and 5, respectively.

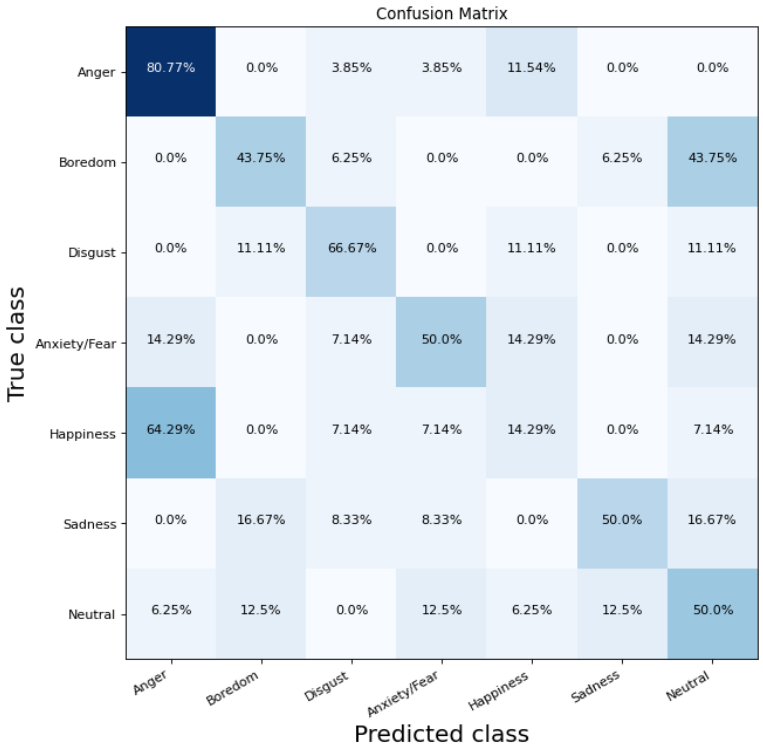

Figure 4 - Confusion matrix for Emo-DB

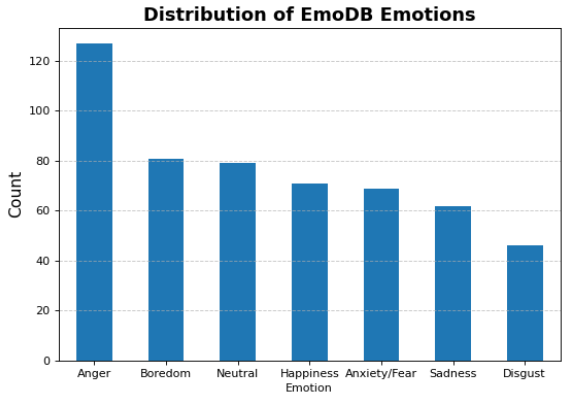

Figure 5 - Distribution of classes for Emo-DB

The emotion *Anger* achieves high recognition accuracy, with 80.77% of *Anger* samples correctly classified. This aligns with the trend observed in the RAVDESS dataset, where high-arousal emotions are more easily identified. Additionally, *Anger* is the majority class in this database, which may contribute to its higher classification performance. In contrast, lower recognition accuracy is observed for emotions such as *Boredom* and *Happiness*. For instance, *Boredom* is correctly identified only 43.75% of the time and is frequently misclassified as *Neutral*. These results highlight the classi-

fier's challenges in distinguishing lower-arousal or overlapping emotional states while excelling in recognizing dominant, high-arousal emotions.

These results highlight that high-arousal emotions such as *Anger* are more readily distinguishable, while lower-arousal and subtle emotions like *Neutral* and *Boredom* are often misclassified, suggesting a need for feature enhancement or additional data to improve classification in these categories. There is also evidence of influence of the class distributions on the classification performance, as the model tends to misclassify emotions as the majority class *Anger*.

4.3 HAR

The clasification results for HAR dataset can be seen in Table 4. The total running time of the pipeline (training and testing) was about 55 minutes.

Table 4 - Classification results (%) over 10 random iterations - HAR

		Accuracy	Precision	Recall	F1
train	Mean	95.9720	95.9905	95.9720	95.9749
	Std	0.2461	0.2474	0.2461	0.2465
test	Top	93.247	93.484	93.524	-

The confusion matrix for the HAR dataset, shown in Figure 6, demonstrates strong classification performance across most activity classes. The class distribution, illustrated in Figure 7, reveals a more balanced dataset, contributing to the overall high performance of the classifier.

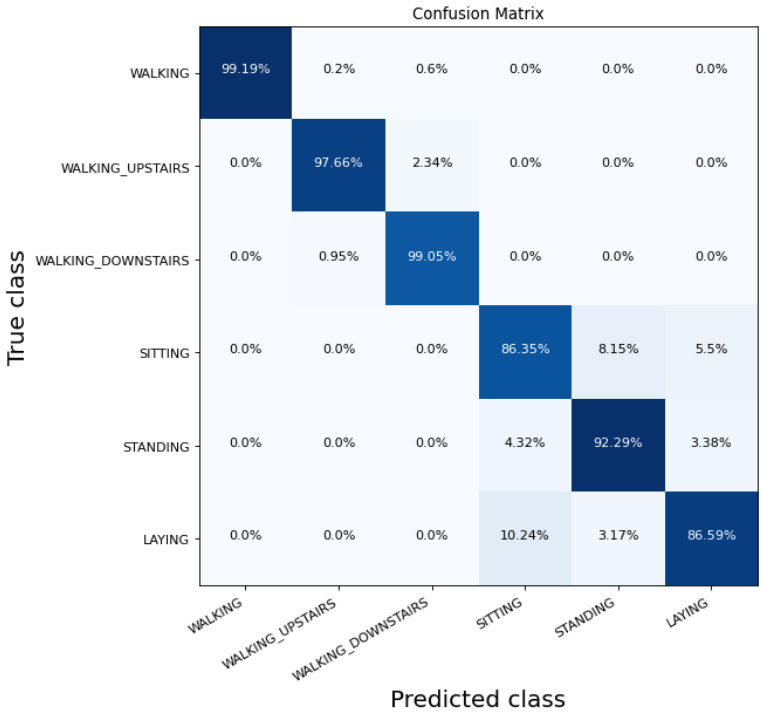

Figure 6 - Confusion matrix for HAR dataset

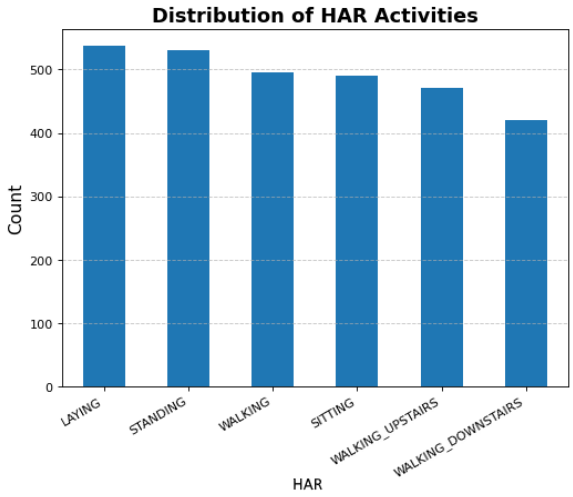

Figure 7 - Confusion Matrix for HAR dataset

The classifier exhibits excellent performance in identifying dynamic activities such as *Walking*, with the *Walking* and *Walking_Upstairs* classes achieving 99.19% and 99.66% accuracy, respectively. However, some confusion is observed between static activities like *Standing* and *Sitting*, with *Sitting* being misclassified as *Standing* in 8.15% of cases, highlighting challenges in distinguishing between these similar postures. Additionally, while the activity *Laying* is generally well-recognized, it has a

10.24% misclassification rate with *Sitting*, possibly due to overlapping features in body position. These results indicate that the classifier performs exceptionally well with dynamic activities but could benefit from further refinement in distinguishing static activities.

Overall, the classifier demonstrates high accuracy in recognizing dynamic activities (e.g., walking variations) but shows slight confusion between certain stationary activities, which could be an area for further optimization.

4.4 Summary of results

Across the three datasets, the classifier generally excels in recognizing dynamic activities and high-arousal emotions but encounters difficulties with classes that exhibit subtle distinctions or overlapping features. Dynamic activities, such as walking variations in the HAR dataset, and high-arousal emotions, such as anger in the Emo-DB and RAVDESS datasets, are consistently well-recognized. This is likely due to these classes possessing distinct and easily separable features, which the NVAR kernel can effectively capture.

In contrast, classes with subtle or overlapping features, such as static activities (e.g., sitting vs. standing) and low-arousal or nuanced emotions (e.g., calm vs. neutral, surprised vs. fearful), exhibit higher misclassification rates. These challenges reflect one of the limitations of the NVAR kernel—its heuristic-based approach to hyperparameter selection may not always yield optimal representations for complex or subtle classes, leading to suboptimal embeddings in such cases.

The results also indicate a noticeable tendency for the classifier to favor the majority class in each dataset, as observed with *Anger* in the Emo-DB dataset. This suggests that the NVAR kernel may be sensitive to class imbalances, potentially due to its reliance on delay embeddings that may not fully account for unequal class distributions. To mitigate this bias towards more frequent classes, additional measures, such as improved class weighting or synthetic data augmentation, could be implemented.

Lastly, no significant differences were observed between training and testing set results, indicating that overfitting was not an issue during the training process. This

consistency between the training and testing phases highlights the robustness of the implemented classification pipeline.

5 CONCLUSION

This study provides a comprehensive evaluation of the NVAR kernel for time-series classification, building on the innovative approach of (FELICE; GOULERMAS; GUSEV, 2023). The NVAR kernel leverages delay embeddings to capture underlying temporal dynamics, offering a computationally efficient alternative to traditional RC-based models. Applied to UTS and MTS, our experiments reveal that the NVAR kernel achieves high classification accuracy for distinct, well-separated classes, such as high-arousal emotions and dynamic physical activities, demonstrating its robustness and practical applicability.

The state-of-the-art classification methods are based on complex neural networks architectures and yield top accuracies of 87% for RAVDESS, 90% for Emo-DB and 99.5% for HAR (LUNA-JIMÉNEZ et al., 2021; SADOK; LEGLAIVE; SéGUIER, 2023; SCHULLER et al., 2017). The accuracies achieved in this work with NVAR kernel SVM are 42% for RAVDESS, 53% Emo-DB and 93% for HAR. Considering that the training and inference of a neural network is computationally expensive, the NVAR kernel SVM shows an advantage in terms of time efficiency, yielding acceptable results for classification in much shorter running time.

However, the NVAR kernel shows certain limitations. It is efficient in using a heuristic approach to determine hyperparameters that may not always provide a good representation for datasets with complicate overlapping classes. This is mainly evident in tasks with subtle emotional states and static physical activities, wherein misclassification rates remain high. Moreover, the sensitivity of the kernel to the class imbalance and some high-dimensional data difficulties marked some research areas that required more work.

Future work could focus on improving the hyperparameter tuning process of the NVAR kernel, possibly incorporating adaptive delay embeddings or feature selection strategies to improve performance in more complex settings. An outstanding area for further investigation would involve advanced classifiers or alternative augmentation techniques from a perspective of balanced class distribution.

REFERENCES

ABDULLAH, M.; AHMAD, M.; HAN, D. Facial expression recognition in videos: An cnn-lstm based model for video classification. In: **2020 International Conference on Electronics, Information, and Communication (ICEIC)**. [S.l.: s.n.], 2020. p. 1–3.

ANUSHA, R. et al. Speech emotion recognition using machine learning. In: **2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)**. [S.l.: s.n.], 2021. p. 1608–1612.

BOLLT, E. On explaining the surprising success of reservoir computing forecaster of chaos? the universal machine learning dynamical system with contrast to var and dmd. **Chaos an Interdisciplinary Journal of Nonlinear Science**, v. 31, n. 1, jan. 2021. Disponível em: <https://pubs.aip.org/aip/cha/article/31/1/013108/341924/On-explaining-the-surprising-success-of-reservoir>.

BURKHARDT, F. et al. A database of German emotional speech. In: . [S.l.: s.n.], 2005. v. 5, p. 1517–1520.

CHURAEV, E.; SAVCHENKO, A. V. Touching the limits of a dataset in video-based facial expression recognition. In: **2021 International Russian Automation Conference (RusAutoCon)**. [S.l.: s.n.], 2021. p. 633–638.

FELICE, G. D.; GOULERMAS, J. Y.; GUSEV, V. Time series kernels based on nonlinear vector autoregressive delay embeddings. In: **Thirty-seventh Conference on Neural Information Processing Systems**. [s.n.], 2023. Disponível em: <https://openreview.net/forum?id=UBUWFEwn7p>.

GAUTHIER, D. J. et al. Next generation reservoir computing. **CoRR**, abs/2106.07688, 2021. Disponível em: <https://arxiv.org/abs/2106.07688>.

HAMILTON, J. D. **Time Series analysis**. [s.n.], 1994. Disponível em: <https://doi.org/10.1515/9780691218632>.

JAEGER, H. The"echo state"approach to analysing and training recurrent neural networks. In: . [s.n.], 2001. Disponível em: <https://api.semanticscholar.org/CorpusID:15467150>.

KOTTI, M.; KOTROPOULOS, C. Gender classification in two emotional speech databases. In: **2008 19th International Conference on Pattern Recognition**. [S.l.: s.n.], 2008. p. 1–4.

LIVINGSTONE, S. R.; RUSSO, F. A. The Ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in North American English. **PLOS ONE**, Public Library of Science, v. 13, n. 5, p. 1–35, 05 2018. Disponível em: <https://doi.org/10.1371/journal.pone.0196391>.

LUNA-JIMÉNEZ, C. et al. A proposal for multimodal emotion recognition using aural transformers and action units on ravdess dataset. **Applied Sciences**, 2021. Disponível em: <https://paperswithcode.com/paper/a-proposal-for-multimodal-emotion-recognition>.

MCFEE, B. et al. librosa: Audio and music signal analysis in python. In: HUFF, K.; BERGSTRA, J. (Ed.). **Proceedings of the 14th Python in Science Conference**. Austin, TX: [s.n.], 2015. p. 18–25.

MIKALSEN, K. Øyvind et al. Time series cluster kernel for learning similarities between multivariate time series with missing data. **Pattern Recognition**, v. 76, p. 569–581, 2018. ISSN 0031-3203. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0031320317304843>.

REYES-ORTIZ, J. et al. **Human Activity Recognition Using Smartphones**. 2013. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C54S4K.

SADOK, S.; LEGLAIVE, S.; SéGUIER, R. **A vector quantized masked autoencoder for speech emotion recognition**. 2023. Disponível em: <https://arxiv.org/abs/2304.11117>.

SCHULLER, B. et al. The interspeech 2017 computational paralinguistics challenge: Emotional load detection in spontaneous speech. **Journal on Multimodal User Interfaces**, v. 11, n. 4, p. 319–328, 2017.

SINITH, M. S. et al. Emotion recognition from audio signals using support vector machine. In: **2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS)**. [S.l.: s.n.], 2015. p. 139–144.

STROGATZ, S. H. **Nonlinear dynamics and chaos**. [s.n.], 2018. Disponível em: <https://doi.org/10.1201/9780429492563>.

TAKENS, F. Detecting strange attractors in turbulence. In: RAND, D.; YOUNG, L.-S. (Ed.). **Dynamical Systems and Turbulence, Warwick 1980**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1981. p. 366–381. ISBN 978-3-540-38945-3.

VIMAL, B. et al. Mfcc based audio classification using machine learning. In: **2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)**. [S.l.: s.n.], 2021. p. 1–4.

YING, S.; ZHANG, X. A study of zero-crossings with peak-amplitudes in speech emotion classification. In: **2010 First International Conference on Pervasive Computing, Signal Processing and Applications**. [S.l.: s.n.], 2010. p. 328–331.

ZEROUAL, A. et al. Deep learning methods for forecasting covid-19 time-series data: A comparative study. **Chaos Solitons & Fractals**, v. 140, p. 110121, jul. 2020. Disponível em: <https://doi.org/10.1016/j.chaos.2020.110121>.

ZHANG, Z. **Multivariate time series analysis in climate and environmental research**. [s.n.], 2017. Disponível em: <https://doi.org/10.1007/978-3-319-67340-0>.

A HOW TO RUN OUR CODE

We describe below how to run the project's pipeline in a local machine assuming Anaconda is already installed and updated to the last version.

1. Download the RAVDESS, Emo-DB, and HAR compressed files, extract them, and place their contents into their respective named sub-folders within the "feature engineering" folder.

2. Create a conda environment for installing dependencies and running the python scripts of the project, running the commands below.

   ```
   conda env create -f environment.yml
   conda activate nvark
   ```

3. Run "feature_engineering.ipynb" inside the "feature engineering" folder to generate train and test files for each dataset.

4. Copy generated files to the correct directories:

   (a) "RAVDESS_TRAIN.tsv", "RAVDESS_TEST.tsv" files to "nvark-kernel/datasets/KM/RAVDESS" folder.

   (b) "EmoDB_TRAIN.tsv", "EmoDB_TEST.tsv" to "nvark-kernel/datasets/KM/EmoDB" folder.

   (c) "HAR_TRAIN.ts", "HAR_TEST.ts" to "nvark-kernel/datasets/HAR" folder.

5. Specify the desired dataset to run experiments on the field "datasets_list" in the top of "main.py" script of "nvark-kernel" folder.

6. Run "main.py" inside of "nvark-kernel" folder.

7. To analyze results, copy the ".npy" files from "nvark-kernel/results" to "results-analysis/acc" folder.

8. Run "analysis_acc.ipynb" script inside "results-analysis/acc" folder.