RED CoMETS: An ensemble classifier for symbolically represented multivariate time series

Luca A. Bennett[1,2⋆(✉)] and Zahraa S. Abdallah[1 [0000−0002−1291−2918]]

[1] School of Engineering Mathematics and Technology
University of Bristol, Bristol, UK
zahraa.abdallah@bristol.ac.uk
[2] Awerian, Cambridge, UK
luca.bennett@awerian.net

Abstract. Multivariate time series classification is a rapidly growing research field with practical applications in finance, healthcare, engineering, and more. The complexity of classifying multivariate time series data arises from its high dimensionality, temporal dependencies, and varying lengths. This paper introduces a novel ensemble classifier called RED CoMETS (Random Enhanced Co-eye for Multivariate Time Series), which addresses these challenges. RED CoMETS builds upon the success of Co-eye, an ensemble classifier specifically designed for symbolically represented univariate time series, and extends its capabilities to handle multivariate data. The performance of RED CoMETS is evaluated on benchmark datasets from the UCR archive, where it demonstrates competitive accuracy when compared to state-of-the-art techniques in multivariate settings. Notably, it achieves the highest reported accuracy in the literature for the 'HandMovementDirection' dataset. Moreover, the proposed method significantly reduces computation time compared to Co-eye, making it an efficient and effective choice for multivariate time series classification.

Keywords: Time series classification · Multivariate time series · Co-eye · Symbolic representation · Ensemble classification

1 Introduction

Problems involving the classification of time series data play a crucial role in various domains, including the sciences, data mining, finance, and signal processing. Time series and their classifiers can be categorised into two types: univariate and multivariate. Despite multivariate time series classification problems being more prevalent in real-world scenarios, the literature has historically focused more on the univariate case [20]. Although recent studies have proposed promising methods to address multivariate time series classification [20], there still exists a gap, emphasising the need for accurate and efficient algorithms in this domain.

⋆ This author was with the University of Bristol while this research was undertaken but is currently affiliated with Awerian.

Traditional time series classifiers typically seek discriminatory features within the time series or adopt a holistic view of the entire series [2]. They often concentrate on a single representation aspect, such as shape or frequency [9]. However, time series classification problems can greatly differ in terms of training and testing sizes, dimensions, classes, series length, and class distribution. Consequently, a single approach cannot effectively handle all types of time series.

In this paper, we extend the techniques introduced by Co-eye for univariate time series classification [1], which draws inspiration from the compound eyes of insects. Co-eye utilizes two symbolic representation transformations, namely Symbolic Aggregate Approximation (SAX) [17] and Symbolic Fourier Approximation (SFA) [21], to extract discriminatory features from the time series. These transformations generate multiple "lenses" that can detect discriminatory features at various levels of granularity, capturing both fine details and broad shapes. By forming an ensemble of these lenses, Co-eye integrates different perspectives from the time and frequency domains, allowing for effective feature extraction in time series classification problems with diverse characteristics.

We propose a novel ensemble classifier for multivariate time series classification that builds upon Co-eye in two significant ways. Firstly, we enhance Co-eye's success in handling univariate problems and propose an improved approach that significantly reduces computation time without sacrificing accuracy. Secondly, we leverage this enhanced univariate approach as a foundation for a novel multivariate classifier, exploring two distinct techniques. Our proposed multivariate classifier is named RED CoMETS, which stands for Random Enhanced Co-eye for Multivariate Time Series. We evaluate RED CoMETS against state-of-the-art classifiers using datasets from the UCR archive [3], and it achieves state-of-the-art results.

The remainder of this paper is organised as follows: Section 2 discusses relevant prior research. Section 3 provides details on our optimized univariate foundation built upon Co-eye. Section 4 outlines the proposed extensions for multivariate classification. Section 5 presents the experimental results, specifically focusing on test accuracy. Finally, Section 6 concludes the paper.

2 Related Work

Co-eye leverages the Symbolic Aggregate Approximation (SAX) [17] and Symbolic Fourier Approximation (SFA) [21] techniques to construct lenses, each offering a distinct view of the time series data in both the time and frequency domains. These lenses, represented by triplets denoted as $< s, \alpha, w >$, where s indicates the choice between SAX and SFA, and α and w are the hyperparameters for alphabet size and word length, respectively, provide Co-eye with a multi-resolution perspective [1]. Through a careful "pair selection" process, Co-eye identifies the most effective set of lenses for a given classification problem. During the classification phase, Co-eye builds a Random Forest [22] for each lens using the transformed time series. These Random Forests' outputs are combined using a dynamic voting method, allowing the most confident lenses to be

matched to specific sequences and effectively extracting discriminatory features [1]. Co-eye has demonstrated competitive accuracies compared to state-of-the-art univariate classifiers when evaluated on datasets from the UCR archive [1].

The reviews by Bagnall et al. [2] and Ruiz et al. [20] provide a comprehensive overview of the strengths and weaknesses of different approaches, highlighting their performance on a range of datasets. This information is crucial in understanding the landscape of existing classifiers and identifying gaps or areas where further improvements can be made.

Dynamic Time Warping (DTW) [14] is chosen as one of the benchmark classifiers. DTW utilizes a unique distance metric in combination with the 1-nearest neighbour classifier and serves as a baseline performance measure for "good" time series classifiers. It was used as a benchmark by both Bagnall et al. [2] and Ruiz et al.[20], making it a compelling target to surpass.

Another benchmark classifier is the Multiple Representation Sequence Learner (MrSEQL) [16], which transforms time series into various symbolic representations and forms an ensemble using a SEQL classifier. While MrSEQL shares similarities with Co-eye in methodology, differences lie in the base classifier, parameterisation of symbolic representations, and voting methods [16].

ROCKET (Random Convolutional Kernel Transform) [9] is a powerful classifier that has demonstrated exceptional performance in both univariate and multivariate time series classification. ROCKET leverages random convolutional kernels to transform time series data and apply a linear classifier to make predictions. It has achieved leading accuracies across the univariate UCR archive datasets while maintaining an extremely low computation time. The effectiveness and efficiency of ROCKET make it a natural choice to benchmark against for state-of-the-art performance.

HIVE-COTE (Hierarchical Vote Collective of Transformation-based Ensembles) [18] is a heterogeneous ensemble classifier that combines multiple transformation based models. Its latest edition, HIVE-COTE 2.0 [19], is currently the best-ranked multivariate time series classifier in terms of accuracy. HIVE-COTE constructs an ensemble of diverse classifiers, including shapelet-based, interval-based, and dictionary-based classifiers, and employs a hierarchical voting strategy to make predictions. The hierarchical nature of HIVE-COTE allows it to capture different levels of temporal patterns and achieve robust performance on a wide range of time series datasets. As the leading multivariate time series classifier, HIVE-COTE serves as the "method to beat" for RED CoMETS.

In the realm of deep learning-based approaches for multivariate time series classification, InceptionTime [12] stands out. It is an ensemble of convolutional neural networks specifically designed for time series classification. InceptionTime introduces the concept of inception modules, which consist of parallel convolutional layers with different filter sizes. This design allows the network to capture diverse temporal patterns at multiple resolutions. InceptionTime has been identified by Ruiz et al. [20] as the leading deep learning-based approach for both univariate and multivariate time series classification. Their review demonstrated that InceptionTime achieved top-performing accuracy across various datasets

and outperformed many traditional and state-of-the-art classifiers. Therefore, it serves as a strong baseline for comparing the performance of RED CoMETS against deep learning-based approaches.

In addition to InceptionTime, deep learning architectures such as Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNNs) have gained popularity in time series classification. LSTM networks, a type of recurrent neural network (RNN), are capable of capturing long-term dependencies in sequential data and have shown promising results for classifying both univariate and multivariate time series [13].

CNNs, on the other hand, are primarily known for their success in computer vision tasks, but they have also been applied to time series classification with remarkable outcomes. In the context of time series, 1D CNNs are often employed to learn hierarchical representations of input sequences by convolving filters across different time steps. This allows them to automatically extract relevant local patterns and capture higher-level representations of the data [4,24].

Deep learning-based approaches offer the advantage of automatically learning relevant features from raw time series data, obviating the need for handcrafted feature engineering. However, they often require large amounts of training data and significant computational resources for model training and optimization. Additionally, the interpretability of deep learning models can be challenging due to their black-box nature.

3 Univariate Foundation

As described in Section 1, we first build on the univariate classification techniques introduced by Co-eye to create a new univariate classifier as a foundation for our multivariate extensions. We adapt the learning process of Co-eye, but introduce a new pair selection method and propose three replacement voting mechanisms.

3.1 Pair Selection

Co-eye adopts a meticulous process for selecting lenses, involving two grid searches over the $\alpha - w$ parameter space for SAX and SFA, respectively. To construct an effective ensemble, each $< s, \alpha, w >$ triplet undergoes cross-validation, and pairs within a 1% margin of the highest cross-validation accuracy are chosen. However, performing an exhaustive search and cross-validation for every $< \alpha, w >$ pair can be computationally demanding, as highlighted by Abdallah and Gaber [1]. To address this bottleneck, we adopt a different approach inspired by the work of Bergstra and Bengio [7]. They suggest that random searches can yield comparable performance to grid searches for hyperparameter selection. Therefore, we incorporate random selection in our methodology.

In Co-eye, the number of pairs is not predetermined. When generating pairs randomly, it is essential to preselect the number of SAX and SFA pairs. To ensure a balanced perspective of the time series and avoid voting bias, we opt for an equal number of SAX and SFA pairs. The selection of pairs is proportional to

the length of the time series, with $\lfloor p*l \rfloor$ pairs independently chosen for SAX and SFA. Here, $0 < p \leq 1$ represents the proportion of pairs, and l denotes the length of the time series. To determine the parameter space for random selection, we draw pairs uniformly from the $\alpha - w$ space defined by Abdallah and Gaber [1]. We evaluate four different values of p, namely 0.05, 0.1, 0.15, and 0.2, denoted as R5%, R10%, R15%, and R20%, respectively. These values enable us to explore the impact of different proportions of pairs on the ensemble construction process.

By adopting this approach, we aim to strike a balance between computational efficiency and lens selection effectiveness, ensuring that Co-eye can efficiently construct an ensemble of lenses while capturing diverse perspectives of the data.

3.2 Voting

To enhance accuracy and robustness, we propose three voting methods to replace Co-eye's existing dynamic voting approach. Let's consider Co-eye applied to a dataset with n classes $c_1, \ldots, c_n$ and m samples. Each base Random Forest classifier generates an $m \times n$ matrix, denoted as:

$$M_i = \begin{array}{c} \\ \text{Sample } 1 \\ \vdots \\ \text{Sample } m \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & \ldots & c_n \end{array} \\ \begin{pmatrix} P(c=c_1) & \ldots & P(c=c_n) \\ \vdots & \ddots & \vdots \\ P(c=c_1) & \ldots & P(c=c_n) \end{pmatrix} \end{array}. \tag{1}$$

Therefore, Co-eye produces a set of matrices, denoted as $S_M = M_1, \ldots, M_k$, where k represents the number of classifiers in the ensemble. Voting can be seen as a function on S_M, resulting in a vector of class labels for the m samples. We introduce three new voting methods based on the sum rule (SR) scheme outlined in Algorithm 1, employing different weight generation functions.

Algorithm 1 Sum Rule Scheme

```
procedure SumRule(S_M)
    w ← getWeights()
    weightedMats ← w * S_M                ▷ Element-wise multiplication.
    sum ← Σ_k weightedMats                ▷ Element-wise addition.
    for row in sum do
        label ← max(row)
    end for
    return labels
end procedure
```

The first voting method is the simplest, employing uniform weights of one across the ensemble. Although efficient, we hypothesize that a more sophisticated weighting scheme could yield better results. Intuitively, matrices with higher confidence in their predictions should carry more weight. Thus, matrices with

greater row-wise maximum confidences can be considered to be more confident. For a matrix $M_i \in S_M$ with m rows, the set of row-wise maxima can be defined as $R_i^{max} = \text{row}max(j) \mid \forall j \in [m]$, where row$max(j)$ represents the maximum value of row j in matrix M_i, and $[m] = 1, \ldots, m$. Let $\overline{R_i^{max}}$ denote the mean of the row-wise maxima. Our second voting scheme then assigns weights as $\boldsymbol{w} = [\overline{R_1^{max}}, \ldots, \overline{R_k^{max}}]$.

Instead of using S_M directly for weight generation, Large et al. [15] demonstrated the effectiveness of weights determined through cross-validation. Hence, our third proposed voting method is as follows: A Random Forest is built for each $< s, \alpha, w >$ triplet, and accuracy is calculated using 5-fold cross-validation, a value supported by Burman [8]. The cross-validation accuracies are then used as weights for their respective matrices. Note that this method is significantly more computationally expensive than the other two approaches. However, unlike Co-eye's pair selection process, cross-validation is applied only to selected triplets rather than the entire $\alpha - w$ parameter space, making it computationally viable.

We refer to the three voting methods as SR Uniform, SR Mean-Max, and SR Validation, respectively.

4 Developing RED CoMETS

We anticipate that extending the multi-resolution perspectives of Co-eye, which is effective for univariate time series classification using the time and frequency domains, will be equally successful for multivariate datasets. In the literature, both forests [23] and symbolic representations [5] have achieved favourable results in this regard. To enable univariate classifiers to handle multivariate time series, we present two approaches. When combined with the univariate foundation established from Co-eye in Section 3, these approaches form RED CoMETS (Random Enhanced Co-eye for Multivariate Time Series).

4.1 Concatenating Approach

One intuitive approach to address multivariate time series classification is to reduce it to the more extensively studied univariate case. This can be achieved by sequentially concatenating the dimensions of a multivariate dataset. For a multivariate time series with a length of n and d dimensions, this method generates a univariate time series of length nd. Algorithm 2 demonstrates the application of this method to our univariate foundation. When utilizing the random pair selection technique described in Section 3.1, the number of lenses is proportional to the length of the time series. However, for computational efficiency, it was decided that if random pair selection is used, the number of lenses will be determined based on the length of the time series before concatenation, i.e., proportional to n rather than nd.

Algorithm 2 Concatenating Approach

```
1: procedure ConcatenatingApproach(TS)          ▷ TS is a multivariate dataset
2:     for dimension ∈ TS do
3:         append(concatTS, dimension)
4:     end for
5:     return UnivariateFoundation(concatTS)
6: end procedure
```

4.2 Ensembling Approach

Another approach to handling multivariate datasets is to construct an ensemble over the dimensions. This method, recommended by Ruiz et al. [20], involves building a univariate classifier for each dimension and combining their predictions for the overall classification.

Since our univariate foundation is an ensemble classifier, this leads to an ensemble of ensembles. Consequently, there are two sub-approaches depending on how the ensemble results are combined. Algorithms 3 and 4 outline these sub-approaches. Approach 1 combines the set of matrices, S_M, produced by each base classifier into a single superset $S_{\text{all}} = S_{M1} \cup S_{M2} \cup \ldots S_{Md}$, where S_{Mi} represents the set of matrices returned for the ith dimension. Voting is then applied as usual to S_{all}. Approach 2 performs voting in two stages. For each dimension, S_{Mi} is fused into a single matrix, F_i, using one of the sum rule methods outlined in Section 3.2. For the ith dimension, $F_i = \sum_k \boldsymbol{w} S_{Mi}$, where $\boldsymbol{w}$ is a vector of weights. Subsequently, a second round of voting is applied to the set of fused matrices across all dimensions, denoted as $S_F = \{F_i \mid \forall i \in [d]\}$, where $[d] = 1, \ldots, d$, resulting in the final classification. Different voting methods can be employed for the fusion and final classification stages.

Algorithm 3 Ensembling Approach 1

```
1: procedure EnsemblingApproach1(TS)            ▷ TS is a multivariate dataset
2:     for dimension ∈ TS do
3:         S_M ← UnivariateFoundation(dimension)
4:         append(S_all, S_M)
5:     end for
6:     return vote(S_all)
7: end procedure
```

4.3 RED CoMETS

The univariate foundation described in Section 3, which builds upon the innovative time series classification approach introduced by Co-eye [1], incorporates a new random pair selection process and three new voting methods. By combining

Algorithm 4 Ensembling Approach 2

```
procedure EnsemblingApproach2(TS)                ▷ TS is a multivariate dataset
    for dimension ∈ TS do
        S_M ← UnivariateFoundation(dimension)
        F_i ← Σ_k w * S_M                        ▷ Element-wise operations
        append(S_F, F_i)
    end for
    return vote(S_F)
end procedure
```

the two proposed multivariate extensions from Sections 4.1 and 4.2 with our univariate foundation, we establish a novel multivariate classifier (RED CoMETS).

5 Experiments and Evaluation

We evaluate our univariate foundation and RED CoMETS on univariate and multivariate datasets respectively from the UCR archive. We demonstrate that our univariate foundation is more accurate and approximately 40 times faster than Co-eye. RED CoMETS is shown to achieve accuracies comparable to the state-of-the-art classifiers outlined in Section 2. Our code and full results are available on GitHub [3].

5.1 Experimental Design

All of our experiments were conducted with the 111 datasets from the UCR archive [3] used by Bagnall et al. [2] and Ruiz et al. [20] in their reviews, consisting of 85 univariate and 26 multivariate datasets. This allows for comparison to the results recorded by Bagnall et al. [2] and Ruiz et al. [20] in their reviews of state-of-the-art classifiers. For consistency and to allow direct comparison, our results show the average over 30 trials on each data using 30 stratified resamples. Each resample is seeded by its sample number, such that each classifier is evaluated on identical samples and results are reproducible. Note that both SAX and SFA z-normalise the time series as their initial step. For the multivariate datasets, this means that the concatenating approach normalises the joint time series while the ensembling approach normalises each dimension independently.

We produce results for Co-eye, our univariate foundation, and RED CoMETS. Results for DTW and univariate ROCKET were taken from Bagnall et al. [2] and Dempster et al. [9] respectively. The results for DTW$_D$, MrSEQL, InceptionTime, and multivariate ROCKET were taken from Ruiz et al. [20]. The results for HIVE COTE-2.0 were taken from the author's website [3]. The default accuracy for predicting the majority class is also included and is taken from Bagnall

[3] https://github.com/zy18811/RED-CoMETS

et al. [2] and Ruiz et al. [20] for the univariate and multivariate datasets respectively. The voting methods proposed in Section 3.2 were evaluated using the R5% pair selection described in Section 3.1 to minimise computation time.

To compare multiple classifiers over multiple datasets, critical difference (CD) diagrams are used [10]. Current literature [6] suggests abandoning the post hoc test originally suggested by Demšar [10], instead forming cliques using pairwise tests, with the Holm correction being made in the case of multiple testing. The classifiers are first ranked using the Friedman test, then grouped into cliques using pairwise Wilcoxon signed rank tests with the Holm adjustment [2,20]. Cliques represent groups of classifiers between which there is no statistically significant pairwise difference. A Python implementation produced by Fawaz et al. [11] was used to create the CD diagrams presented in this paper.

5.2 Univariate Foundation

Pair Selection The four random pair selection methods outlined in Section 3.1 were evaluated on the 85 univariate datasets from the UCR archive in order to evaluate their effectiveness against Co-eye. Default accuracy, DTW, and univariate ROCKET are included as benchmarks. Figure 1 shows the test accuracy critical difference (CD) diagram for the pair selection methods. It can be seen that there are two distinct cliques containing R10%, R15%, and R20% and Co-eye and R5% respectively, with DTW found in both. Both cliques outperformed default accuracy with statistical significance. ROCKET significantly outperformed all others. R10%, R15%, and R20% all performed worse in terms of accuracy than Co-eye, and are removed from contention. There is no statistically significant pairwise difference in test accuracy between R5% and Co-eye.

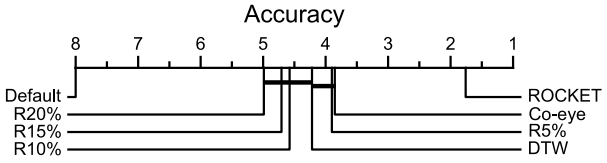

Fig. 1. Test accuracy critical difference diagram for random pair selection methods against Co-eye averaged over 30 resamples for each of the 85 univariate UCR datasets. Default accuracy, DTW, and ROCKET are included as benchmarks.

Figure 2 shows a pairwise comparison of mean train and test time between Co-eye and R5% on the 85 univariate UCR datasets. It can be seen that R5% is significantly faster than Co-eye in all cases, averaging approximately 40 times faster over the 85 datasets. As such, R5% is a pronounced improvement over Co-eye: 40 times faster with no statistically significant difference in test accuracy. For R5%, Kendall's τ coefficient was calculated between characteristics of each dataset and the associated total train and test time, with values of 0.41,

0.42, 0.33, and 0.78 for train size, test size, number of classes, and series length respectively. As one would expect, there is a positive correlation for all values, with series length as the most significant determinant of train and test time.

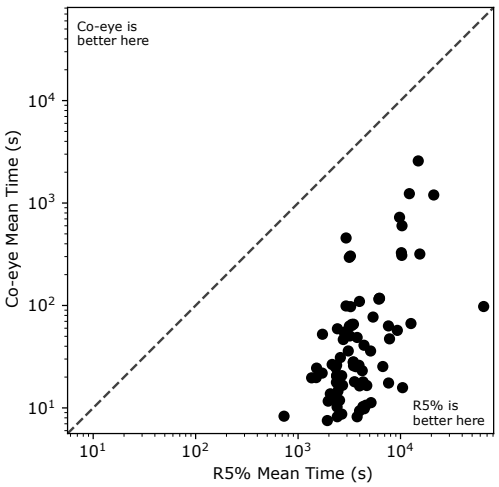

Fig. 2. Pairwise comparison of total mean train and test time between Co-eye and R5% averaged over 30 stratified resamples of the 85 univariate UCR datasets.

Voting Section 3.2 proposed three voting methods, aiming to outperform the dynamic voting method used by Co-eye in terms of test accuracy. As done above for pair selection, the voting methods were evaluated on the 85 univariate datasets from the UCR archive with default accuracy, DTW, and univariate ROCKET as benchmarks. Figure 3 shows the test accuracy CD diagram for the voting methods. It can be seen that the three proposed voting methods all performed better than Co-eye's dynamic voting method with statistical significance. The three voting methods are cliqued, indicating no significant pairwise difference between them. As such, all three voting methods are taken forward for evaluation as part of RED CoMETS.

5.3 RED CoMETS

There are nine variants of RED CoMETS, which result from different combinations of a voting method and the multivariate extension. These variants are referred to by the names presented in Table 1. It is worth noting that the validation voting method is not utilised with the ensembling dimensions multivariate extension due to initial experiments demonstrating computational infeasibility.

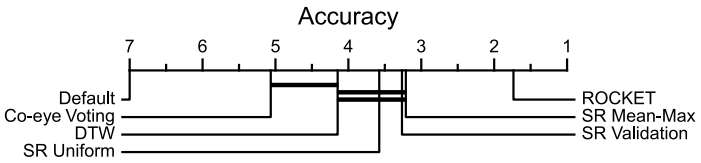

Fig. 3. Test accuracy critical difference diagram for proposed voting methods against Co-eye averaged over 30 resamples for each of the 85 univariate UCR datasets. Default accuracy, DTW, and ROCKET are included as benchmarks.

Table 1. RED CoMETS variants.

Name	Approach	Sub-Approach	Voting Method 1	Voting Method 2
RED CoMETS-1	Concatenating	n/a	Uniform	n/a
RED CoMETS-2	Concatenating	n/a	Mean-Max	n/a
RED CoMETS-3	Concatenating	n/a	Validation	n/a
RED CoMETS-4	Ensembling	1	Uniform	n/a
RED CoMETS-5	Ensembling	1	Mean-Max	n/a
RED CoMETS-6	Ensembling	2	Uniform	Uniform
RED CoMETS-7	Ensembling	2	Uniform	Mean-Max
RED CoMETS-8	Ensembling	2	Mean-Max	Mean-Max
RED CoMETS-9	Ensembling	2	Mean-Max	Uniform

We evaluate the RED CoMETS variants against each other and the multivariate benchmarks discussed in Section 2 (DTW_D, MrSEQL, InceptionTime, ROCKET, and HIVE-COTE 2.0). When evaluated by Ruiz et al. [20], InceptionTime and MrSEQL were unable to complete all 26 datasets, with InceptionTime failing on 'EigenWorms' due to memory errors and MrSEQL failing to complete 'FaceDetection' and 'PhonemeSpectra' within the set time constraints. Likewise, all variants of RED CoMETS were unable to complete 'Eigenworms'. As such, results for these datasets will not be used in our comparison, leaving 23 datasets for evaluation. Based on the results shown in Section 5.2, all variants of RED CoMETS are evaluated using the R5% pair selection method. As Section 5.2 demonstrated no statistically significant difference between the three proposed voting methods, all nine RED CoMETS variants shown in Table 1 are evaluated.

We first analyse the nine variants of RED CoMETS. It can be seen in Figure 4 that there is no statistically significant pairwise difference in test accuracy between the nine RED CoMETS variants, with the default accuracy being outperformed with statistical significance in all cases. However, looking at the results shown in Table 2, RED CoMETS-3 has both the highest mean accuracy and number of wins, indicating that it is both the most accurate and most reliable of the nine RED CoMETS variants.

Having identified RED CoMETS-3 as the most effective variant, we now evaluate it against the state-of-the-art methods identified in Section 2. It can be seen from Figure 5 that, excluding default accuracy, RED CoMETS-3 has the lowest ranking in terms of test accuracy. However, the cliques indicate that there

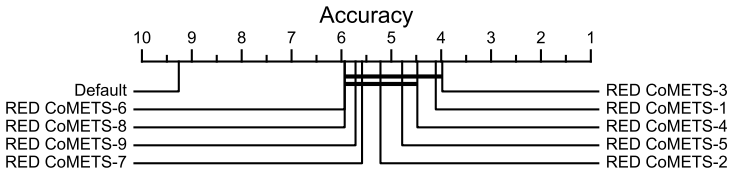

Fig. 4. Test accuracy critical difference diagram for RED CoMETS variants averaged over the 23 UCR datasets.

Table 2. Summary of RED CoMETS results showing mean accuracy across 30 resamples for each variant and multivariate dataset. The mean and number of wins are also shown. The greatest values on each row are shown in underlined bold.

	RED CoMETS-<N> (%)								
Dataset	1	2	3	4	5	6	7	8	9
AWR	**97.73**	97.60	**97.73**	96.22	96.02	95.20	95.16	94.91	94.18
AF	30.00	29.11	29.78	28.89	28.89	**32.00**	**32.00**	31.33	31.33
BM	**98.17**	98.00	**98.17**	79.58	79.42	81.67	81.92	81.75	82.08
CR	97.08	**97.13**	**97.13**	92.59	92.73	89.31	89.31	89.58	89.68
DDG	59.60	54.47	**62.27**	20.53	19.13	20.47	20.53	19.60	19.40
EP	85.14	83.60	**85.29**	60.31	53.26	58.36	57.83	51.79	50.46
ER	**93.54**	92.38	93.51	91.23	91.19	85.68	85.63	85.88	85.79
EC	27.55	27.60	27.59	33.13	**33.36**	32.60	32.56	32.69	32.53
FM	51.93	50.30	51.60	52.20	**52.53**	52.10	52.10	52.40	52.43
HMD	54.20	54.57	**55.30**	44.36	44.68	42.40	42.40	42.81	42.99
HW	**32.73**	31.67	32.60	28.97	29.05	27.33	27.32	27.64	27.68
HB	66.44	65.38	66.50	71.02	70.98	71.02	**71.12**	70.98	71.04
LIB	**78.33**	75.93	**78.33**	73.33	72.89	58.85	58.85	57.85	57.85
LSST	15.96	05.76	**50.93**	08.90	08.07	05.35	05.21	04.03	03.71
MI	51.00	51.20	50.97	51.33	51.50	51.37	51.40	**51.57**	51.53
NATO	82.04	81.81	**82.30**	73.54	73.78	72.41	72.83	72.15	72.72
PEMS	78.30	77.59	78.30	90.98	91.89	92.08	92.49	93.14	**93.66**
PD	88.00	82.16	**88.17**	76.32	76.24	63.64	63.64	64.21	64.21
RS	**83.05**	72.74	82.87	78.60	78.82	75.46	75.70	75.61	75.77
SRS1	85.46	85.51	85.46	86.47	**86.50**	86.35	86.36	86.38	86.41
SRS2	51.89	52.02	52.00	**52.39**	52.35	52.37	52.37	52.35	52.33
SWJ	38.89	38.44	38.44	43.33	43.33	44.67	**44.89**	44.22	44.44
UW	**88.61**	88.53	88.60	84.20	84.14	81.09	80.99	80.95	80.83
Mean	66.77	64.94	**68.43**	61.67	61.34	59.64	59.68	59.30	59.26
Wins	5.5	0.5	**8**	1	3	0.5	2.5	1	1

is no statistically significant difference in accuracy between RED CoMETS-3 and DTW_D, MrSEQL, and InceptionTime, demonstrating that RED CoMETS-3 is competitive with state-of-the-art multivariate classifiers.

We now further analyse the performance of RED CoMETS-3 in relation to the benchmarks, with Table 3 showing the differences in test accuracy. RED CoMETS-3 was able to beat all of the benchmarks on at least four of the datasets.

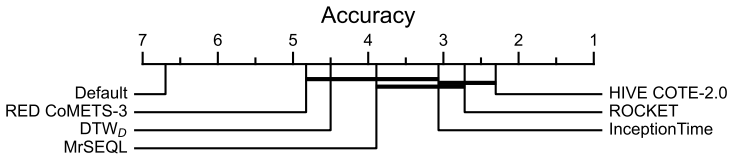

Fig. 5. Test accuracy critical difference diagram for RED CoMETS-3 against the state-of-the-art classifiers averaged over the 23 UCR datasets

Both the mean and median difference in accuracy between RED CoMETS-3 and DTW$_D$, MrSEQL, and InceptionTime is less than 5%, concurring with Figure 5. Looking at the maxima and minima, it can be seen that RED CoMETS-3 greatly outperforms the benchmarks on some datasets and vice versa. In fact, RED CoMETS-3 consistently outperforms the state-of-the-art benchmarks on a small number of datasets, beating all of the benchmarks on HMD, four on AF and DDG, and three on ER, SRS1, and SRS2. In other words, just six datasets account for 22 out of the 28 wins shown in Table 3. Five of these six datasets are categorised as EEG, ECG, or spectrographic. Hence, it is apparent that RED CoMETS attains its best performance on datasets with minimal phase shifting (this was also found to be the case for Co-eye by Abdallah and Gaber [1]).

Table 3. Summary of the test accuracy differences between RED CoMETS-3 and the benchmarks for the multivariate UCR datasets. Negative is better for RED CoMETS-3.

Classifier	Mean (%)	Median (%)	Max (%)	Min (%)	STD (%)	Wins	Losses
DTW$_D$	0.68	1.69	28.60	-24.98	10.32	9	14
MrSEQL	4.39	3.93	65.60	-33.33	18.31	6	17
InceptionTime	3.64	2.63	64.51	-65.59	21.98	5	18
ROCKET	5.09	5.25	24.06	-16.13	8.66	4	19
HIVE COTE-2.0	7.50	5.33	51.50	-15.52	13.17	4	19

HIVE COTE-2.0 and ROCKET are considered the current best within the state-of-the-art as discussed in Section 2. Figure 5 corroborates this, with them being ranked first and second respectively. We now compare RED CoMETS-3 against them in more detail, seeking to better understand the disparities shown in Table 3. It can be seen from Table 4 that HIVE-COTE 2.0 retains its place as the current best classifier in terms of test accuracy with both the greatest mean accuracy and number of wins. However, RED CoMETS-3 is still able to hold its own against ROCKET and HIVE-COTE 2.0, beating both of them on four of the datasets. Furthermore, the result obtained for the HMD dataset, 55.30%, is greater than any reported in the literature [3], representing a notable improvement to the state-of-the-art.

Table 4. Results for ROCKET, HIVE-COTE 2.0, and RED CoMETS-3 showing mean test accuracy across 30 resamples of each multivariate dataset. The mean and number of wins are also shown. The greatest values on each row are shown in underlined bold.

Dataset	ROCKET (%)	HIVE COTE-2.0 (%)	RED CoMETS-3 (%)
AWR	99.56	**99.58**	97.73
AF	24.89	28.22	**29.78**
BM	**99.00**	98.92	98.17
CR	**100.00**	99.95	97.13
DDG	46.13	49.87	**62.27**
EP	99.08	**99.83**	85.29
ER	98.05	**98.51**	93.51
EC	44.68	**79.09**	27.59
FM	**55.27**	55.23	51.60
HMD	44.59	39.77	**55.30**
HW	**56.67**	56.34	32.60
HB	71.76	**72.86**	66.50
LIB	90.61	**92.69**	78.33
LSST	63.15	**63.70**	50.93
MI	53.13	**53.17**	50.97
NATO	88.54	**89.20**	82.30
PEMS	**99.56**	**99.56**	88.17
PD	85.63	**99.81**	78.30
RS	92.79	**93.05**	82.87
SRS1	86.55	**87.87**	85.46
SRS2	51.35	50.46	**52.00**
SWJ	**45.56**	43.78	38.44
UW	94.43	**94.89**	88.60
Mean	73.52	**75.93**	68.43
Wins	5.5	**13.5**	4

6 Conclusion

RED CoMETS is a novel ensemble classifier for multivariate time series that builds on the success of Co-eye. In order to build a univariate foundation for our classifier, we adapted Co-eye's use of multiple symbolic representations to gain a multi-resolution perspective of both the time and frequency domains. However, we introduced a random pair selection process in order to overcome the bottleneck in Co-eye [1]. We also proposed and evaluated three new voting methods. Our adaption of Co-eye was extremely successful, achieving an approximately 40 times increase in speed and small but statistically significant gains in accuracy in comparison to Co-eye.

Two multivariate extensions were then applied to our univariate classifier. The different possible combinations of the multivariate extensions and voting methods resulted in the nine variants of RED CoMETS shown in Table 1. These were evaluated against state-of-the-art classifiers on 23 multivariate datasets from the UCR archive [3], following the methodology of Ruiz et al. [20].

RED CoMETS-3 was identified as the clear best out of the nine variants in both accuracy and reliability and was demonstrated to have no statistically significant pairwise difference in accuracy to several of the state-of-the-art benchmarks. RED CoMETS-3 was able to outperform both ROCKET and HIVE COTE-2.0, the current best-in-class, on four of the 23 datasets and achieved an accuracy greater than reported by any classifier in the literature on the 'HandMovementDirection' dataset. It was noted that RED CoMETS attains its best performance on datasets with no significant phase shifting.

There is room to further improve RED CoMETS-3 in both the R5% pair selection and SR Validation voting method. For R5%, a subset of the datasets could be used to learn the optimal bounds for the $\alpha - w$ parameter space, similar to the methodology used by Dempster et al. [9] when learning the kernel parameter space for ROCKET. SR Validation could be improved by emulating the scheme proposed by Large et al. [15] in which the weights are raised to a power in order to amplify differences between base classifiers.

References

1. Abdallah, Z.S., Gaber, M.M.: Co-eye: a multi-resolution ensemble classifier for symbolically approximated time series. Machine Learning **109**(11), 2029–2061 (11 2020). https://doi.org/10.1007/s10994-020-05887-3
2. Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E.: The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery **31**(3), 606–660 (5 2017). https://doi.org/10.1007/s10618-016-0483-9
3. Bagnall, A., Keogh, E., Lines, J., Bostrom, A., Large, J., Middlehurst, M.: UEA & UCR Time Series Classification Repository, www.timeseriesclassification.com
4. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271 (2018)
5. Baydogan, M., Runger, G.: Learning a symbolic representation for multivariate time series classification. Data Mining and Knowledge Discovery **29**, 1–23 (03 2014). https://doi.org/10.1007/s10618-014-0349-y
6. Benavoli, A., Corani, G., Mangili, F.: Should we really use post-hoc tests based on mean-ranks? Journal of Machine Learning Research **17**(5), 1–10 (2016), http://jmlr.org/papers/v17/benavoli16a.html
7. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. Journal of Machine Learning Research (2012)
8. Burman, P.: A Comparative Study of Ordinary Cross-Validation, v-Fold Cross-Validation and the Repeated Learning-Testing Methods. Biometrika **76**(3), 503 (9 1989). https://doi.org/10.2307/2336116
9. Dempster, A., Petitjean, F., Webb, G.I.: ROCKET: exceptionally fast and accurate time series classification using random convolutional kernels. Data Mining and Knowledge Discovery **34**(5), 1454–1495 (9 2020). https://doi.org/10.1007/s10618-020-00701-z
10. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research **7**(1), 1–30 (2006), http://jmlr.org/papers/v7/demsar06a.html

11. Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A.: Deep learning for time series classification: a review. Data Mining and Knowledge Discovery **33**(4), 917–963 (7 2019). https://doi.org/10.1007/s10618-019-00619-1
12. Ismail Fawaz, H., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D.F., Weber, J., Webb, G.I., Idoumghar, L., Muller, P.A., Petitjean, F.: InceptionTime: Finding AlexNet for time series classification. Data Mining and Knowledge Discovery **34**(6), 1936–1962 (11 2020). https://doi.org/10.1007/s10618-020-00710-y
13. Karim, F., Majumdar, S., Darabi, H., Harford, S.: Multivariate lstm-fcns for time series classification. Neural networks **116**, 237–245 (2019)
14. Keogh, E.J., Pazzani, M.J.: Scaling up dynamic time warping for datamining applications. In: Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '00. pp. 285–289. ACM Press, New York, New York, USA (2000). https://doi.org/10.1145/347090.347153
15. Large, J., Lines, J., Bagnall, A.: A probabilistic classifier ensemble weighting scheme based on cross-validated accuracy estimates. Data Mining and Knowledge Discovery **33**(6), 1674–1709 (11 2019). https://doi.org/10.1007/s10618-019-00638-y
16. Le Nguyen, T., Gsponer, S., Ilie, I., O'Reilly, M., Ifrim, G.: Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations. Data Mining and Knowledge Discovery **33**(4), 1183–1222 (7 2019). https://doi.org/10.1007/s10618-019-00633-3
17. Lin, J., Keogh, E., Wei, L., Lonardi, S.: Experiencing SAX: A novel symbolic representation of time series. Data Mining and Knowledge Discovery **15**(2), 107–144 (10 2007). https://doi.org/10.1007/s10618-007-0064-z
18. Lines, J., Taylor, S., Bagnall, A.: Hive-cote: The hierarchical vote collective of transformation-based ensembles for time series classification. In: 2016 IEEE 16th International Conference on Data Mining (ICDM). pp. 1041–1046 (2016). https://doi.org/10.1109/ICDM.2016.0133
19. Middlehurst, M., Large, J., Flynn, M., Lines, J., Bostrom, A., Bagnall, A.: Hive-cote 2.0: a new meta ensemble for time series classification. Machine Learning **110**(11), 3211–3243 (Dec 2021). https://doi.org/10.1007/s10994-021-06057-9, https://doi.org/10.1007/s10994-021-06057-9
20. Ruiz, A.P., Flynn, M., Large, J., Middlehurst, M., Bagnall, A.: The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery **35**(2), 401–449 (3 2021). https://doi.org/10.1007/s10618-020-00727-3
21. Schäfer, P., Högqvist, M.: SFA: A symbolic fourier approximation and index for similarity search in high dimensional datasets. In: Proceedings of the 15th International Conference on Extending Database Technology - EDBT '12. p. 516. ACM Press, New York, New York, USA (2012). https://doi.org/10.1145/2247596.2247656
22. Tin Kam Ho: Random decision forests. In: Proceedings of 3rd International Conference on Document Analysis and Recognition. pp. 278–282. IEEE Comput. Soc. Press (1995). https://doi.org/10.1109/ICDAR.1995.598994
23. Tuncel, K., Baydogan, M.: Autoregressive forests for multivariate time series modeling. Pattern Recognition **73** (08 2017). https://doi.org/10.1016/j.patcog.2017.08.016
24. 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). pp. 1578–1585. IEEE (2017)