DATA SCIENCE JOURNAL

Wang, H, et al. 2018 Shapelet Classification Algorithm Based on Efficient Subsequence Matching. *Data Science Journal,* 17: 6, pp. 1–12, DOI: https://doi.org/10.5334/dsj-2018-006

RESEARCH PAPER

Shapelet Classification Algorithm Based on Efficient Subsequence Matching

Huiqing Wang, Chun Li, Hongwei Sun, Zhirong Guo and Yingying Bai

Taiyuan University of Technology, CN

Corresponding author: Chun Li (lichun0364@link.tyut.edu.cn)

Shapelet classification algorithms are an accurate classification method for time series data. Existing shapelet classifying processes are relatively inefficient and slow due to the large amount of necessary complex distance computations. This paper therefore introduces piecewise aggregate approximation(PAA) representation and an efficient subsequence matching algorithm for shapelet classification algorithms; the paper also proposes shapelet transformation classification algorithm based on efficient series matching. First, the proposed algorithm took the PAA representation for appropriate dimension reduction, and then used a subsequence matching algorithm to simplify the data classification process. The research experimented on 14 public time series datasets taken from UCI and UCR, used the original and new algorithm for classification, and compared the efficiency and accuracy of the two methods. Experimental results showed that the efficient subsequence matching algorithm could be combined with the shapelet classification algorithm; the new algorithm could ensure relatively high classification accuracy, effectively simplified the algorithm calculation process, and improved classification efficiency.

Keywords: shapelets; shapelets transformation; time series classification; subsequence to subsequence matching; PAA

1. Introduction

As a type of high-dimensional massive data, time series are common in fields such as meteorology, finance, geology, medicine, electronic information, and network security. They are also a major research subject in data mining (Esling and Agon 2012). Time series research includes similarity searching (Rakthanmanon et al. 2012), clustering (Aghabozorgi and Wah 2014), classification (Petitjean et al. 2015), pattern recognition (Begum and Keogh 2014), and prediction (Aljumeily and Hussain 2015). Among these, time series classification (TSC) has become a hot topic because of its fundamentality. Time series classification obtains identification features that can distinguish between different time series by learning from training sets with known class tags, and then automatically assign class tags to untagged time series.

Initially, the research staff used the nearest neighbor algorithm to process time series classifications (Ding et al. 2008; Batista et al. 2011; Deng et al. 2013; Alonso et al. 2005; Jeong et al. 2011; Buza 2011). Despite the fact that the nearest neighbor algorithm was simple and involved fewer parameters, new research suggested that it needed to search and store the entire dataset during the time series classification process, which resulted in relatively high time and space complexity. Researchers hoped to achieve high classification accuracy and derive implicit messages from the experiment; this could not be achieved with the nearest neighbor algorithm. Additionally, these methods often resulted in unsatisfactory results because some time series were very similar, and the resulting noise could obscure the subtle differences between similar time series. Therefore, the above algorithm was not effective at classifying time series that had subtle differences.

Researchers have been working to solve the above problem with a new classification algorithm that better solves time series classification problems. Ye, Keogh (2009), and other researchers first introduced shapelet algorithms to classify time series that only had minor partial differences. Shapelet algorithms use partial time series fragments for classification, which reduce noise and lead to better accuracy and robustness.

Shapelet classification could also produce results with higher explanatory power, which could clearly show class differences and help researchers better understand data. Since then, shapelet classification algorithms have been widely used in various fields involving time-series studies (Hartmann 2010; Xing et al. 2011; Shajina et al. 2012). Compared with the existing classification, shapelet time series classification algorithms were more accurate, but the shapelet extraction process was slow, which made it prohibitive for very large datasets. Therefore, shapelet classification research has mostly focused on accelerating the extraction process. Ye and Keogh (2011), Mueen (2011), He (2012), Rakthanmanon (2013), and other researchers proposed improved algorithms that expedited the process. Lines and Bagnall (2012) comprehensively analyzed the pros and cons of several quality metrics during the extraction process. However, these improvements did not fundamentally address the problem of how to best use shapelet classification algorithms to solve time series classification. Bagnall (2013) and other researchers demonstrated the importance of using an integrated approach to isolate data transformation from the classification algorithm. Lines, Davis (2012), and other researchers proposed the concept of shapelet transformation, and broke the restriction requiring shapelet classification to use decision trees. They utilized the distance of the original time series from the shapelets to convert data and create a new dataset, and then used the generic classifier for classification.

This article introduces PAA time series representation and an efficient subsequence matching method in the shapelet classification algorithm, and proposes an improved shapelet conversion classification algorithm. The proposed algorithm preprocesses the original data with a PAA time series representation to reduce data dimensions, and then uses highly efficient subsequence matching methods to simplify the subsequence distance calculation during the extraction and conversion processes of the shapelet classification algorithm to reduce computing complexity and improve efficiency. We made the following contributions: (1) We proposed a shapelet conversion classification algorithm based on highly efficient subsequence matching; (2) We studied the impact of PAA representation to process the original time series on shapelet classification; (3) We carried out experiments on real datasets and validated that the proposed method is feasible and efficient; (4) We analyzed the results using a variety of common classifiers to convert shapelet classification data.

This paper is organized as follows. Section 2 briefly provides necessary definitions. Section 3 describes the proposed shapelet conversion classification algorithm based on highly efficient subsequence matching. Section 4 includes our experiment on a public dataset, shows the experimental results, and presents our analysis and discussion of the results. Finally, Section 5 summarizes the paper.

2. Definitions and notation

The key terms are as follows:

Time series: A time series is a series of chronologically ordered real data obtained at regular intervals, $T = t_1, t_2, \ldots, t_m$, in which t_i can be any infinite number and m is the length of T.

Time series subsequence: A time series subsequence is a fragment of a complete series, $S = T_i^l = t_i, t_{i+1}, \ldots, t_{i+l-1}$, in which l is the length of S ($l < m$), and i is the subsequence starting position.

Time series classification: For a time series collection with size n, $Q = \{T_1, T_2, \ldots, T_n\}$, in which T_i is consist of m real-valued attributes and a class label c. That is,

$$T_i = < t_1, t_2, \ldots, t_m, c > \tag{1}$$

The task of time series classification is to classify the time series of T_i, and assign class label c to each.

Time series Euclidean distance: The Euclidean distance of time series S_0 and T_0 that are the same length is the sum of corresponding square dot difference, i.e.,

$$\mathrm{dist}(S_0, T_0) = \sum\nolimits_{i=1}^{l} (s_i - t_i)^2 \tag{2}$$

Subsequence distance: Generally, the distance of subsequence S and time series T is the minimum distance of all series of T with length l to S, i.e., $\mathrm{dist}(S, T) = \min_i \mathrm{dist}(S, T_i^l)$.

3. Shapelet transformation classification algorithm based on efficient subsequence matching

The shapelet transformation method is much more accurate than traditional classification algorithms. However, the high computational complexity of the optimal shapelet extraction process is very time consuming. Therefore, the efficient subsequence matching algorithm was introduced to the shapelet transformation method. The efficient subsequence matching algorithm applies the strategy of roughly screening first, then finely screening second, which eliminates unnecessary calculations based on rough estimates to obtain a set of possible matching subsequence. Then, it uses the DTW distance calculation method to accurately calculate the final matching subsequence and the distance. Applying an efficient subsequence matching algorithm during the optimal shapelet extraction process can significantly reduce series distance calculation complexity and ultimately improve algorithm classification efficiency.

3.1. PAA time series representation

PAA representation was applied to high-dimensional time series to achieve efficient storage and simplified computation. PAA representation is a general approximation representation method, which was proposed by Keogh (2011). It is useful for dimension reduction of time series, it has relatively good indexing speed and flexibility, and it also slightly de-noises. As shown in **Figure 1**, PAA representation segments time series based on fixed length, which divides the series into same-length segments and takes the average of each segment to approximately represent the series segments and establish an index.

PAA representation is determined by the time series' compression ratio *v* (ie segment length); the larger the *v*, the greater the dimension reduction, which means more information will be lost; on the contrary, the smaller the *v*, the less the dimension reduction, which means higher approximate representation quality. Therefore, when applying PAA representation, it is important to balance dimension reduction and quality.

3.2. Efficient subsequence matching algorithm

The most basic but deterministic part of time series data mining tasks is calculating the distance between the time series and matching based on their similarities. The commonly used methods for calculating the distance for a large number of high-dimensional, non-aligned time series are very computationally complex, which means that they are very time consuming despite simple Euclidean distance. Vineetha Bettaiah et al. (2014) proposed an efficient time series subsequence matching method to solve this problem. The method ignores small fluctuations within the time series and identifies crests and troughs that will significantly determine the overall shape of time series. It treats local maximum and minimum points as the main breakpoints, segments the time series, matches the rough prior to the actual distance computation to get possible matching series segments, and computes the accurate value.

Algorithm 1: Efficient_subsequence_matching (T_1, T_2)

```
($p_1, p_2, p_3, \ldots, p_N$) = Finding_Breakpoints ($T_1$);
($q_1, q_2, q_3, \ldots, q_M$) = Finding_Breakpoints ($T_2$);
A = Relational_Matrix ($p_1, p_2, p_3, \ldots, p_N$);
B = Relational_Matrix ($q_1, q_2, q_3, \ldots, q_M$);
C = Matching_Matrix (A, B);
Matching_List = Matching_Breakpoints (C);
return (Matching_List);
```

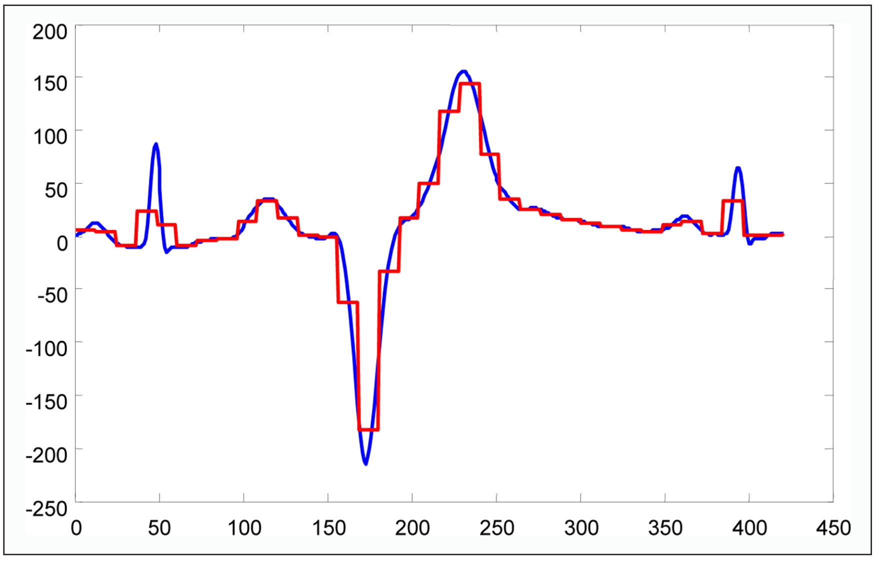


Figure 1: PAA representation of time series.

The algorithm first divides the time series into monotonous non-decreasing segments and monotonous non-increasing segments. It then treats each endpoint segment as the local minimum or minimum value of each time series, and calculates based on the increment (decrement) after the maximum value. It calculates the average increment or decrement value of the corresponding maximum value, selects the points with absolute values above the average as key breakpoints, and then creates indexes with its corresponding series number in the time series and point value. It then checks and gets the time series between the adjacent local minimum value points to ensure no omissions exist, and gets the final set of key segments. As shown in **Figure 2**, the time series partition with the key time series segment breakpoints and endpoints.

Create a set $\{p_1, p_2, p_3, \ldots, p_N\}$ with the key breakpoints extracted from the time series T_1, and construct a N*N logical matrix **A** with this set a_{ij}, which has any elements in **A**, is a vector from p_i to p_j, which indicates the relationship between p_i and p_j. Similarly, construct the M*M logical metrics **B** with key breakpoints $\{q_1, q_2, q_3, \ldots, q_M\}$. If the relationship between p_i and p_j within T_1 is similar to the relationship between q_l and q_k within T_2, then the logical vector a_{ij} and b_{lk} are approximately the same. In this case, the series of points p_i and p_j may match series of q_l and q_k, and point p_i corresponds to q_l, p_j corresponds to q_k, respectively.

Iterate through vectors in matrices **A** and **B** to construct a matching matrix **C**, and compute the matching of each breakpoint in **C**. If c_{il} of **C** is a large value, points p_i and q_l is most likely match; if the value of c_{jk} is 0, p_j and q_k are less likely a match. The algorithm provides a rough estimate and may lead to false positives. It therefore requires verifying calculations after the matching process to remove false matches. Then, it determines the ultimate matching points according to the value, calculates the accurate distance, and takes the minimum as the distance of the time series subsequence.

3.3. Shapelet transformation classification algorithm based on efficient subsequence matching

Shapelet conversion classification algorithms extract the local time series characteristics, ignore data without obvious features, and replace overall data with distinguishing parts to classify. Shapelet conversion algorithms have greatly improved efficiency and accuracy, but the computational complexity of the shapelet extraction process is still high. For a dataset Q with n time series of length m, the candidate shapelets series number is $O(nm^2)$, and the computation complexity for the distance of each shapelet and Q is $O(nm^2)$, thus, the complexity of the entire shapelet extraction algorithm reaches $O(n^2m^4)$. Therefore, shortening the time series length or simplifying the calculation distance can effectively improve the shapelet extraction algorithm efficiency. So, the PAA time series representation and an efficient subsequence matching algorithm were correspondingly introduced to improve shapelet time series classification efficiency.

Since the original time series is too long and its classification features may only be reflected in some segments, using a common classifier will produce results only slightly better than random guessing, which provides no practical value. Therefore, features are extracted in a training set, namely shapelets extraction, to extract a class of time series that is most different from other fragment types. When dealing with the new dataset, the shapelets are used to transform the original time series, and then build a common classifier for

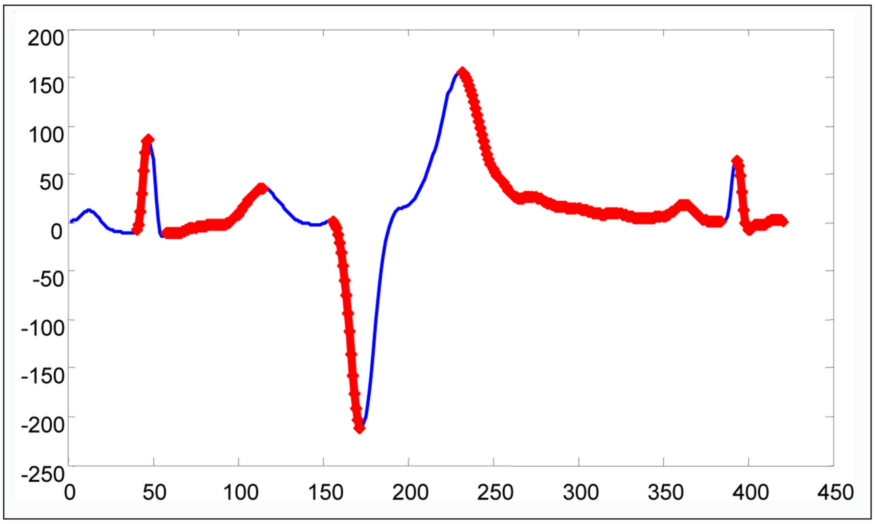


Figure 2: Subsequence matching section.

classification. As shown in **Figure 3**, the marked part is one of the series which has better distinguishing features, i.e., the optimal shapelet.

3.3.1. Standardization and dimension reduction of the original series

Scaling may be different in the experimental data, so it is necessary to standardize to ensure that matching is performed in the same dimension to achieve the best matching results. Then, use the PAA representation mentioned in section 3.1 to perform dimension reduction to the original data within an acceptable simplification range. To represent $T = t_1, t_2,\ldots, t_m$ with PAA representation with segment length v, we get $T_i = t_1', t_2',\ldots,t_{m/v}'$, wherein the segment length v is the compression ratio. It has good approximation to use PAA representation to represent time series, which can effectively achieve dimensional reduction of the original time series.

3.3.2. Shapelet candidate selection

Generally, the algorithm iterates original time series with a specified range with a sliding window algorithm to obtain all shapelet candidates. For a time series containing n datasets $Q = T_1, T_2,\ldots, T_n$, the candidate set of its shapelets series is the union of candidate sets of each series. Setting the shapelet length as l, we can obtain $(m–l) + 1$ shapelet candidates within a time series of length m. The standardized subsequence of length l obtained from the series can be expressed as $W_{i,l}$, then, all subsequence sets of length l in dataset Q are:

$$W_l = \{W_{1,l} \cup W_{2,l} \cup \ldots \cup W_{n,l}\} \tag{3}$$

Then, all candidate shapelets set within Q are:

$$W = \{W_{min} \cup W_{min+1} \cup \ldots \cup W_{max}\} \qquad \min \geq 3, \quad \max \leq m \tag{4}$$

Set W includes $|W| = \sum_{l=\min}^{\max} n(m-l+1)$ candidate shapelets.

3.3.3. Efficient series matching algorithms to extract the optimal shapelets

Due to high computation requirements, the time series distance calculation generally uses a simple Euclidean distance metric. From Section 2, we know that we can take the minimum distance of S and all subsequence in T_i with length l as the distance between the time series T_i and shapelet S of length l, i.e.,

$$D_{S,i} = \text{dist}(S, T_i) = \min_i \text{dist}(S, T_i') \tag{5}$$

Shapelet extraction tasks determine the most distinguished shapelets. Thus, absolute subsequence distance accuracy is not required. We can calculate the distance of shapelet S to all series in dataset Q with an effective subsequence matching algorithm:

$$D_S = < D_{S,1}, D_{S,2},\ldots, D_{S,n} > \tag{6}$$

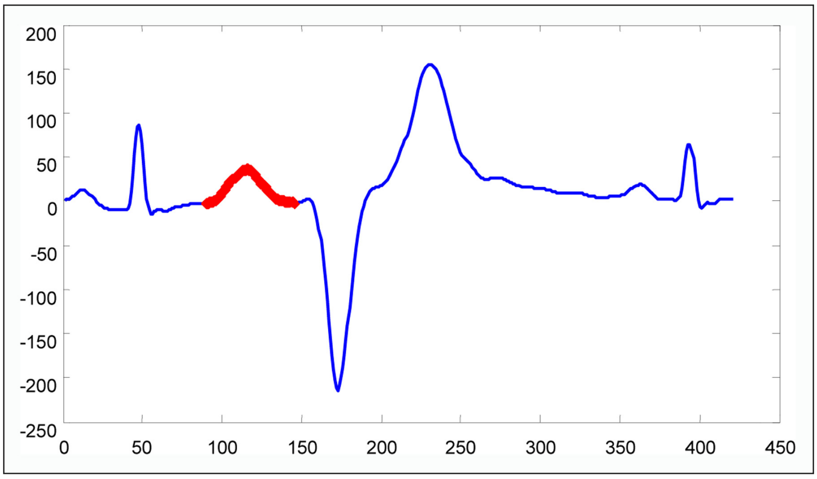


Figure 3: Time series shapelet.

We need to assess shapelet quality to obtain the best classification shapelets. The most common methods are information gain, the Kruskal-Wallis test, the F statistical test, and the Mood median test. We use the classification quality of each shapelet as an indicator to sort all shapelets and select the first k_0 shapelets as the preliminary results.

We need to process the preliminary shapelets to make shapelets more accurately and comprehensively represent the time series class characteristics. First, there could be overlapping shapelets when they are extracted from the same time series, resulting in redundant computation.

Thus, we need to filter the series with an overlapping exponent *e*, to remove shapelets that overlap more with others. Second, to further reduce the number shapelets, simplify calculation, and extend shapelet dissimilarity, we need to cluster shapelets with exponent *k* and select a shapelet from each class as to represent time series features more comprehensively.

3.3.4. Shapelet transformation of the original series

After the above steps, we obtained the final *k* shapelets. Then, the shapelets were used to transform the original series. Shapelet transformation converts the shapelet classification problem to a general classification problem, so that the solution is no longer restricted to a decision tree, but a variety of common classifiers.

Shapelet transformation is achieved by calculating the subsequence distance. For dataset *Q*, we calculated the distance of T_i to *k* shapelets subsequence $D_{i,1}, D_{i,2},\ldots, D_{i,k}$, where $D_{i,k} = \text{dist}(S_k, T_i)$. We created $P_i = D_{i,1}, D_{i,2}, \ldots, D_{i,k}$ as a new entity in the dataset, and constructed $P_1, P_2,\ldots, P_n$ as a new dataset *P*, i.e., we transformed the dataset. In the new dataset *P*, the entity P_i represents the original time series T_i, and each column attributes of the entity was associated with a shapelet. We used a common classifier to classify the new dataset P to determine the class of the original series.

Algorithm 2: Improved_Shapelets_Transform (T_1,T_2)

```
for $T_{\mathrm{i}}$ in $Q$ do
    $T_{\mathrm{i}}$ = PAA ($T_i$, $v$);
    for $l$ = min to max do
        $W_{i,l}$ = Slidingwindow_Traverse ($T_i$, $l$);
        for $S$ in $W_{i,l}$ do
            Matching_List = Efficient_sunseries_matching ($S$, $T$);
            $D_s$ = Calculating_Sub_Distance (Matching_List);
            $quality_s$ = Evaluation ($S$, $D_s$);
            shapelets.add ($S$, $quality_s$);
shapelets = Taking_First_$k_0$(Reorder (shapelets, $quality_s$));
shapelets = Filter_Selfsimilar (shapelets);
k_shapelets = Cluster (shapelets, $k$);
$P$ = Shapelets_Transform ($Q$, k_shapelets)
Classification_Result = General classification ($P$);
return (Classification_Result);
```

4. Computational Experiments

The experiments were conducted in the Java environment integrating with the Weka platform. The computer's configurations were as follows: Windows 7, 8G memory, Intel (R) Core (TM) i7-3770 CPU @ 3.40 GHz.

The experiments were designed to verify the feasibility of integrating the PAA representation and efficient subsequence matching method into the shapelets conversion classification algorithm. The experiments consisted of the following steps:

1. To select the appropriate parameters of PAA Representation, we applied two different time series classification methods, including direct classification and the shapelet classification method based on PAA Representation. We completed ten-fold cross validation on the classification of the whole dataset with the Naive Bayes classifier and analyzed the runtime and classification accuracy.
2. We applied conventional shapelet extraction based on PAA Representation with and without efficient sequence matching to process the whole dataset respectively, and compare the computation complexity.
3. We completed train-test classification with SVM, logistic regression, C4.5 decision trees, random forests, and other general classification algorithms to verify the improved algorithm's accuracy.

4.1. Test Data

Part of the experimental data consisted of five datasets from the UCR Time Series Database including ECGFiveDays, GunPoint, DiatomSizeReduction, Ham, and Herring. The rest comes from UCI series library shared by Professor Keogh's experiment team at the University of California, which included a total of 8 datasets of the X-ray image contour series of human finger bones at different ages (infant, youth, juvenile). As shown in **Table 1**, these 13 public datasets were divided into training and test sets in the experiments. The experimental data was considered to be generalized and representative because records with various time series, lengths, and classes were included in the datasets.

4.2. Quality Evaluation of Shapelets Extraction

In the early stage, information gain was characterized as the indicator of shapelets extraction quality (Ye and Keogh 2011; Mueen and Keogh 2011). Information gain (IG) is an asymmetric metric measurement method used to measure the difference between two probability distributions. In classification, information gain is calculated in terms of data properties, and can be used to measure each property's information size. In section 3.3, based on the sorted distance set D_s, the quality of candidate series S can be evaluated by calculating the maximum information gain of every possible split point (sp).

Relative information gain using KW, F-stat, and MM does not need clearly segmented D_s, and can significantly reduce the overhead time (Lines and Bagnall 2012). Jon Hills et al. (2014) demonstrated that in most time series dataset classifications, F-stat performed better in classification accuracy and time consumption in shapelet quality evaluation compared with other indicators. They suggested, "The F-stat should be the default choice for shapelet quality."

The F statistic is used for testing hypotheses on the mean difference of the dataset consisting of C class samples. The statistical value of the hypothesis test indicated the difference proportion within and between groups. The greater the statistical value, the greater the difference between groups and the smaller the difference within a group. High-quality shapelets have smaller distances to inner class members, and have larger distances to members outside the class. Therefore, shapelets with a good classification quality will generate greater F-stat values. For $D_s = <D_{s,1}\ D_{s,2},\ldots,D_{s,n}>$, they will be grouped based on their categories so that D_i may include all distances between the candidate shapelet *S* and the time series in the corresponding category *i*. Then, the F-stat for quality evaluation of shapelet *S* is:

$$F = \frac{\sum_i \frac{\left(\bar{D}_i - \bar{D}\right)^2}{C-1}}{\sum_{i=1}^{C} \sum_{d_j \in D_i} \frac{\left(d_j - \bar{D}_i\right)^2}{n-C}} \tag{7}$$

n is the number of time series, $\bar{D}$ is the overall mean of *D*, and $\bar{D}_i$ is the average distance from the shapelet to all time series in category *i*.

Table 1: Test data.

Datasets	Partition	Instances (train/test)	Length	Number (classes)
ECGFiveDays	Train/Test	23/861	136	2
GunPoint	Train/Test	50/150	150	2
DiatomSizeReduction	Train/Test	16/306	345	4
Ham	Train/Test	109/105	431	2
Herring	Train/Test	64/64	512	2
DP_Little	Train/Test	400/645	250	3
DP_Middle	Train/Test	400/645	250	3
DP_Thumb	Train/Test	400/645	250	3
MP_Little	Train/Test	400/645	250	3
MP_Middle	Train/Test	400/645	250	3
PP_Little	Train/Test	400/645	250	3
PP_Middle	Train/Test	400/645	250	3
PP_Thumb	Train/Test	400/645	250	3

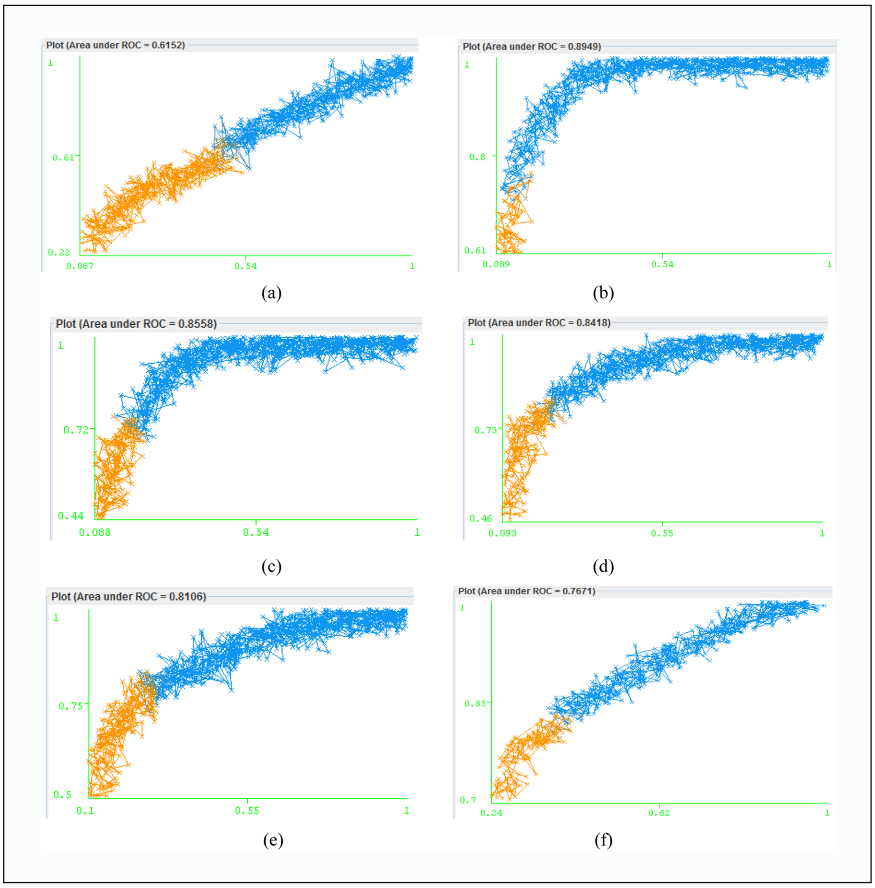
4.3. PAA compression ratio selection

The PAA representation compression ratio directly affects the reduction degree and the time series information integrity. The time series features need to be reserved as much as possible in classification. Therefore, both the simplification and accuracy degree should be considered in the compression ratio selection. The following experiments were conducted to select the appropriate compression ratio.

Experiment 1: We selected 100 shapelets with a length of 5–30 and the compression ratio of 1–5 in PAA Representation. We analyzed classification accuracy based on the ROC curve and the AUC area below it.

Figure 4 shows the results of a representative experiment generated with DP_Middle dataset. **Figure 4(a)** shows the ROC curve by applying the direct classification without shapelet extraction. **Figure 4(b–f)** show the classification results after applying shapelets extraction and PAA Representation. **Table 2** shows the detailed AUC values and the corresponding run times.

The AUC value in **Figure 4(a)** was about 0.62, which was only slightly higher than random guessing accuracy. This is because the feature segments with characteristic identification are only a small part of the entire time series, and in direct classification, it is difficult to identify their characteristics with other influencing factors such as noisy data. As a result, the time series cannot be accurately classified. The AUC values in **Figure 4(b–f)** gradually reduced from 0.89 to 0.77, and the run times reduced from 72.5 hours to



(a) (b) (c) (d) (e) (f)

Figure 4: The ROC curve under different compression ratio.

Table 2: Computing time and the value of AUC.

Value of v	Computing time (s)	The value of AUC
–	–	0.6152
v = 1	261097	0.8949
v = 2	55889	0.8558
v = 3	20790	0.8418
v = 4	9970	0.8106
v = 5	5193	0.7671

1.4 hours, which was due to the increase of dimension reduction and information loss resulting from the increasing compression ratio.

Therefore, shapelet extraction can significantly improve time series classification accuracy. As *v* increases, runtime decreases and classification accuracy gradually decreases. After analysis and comparison, when $v = 3$, run time and accuracy achieve a balance for favorable experimental results. So, the following experiments were developed with $v = 3$.

4.4. Shapelet classification algorithm based on efficient sequence matching

The following experiments were designed to validate the feasibility of the new algorithm based on PAA representation and the efficient subsequence matching method on shapelet extraction optimization and significant computational complexity reduction.

Experiment 2: We selected 100 shapelets with a length of 5–30 and a compression ratio of 3 in PAA representation. We applied conventional shapelets extraction, shapelets extraction combined with PAA representation, and shapelets extraction based on both PAA representation and efficient sequence matching, and recorded the run times. **Table 3** shows the results.

From **Table 3**, for all of the time series datasets involved in the experiment, utilizing PAA representation and efficient subsequence matching in shapelet extraction significantly improved computational efficiency. The shapelet extraction process of the ECGFiveDays Dataset was accelerated 21.3 times, and the remaining datasets were accelerated by about 28–32 times. It was inevitable that the experiment would suffer from time inefficiencies, such as computing preparation time. The small magnitude of the ECGFiveDays dataset affected the results. However, the time consumption was negligible for the remaining datasets with larger magnitudes.

Experiment 3: We selected 100 shapelets with lengths of 5–30 and a compression ratio of 3 in PAA representation to complete the "train – test" standard classification. First, we applied optimal shapelet extraction to training datasets; then, we utilized shapelets to convert the training datasets, and used SVM, logistic regression, C4.5 decision trees, random forests, and other general classification algorithm to classify the converted datasets. Classification accuracy as shown in **Table 4**.

These classification algorithms showed good performance in converted dataset classification accuracy. The AUC values were generally 0.7 or more. The optimal classification algorithm can even make the AUC values be 0.85 or more on datasets except Ham. The accuracy of the Ham dataset was relatively low due to high data similarity. As shown in **Figure 5**, comparing the accuracy of different classification algorithms on different datasets, the SVM and random forest performed better on the time series datasets with smaller magnitudes. With the increase of magnitude, the logistic regression algorithm surpassed other algorithms and achieved the highest accuracy, while the SVM classifier still showed good performance. Overall, the accuracies of the C4.5 decision tree and the KNN classification algorithm were relatively low, while the SVM classifier generated the optimal classification results.

Table 3: Comparison of computing time (s) between the improved and original algorithm.

Datasets	Traditional shapelet	Shapelet extract with PAA	Shapelet extract with PAA and efficient subsequence matching	Upgrade multiples of computing speed
ECGFiveDays	32	3.6	1.5	**21.3**
GunPoint	195	16.4	6.7	29.1
DiatomSizeReduction	1334	128	46.4	28.75
Ham	6211	577	204	30.44
Herring	4873	365	151	**32.27**
DP_Little	37541	3057	1287	29.17
DP_Middle	38378	3106	1324	28.98
DP_Thumb	38332	3096	1318	29.08
MP_Little	38454	3122	1357	28.34
MP_Middle	37661	3084	1306	28.84
PP_Little	38339	3155	1388	**27.62**
PP_Middle	37854	3088	1315	28.79
PP_Thumb	38287	3135	1373	27.89

Table 4: General classifier Accuracy value using improved algorithm.

Datasets	C4.5 Decision Tree	Logistic Regression	SVM	Random Forests	KNN	Naïve Bayesian
ECGFiveDays	0.9334	0.9413	**0.9614**	**0.9735**	0.9512	0.9566
GunPoint	0.9323	0.9411	**0.9812**	**0.9633**	0.9025	0.9364
DiatomSizeReduction	0.8324	0.8847	**0.9077**	0.8522	**0.9211**	0.8913
Ham	0.7987	0.8214	**0.8425**	**0.8333**	0.8327	0.8185
Herring	0.8668	0.8843	**0.9102**	**0.9121**	0.8992	0.9058
DP_Little	0.7445	**0.8753**	**0.8541**	0.8336	0.7525	0.8425
DP_Middle	0.7300	**0.8777**	**0.8635**	0.8377	0.7356	0.8418
DP_Thumb	0.7364	**0.8784**	**0.8621**	0.8324	0.7412	0.8455
MP_Little	0.7544	**0.8784**	**0.8758**	0.8367	0.7664	0.8441
MP_Middle	0.7468	**0.8823**	0.8654	0.8552	0.7630	**0.8663**
PP_Little	0.7568	**0.9002**	**0.8734**	0.8651	0.7811	0.8667
PP_Middle	0.7633	**0.8987**	0.8787	0.8600	0.7798	**0.8792**
PP_Thumb	0.7618	**0.9013**	**0.8842**	0.8631	0.7744	0.8725

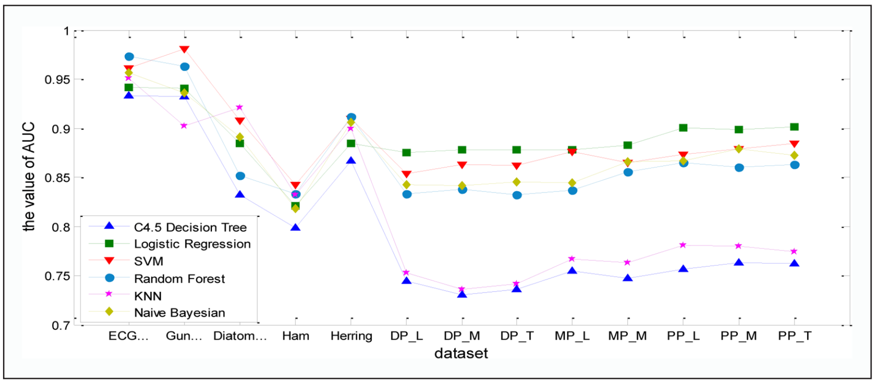


Figure 5: Accuracy comparison with different classifiers.

As discussed above, combined with the PAA representation and efficient sequence matching algorithm, the efficiency of shapelets conversion classification algorithm can be improved, and run time can be reduced. The improved shapelets conversion classification algorithm had better adaptability. It kept high classification accuracy with various classifiers, in which SVM, logistic regression, and random forests integrating with efficient sequence matching have relatively better performance.

5. Conclusions

In this paper, we proposed improved shapelet conversion classification algorithm, which integrated PAA representation with efficient sequence matching algorithms. The improved algorithm effectively solved time consumption problems in the optimal shapelet extraction process, greatly improved computational efficiency, and efficiently and accurately classified the high-dimensional time series e. We performed experiments on 13 experimental datasets. The results showed that the improved shapelets classification algorithm had general feasibility in achieving better classification results in different time series types and magnitudes. Future work would examine ways to further improve subsequence-matching speed, seek better methods for dimension reduction instead of PAA notation, and analyze the adaptability of various classifiers on shapelets classifications.

Funding Information

We are grateful to Tony from the University of East Anglia, who provided helpful shapelet code and data. This work was financially supported by the National Youth Science Foundation of China (No.61503272), Scientific and technological project of Shanxi province of China (No.201603D221037-2).

Competing Interests

The authors have no competing interests to declare.

References

Aghabozorgi, S and **Wah, T Y** 2014 Clustering of large time series datasets. *Intelligent Data Analysis*, 18(5): 793–817.

Aljumeily, D and **Hussain, A J** 2015 The performance of immune-based neural network with financial time series prediction. *Cogent Engineering*, 2(1): 985005.

Bagnall, A, Davis, L, Hills, J and **Lines, J** 2012 Transformation Based Ensembles for Time Series Classification. DOI: https://doi.org/10.1137/1.9781611972825.27

Batista, G E A P A, Wang, X and **Keogh, E J** 2011 A Complexity-Invariant Distance Measure for Time Series. In: *Eleventh Siam International Conference on Data Mining*, 699–710. SDM 2011, April 28–30, Mesa, Arizona, Usa. DOI: https://doi.org/10.1137/1.9781611972818.60

Begum, N and **Keogh, E** 2014 Rare time series motif discovery from unbounded streams, VLDB Endowment. DOI: https://doi.org/10.14778/2735471.2735476

Bettaiah, V and **Ranganath, H S** 2014 An effective subsequence-to-subsequence time series matching approach. In: *Science and Information Conference,* 112–122. DOI: https://doi.org/10.1109/SAI.2014.6918179

Buza, K A 2011 Fusion methods for time-series classification, Ph.D. thesis. University of Hildesheim, Germany.

Deng, H, Runger, G, Tuv, E and **Vladimir, M** 2013 A time series forest for classification and feature extraction. *Information Sciences*, 239(4): 142–153. DOI: https://doi.org/10.1016/j.ins.2013.02.030

Ding, H, Trajcevski, G, Scheuermann, P, Wang, X and **Keogh, E** 2008 Querying and mining of time series data. *Proceedings of the Vldb Endowment*, 1(2): 1542–1552. DOI: https://doi.org/10.14778/1454159

Esling, P and **Agon, C** 2012 Time-series data mining. *Acm Computing Surveys*, 45(1): 1–34. DOI: https://doi.org/10.1145/2379776.2379788

Hartmann, B and **Link, N** 2010 Gesture recognition with inertial sensors and optimized DTW prototypes. In: *IEEE International Conference on Systems Man and Cybernetics*, 2102–2109.

He, Q, Dong, Z, Zhuang, F, Shang, T and **Shi, Z** 2012 Fast Time Series Classification Based on Infrequent Shapelets. In: *International Conference on Machine Learning and Applications*, 215–219. DOI: https://doi.org/10.1109/ICMLA.2012.44

Hills, J, Lines, J, Baranauskas, E, Mapp, J and **Bagnall, A** 2014 Classification of time series by shapelet transformation. *Data Mining & Knowledge Discovery*, 28(4): 851–881.

Jeong, Y S, Jeong, M K and **Omitaomu, O A** 2011 Weighted dynamic time warping for time series classification. *Pattern Recognition*, 44(9): 2231–2240. DOI: https://doi.org/10.1016/j.patcog.2010.09.022

Lines, J and **Bagnall, A** 2012 Alternative quality measures for time series shapelets, Intelligent data engineering and automated learning (IDEAL), *Lect Notes Comput Sci*, 7435, 475–483.

Lines, J, Davis, L M, Hills, J and **Bagnall, A** 2012 A shapelet transform for time series classification. In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 289–297. DOI: https://doi.org/10.1145/2339530.2339579

Mueen, A, Keogh, E and **Young, N** 2011 Logical-shapelets:an expressive primitive for time series classification. In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1154–1162. DOI: https://doi.org/10.1145/2020408.2020587

Petitjean, F, Forestier, G, Webb, G I, Nicholson, A E, Chen, Y and **Keogh, E** 2015 Dynamic Time Warping Averaging of Time Series Allows Faster and More Accurate Classification. In: *IEEE International Conference on Data Mining*, 470–479. DOI: https://doi.org/10.1109/ICDM.2014.27

Rakthanmanon, T, Campana, B, Mueen, A, Batista, G, Westover, B, Zhu, Q, Zakaria, J and **Keogh, E** 2012 Searching and mining trillions of time series subsequences under dynamic time warping. In: *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 262–270. DOI: https://doi.org/10.1145/2339530.2339576

Rakthanmanon, T and **Keogh, E** 2013 Fast shapelets: A scalable algorithm for discovering time series shapelets. *Proceedings of the 2013 SIAM International Conference on Data Mining.* DOI: https://doi.org/10.1137/1.9781611972832.74

Shajina, T and **Sivakumar, P B** 2012 Human Gait Recognition and Classification Using Time Series Shapelets. In: *International Conference on Advances in Computing and Communications*, 31–34. DOI: https://doi.org/10.1109/ICACC.2012.8

Ye, L and **Keogh, E** 2009 Time series shapelets: a new primitive for data mining. In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 947–956. Paris, France, June 28–July. DOI: https://doi.org/10.1145/1557019.1557122

Ye, L and **Keogh, E** 2011 Time series shapelets: a novel technique that allows accurate, interpretable and fast classification. *Data Mining & Knowledge Discovery*, 22(1–2): 149–182.

How to cite this article: Wang, H, Li, C, Sun, H, Guo, Z and Bai, Y 2018 Shapelet Classification Algorithm Based on Efficient Subsequence Matching. *Data Science Journal,* 17: 6, pp. 1–12, DOI: https://doi.org/10.5334/dsj-2018-006

Submitted: 08 October 2017 **Accepted:** 02 February 2018 **Published:** 01 March 2018

]u[*Data Science Journal* is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS