# Unsupervised Discovery of Motifs under Amplitude Scaling and Shifting in Time Series Databases

Tom Armstrong and Eric Drewniak

Wheaton College, Norton MA 02766, USA
{tarmstro,drewniak_eric}@wheatoncollege.edu

**Abstract.** We introduce an algorithm, MD-RP, for unsupervised discovery of frequently occurring patterns, or motifs, in time series databases. Unlike prior approaches that can handle pattern distortion in the time dimension only, MD-RP is robust at finding pattern instances with amplitude shifting and with amplitude scaling. Using an established discretization method, SAX, we augment the existing real-valued time series representation with additional features to capture shifting and scaling. We evaluate our representation change on the modified randomized projection algorithm on synthetic data with planted, known motifs and on real-world data with known motifs (e.g., GPS). The empirical results demonstrate the effectiveness of MD-RP at discovering motifs that are undiscoverable by prior approaches. Finally, we show that MD-RP can be used to find subsequences of time series that are the least similar to all other subsequences.

## 1 Introduction

A time series motif is a subsequence that reoccurs in a data source with relatively high frequency with respect to all subsequences. Finding previously unknown motifs in data is the problem of unsupervised motif discovery. This paper provides a novel approach to representing time series data and extends a motif-discovery algorithm to locate previously undiscoverable motifs. Our representation draws inspiration from speech processing and enables current algorithms to handle motif instances that vary in amplitude either by shifting or scaling. Previous approaches considered motif discovery under uniform scaling in the time dimension only [1] or were not robust to these transformations [2].

The problems of mining time series for patterns, indexing signal data for interesting occurrences, and efficiently querying large datasets for particular subsequence have received significant attention. The formulation of the motif discovery problem comes from the genomics research community interested in finding exact or near-exact genomic subsequences. The extension to real-valued data in multiple dimensions has expanded the potential applications of the work.

P. Perner (Ed.): MLDM 2011, LNAI 6871, pp. 539–552, 2011.

Traditionally, experts or trained analysts scoured over these data to find patterns and interesting subsequences for closer inspection. Relying on human expertise or intervention is, unfortunately, slow and increasingly error prone. For example, an obstetrician may be called away only to return to the fetal heart rate monitor after several minutes – missing key data values. Humans necessarily cannot provide real-time monitoring of massive amounts of real-valued data. Additionally, the proliferation and volume of information make it impossible for humans to solve these problems efficiently for all data. Therefore, we must turn to computational approaches.

In the following sections, we review background on the motif-discovery problem and perspectives from other data mining approaches. Next, we discuss the related approaches and point to a gap in the literature. Then, we detail the algorithm MD-RP and derived features. Finally, we conclude with empirical results and point toward future work.

## 2 Background

The formulation of and approaches to solving the motif discovery problem grew out of handling large quantities of biological data and mining them for information [3,4]. Buhler et al. provided an algorithm using random project to find motifs in biological sequences and defined the problem as:

> "**Planted (l, d)-Motif Problem**: Suppose there is a fixed but unknown nucleotide sequence $M$ (the *motif*) of length $l$. The problem is to determine $M$, given $t$ nucleotide sequences each of length $n$, and each containing a planted variant of $M$. More precisely, each such planted variant is a substring that is $M$ with exactly $d$ point substitutions." [5]

To cast the motif discovery problem in real-value time series into this problem framework, we being with the symbolic aggregate approximation (SAX) as our initial time series representation [6]. SAX, and more recently iSAX [7], has gained traction as the *de facto* representation for time series in the large space of representation choices [6]. SAX allows for a variable-sized alphabet in its discretization. The alphabet symbols have an implicit total ordering. SAX uniquely has a lower-bounding distance measure. Figure 1 contains a plot of the original time series above a plot of the SAX representation where $|\Sigma| = 8$. Our approach uses the SAX representation of the time series as the basis for our derived representation.

How we derive a new set of features from the SAX representation is motivated by the augumented features used in speech processing. Finding recurring episodes in signals, segmenting those signals, and querying the data for particular subsequences are common challenges in speech processing. For example, in a speech processing pipeline [8], speech waveforms are preprocessed and sequences

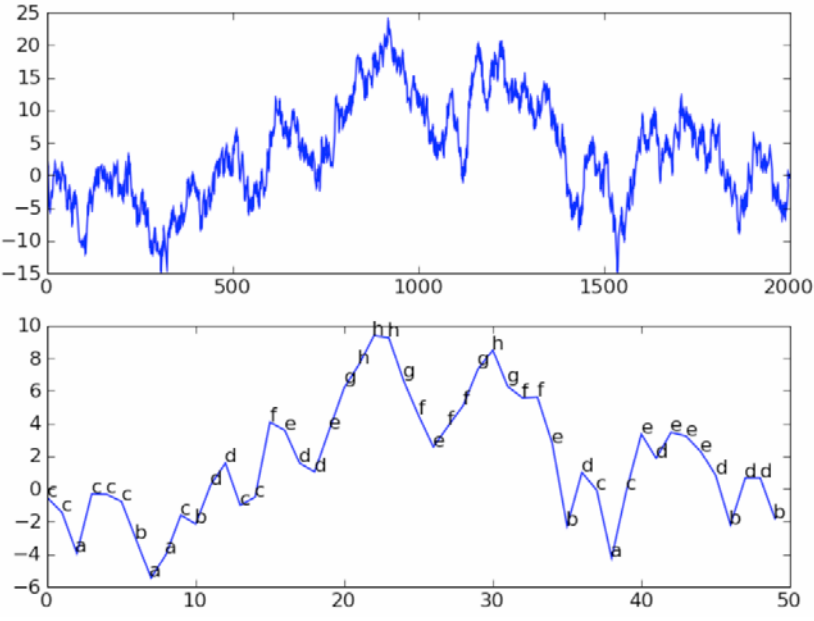


**Fig. 1.** (Top) The original time series. (Bottom) A time series and SAX alphabet symbols where the size of the alphabet is 8.

of spectral feature vectors (e.g., mel-frequency cepstral coefficients) are extracted as a proxy for the original sound.

In addition to the spectral features, the signal representation is augmented with derived features. The first and second derivates, the *deltas* and *delta-deltas*, are added as additional features to capture changes over time in the signal. Motif discovery algorithms have not exploited the potential of this additional information. In biological domains, where the alphabets of the datasets have no ordering, this information is unavailable. But, the SAX string representation of real-valued time series allows for consideration of time-derived features like the deltas and delta-deltas.

## 3 Related Work

Unsupervised discovery of novel, recurring subsequences in time series databases is an open problem. However, given sufficient space and time, an exhaustive indexing of the entire time series solves the motif discovery problem. Many time series domains (e.g., streaming, real-time) and actual space and time requirements preclude a full time series index.

To address the problem of unsupervised motif discovery without a full index, several approaches employ randomized algorithms. One leverages hashing

collisions to indicate likelihood of motifs [9] continuing work done using the piecewise aggregate approximation [10]. An extension to this approach handles uniform scaling of the motifs in the time dimension [1]. Both approaches use the SAX approximation, but the size of the discovered motifs are limited to the length of the user-selected window length. As with different motif lengths, in the multidimensional case, not all motifs occur in every dimension nor over the same time scale in each dimension – the problem of subdimensional motif discovery. One approach extends the univariate randomized algorithm to approximate which dimensions contain the motif exemplars [11].

Other approaches do not explicitly discretize the inputs and do not limit the size of the motifs. The SAND algorithm takes a sampling approach to discover portions of motifs in real-value time series [2] and uses Dynamic Time Warping (DTW) to handle candidate motifs that differ in the time dimension. The small motif chunks are stitched together to form larger motifs. A probabilistic sampling approach can be improved if constraints on the motifs in the time series are known [12]. These approaches have been shown to discover motifs in exercise data [13], episodes in activity data [14], and clusters of mobile robots experiences [15,16].

## 4 Algorithm

We propose a randomized algorithm (called **MD-RP**) that extends the traditional random projection algorithm (RP [9]) through the use of novel representation characteristics. Unlike other extensions that find motifs that vary uniformly in the time dimension [1], we consider finding motifs that vary by an amplitude shift or amplitude scaling.

The stages of MD-RP operate analogously to RP, but the input is a derived representation from SAX features – a longer string of features. The motivation for the derived features comes from speech processing. With speech waveforms, frequently occurring subsequences need not always have the same waveform and may vary in amplitude. Exclusively using the SAX alphabet string representations for time series has potential pitfalls in some domains. For example, depending on the size of the alphabet and the time series, motfis that differ by slight amplitude shifts in the SAX representation may have completely different representations – the motif representations will differ in most positions.

The RP algorithm begins with a SAX representation of a time series which has been windowed and normalized. The alphabet size for the SAX representation is a user-selected parameter (we selected values between 10 and 20). The user also selects the window size to consider when processing the SAX string. RP passes a fixed-length window over the SAX string resulting in a list of equal length substrings – candidate motif windows. Next, the algorithm performs a fixed number of randomized projection rounds to create a collision matrix. In each round, a fixed number of columns are randomly selected as masking columns and the

```
MD-RP(timeseries, w, Σ, n)
 1  SAXstring ← SAX(timeseries, |Σ|)
 2  derived ← DerivedFeatures(SAXstring)
 3  wins ← [ ]
 4  for i < |timeSeries| − w
 5        do
 6           wins ← wins ⋃ [derived[i, i + w]]
 7  for i < n
 8        do
 9           cols ← PickMaskingColumns()
10           collisions ← RandProj(wins, cols)
11           UpdateCollMatrix(collisions)
12  candidateMotifs ← AnalyzeCollMatrix()
```

**Fig. 2.** Motif Discovery-Random Projection (MD-RP) Pseudocode

resulting shortened strings are used as keys into a hash table. Candidate motif windows that hash to the same location are considered similar because their SAX representations match at the unmasked positions. Over the subsequent rounds, strings that are similar in most positions tend to hash to the same location more often than those that do not. A collision matrix stores the hashing collisions between all string pairs. Entries in the collision matrix that exceed a thresholding statistic are considered to be candidate motifs.

Unlike RP, we do not use the SAX string alone as input to the random projection portion of the algorithm. Instead, we developed several derived features to use in place of the SAX string. We selected these features to capture additional motifs for domains that do not always contain perfect motif matches. One underutilized feature of the SAX representation is its implicit ordering of alphabet symbols. There is a strict ordering on alphabet elements, and we exploit this ordering to extend the representation with derived features. Motivated by the speech processing derived features, we augment the SAX strings with three additional features. First, we use the change in value between the SAX string letters as a proxy for the first time derivative. We call this the *delta* or 1st order feature, and for example, the SAX letters *abaa* has a $[1, -1, 0]$ delta feature – unit differences between consecutive alphabet symbols.

Second, we use the change in value between the delta features as the second time derivative. We call this the *delta-delta* or 2nd order feature, and for example, the delta feature $[1, -1, 0]$ has a $[-1, 1]$ delta-delta feature. Third, we approximate the concavity or convexity of the subsequence with the sign of the delta feature. We call this the *shape* feature, and for example, a $[1, -1, 0]$ delta feature has a $[+, -, +]$ shape feature. Consider the SAX string *bbdddaaacacccd* and the derived features for a window of length four in Table 1.

The delta, delta-delta, and shape features are intended to capture amplitude shifting and scaling that would normally result in a non-match with RP. The

**Table 1.** A pre-collision matrix listing for the SAX string *bbdddaaacacccd*

| Index | $1^{st}$ Order | $2^{nd}$ Order | Shape |
|---|---|---|---|
| 00 | 0 2 0 | 2 -2 | + + + |
| 01 | 2 0 0 | -2 0 | + + + |
| 02 | 0 0 -3 | 0 -3 | + + − |
| 03 | 0 -3 0 | -3 3 | + − − |
| 04 | -3 0 0 | 3 0 | − − − |
| 05 | 0 0 2 | 0 2 | + + + |
| 06 | 0 2 -2 | 2 -4 | + + − |
| 07 | 2 -2 2 | -4 4 | + − + |
| 08 | -2 2 0 | 4 -2 | − + + |
| 09 | 2 0 0 | -2 0 | + + + |
| 10 | 0 0 1 | 0 1 | + + + |

delta and delta-delta features primarily serve as a mechanism to account for amplitude shifted motifs. The shape feature captures amplitude scaling where the delta and delta-delta features would not match.

Consider the following three SAX strings: (1) *aaabbbaaa*, (2) *cccdddccc*, and (3) *aaadddaaa*. Given the ordering of the discretization alphabet, these two strings are similar, but (1) and (2) are shifted by two SAX letters. RP using SAX strings alone to discover motifs disallows the possibility for strings (1) and (2) to be instances of the same motif. The delta and delta-delta features match for (1) and (2). Alternatively, (1) and (3) match on some delta and delta-delta features, but do not in the middle of the strings. The shape features, however, do match allowing these two as possible motifs.

We present MD-RP, the augmented version of RP, in Figure 2 where $w$ is the size of the window, *timeseries* is the time series data, $\Sigma$ is the size of the SAX alphabet, and $n$ is the number of random projection rounds. The additional features overall do not change the computational complexity of the algorithm in practice. The only potential increase is on the order of the window size, but the window size is, typically, a small constant to keep this approach tractable.

### 4.1 Thresholding

The selection of potential motifs from the time series depends upon the collision matrix values after multiple rounds of randomized projection. We established a threshold such that any subsequence with a collision matrix entry of at least the threshold value is an instance of a motif. Let $\mu$ be the mean of all collision matrix entries and $\sigma$ be the standard deviation of all collision matrix entries. The threshold is the minimum of $\mu + 2 * \sigma$ and 20 (a cutoff for evaluation). In our experience with the data considered in this paper, the histogram of collision matrix entries appears normally distributed (see Figure 3).

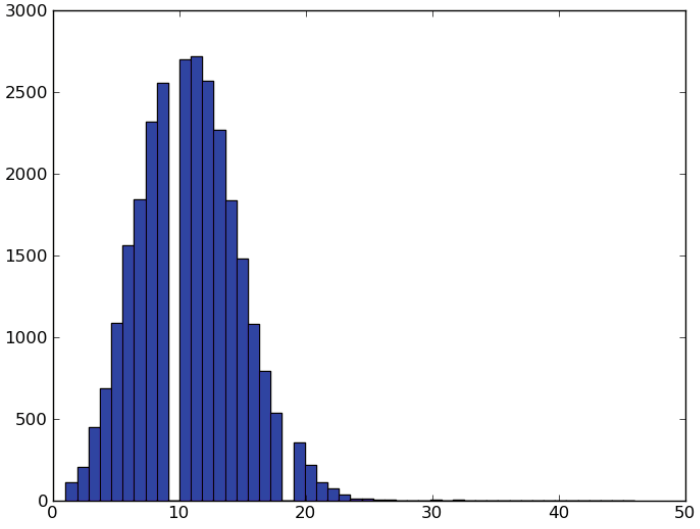


**Fig. 3.** Synthetic Data (see data description in Section 5.1 and Figure 4) Collision Matrix Histogram (x-axis is collision matrix entry values; y-axis is counts)

## 5 Experimental Results

To evaluate the efficacy of MD-RP compared with the established RP, we considered three categories of data: randomly generated synthetic datasets with randomly inserted motifs and real-world datasets with known motif locations. First, we discuss the evaluation of MD-RP and RP on random walk data containing inserted motifs. Second, we consider a baseline industrial dataset with known motifs. Third, we evaluate MD-RP on finding motifs on-body sensor data.

### 5.1 Synthetic Data

We generated a random dataset of length 500 with values sampled from a Gaussian distribution with mean of 0 and standard deviation of 2. To assess the robustness of MD-RP with respect to motifs shifted or scaled in amplitude, we randomly inserted generated motifs in two locations in the dataset. For amplitude shifting, the randomly created motif was the same length and sequence of values, but uniformly shifted a random amount. Below are the datasets generated using this random dataset and randomly generated motifs. Each dataset is accompanied with a figure depicting the pair of motifs discovered by MD-RP, the SAX representation of the motifs, and the motif pair highlighted in the original dataset.

– **Synthetic Sanity Check** (Figure 4)
  We inserted two identical copies of a motif at random locations in the synthetic dataset such that the two were non-overlapping. The motif is a

sequence of 30 values centered about a single peak. This dataset is designed as a sanity check to determine if MD-RP and RP perform equivalently on datasets where we expect them to perform equally.

– **Synthetic Amplitude Shifting Data** (Figure 5)
We inserted two copies of a motif at random locations in the synthetic dataset such that the two were non-overlapping. The motif is a sequence of 30 values centered about a single peak. One copy of the motif was inserted as normal. The other copy of the motif was inserted after increasing each value by 1.5.

– **Synthetic Amplitude Scaling Data** (Figure 6)
We inserted two copies of a motif at random locations in the synthetic dataset such that the two were non-overlapping. The motif is a sequence of 30 values centered about a single peak. One copy of the motif was inserted as normal. The other copy of the motif was inserted scaling each value by a factor of 3 and beginning at the original amplitude.

– **Synthetic Amplitude Scaling and Shifting Data** (Figure 7)
We inserted two copies of a motif at random locations in the synthetic dataset such that the two were non-overlapping. The motif is a sequence of 30 values centered about a single peak. One copy of the motif was inserted as normal. The other copy of the motif was inserted with each multiplied by a factor of 3.

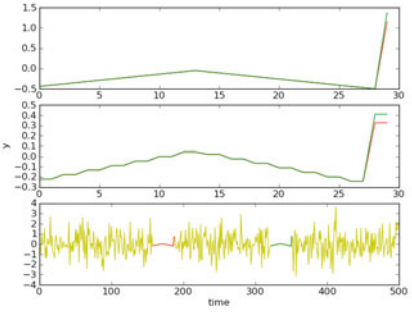


**Fig. 4.** Synthetic Sanity Check: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Random walk data with a pair of inserted motifs – note the overlap in the discovered motifs

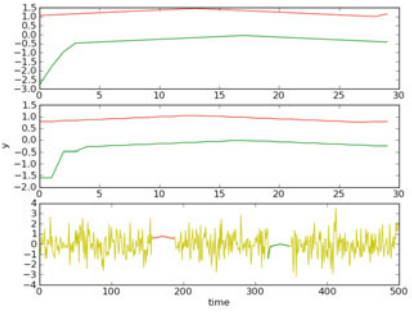


**Fig. 5.** Synthetic Amplitude Shifting: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Random walk data with a pair of inserted motifs that differ by a shift in amplitude – note the lack of overlap in the discovered motifs

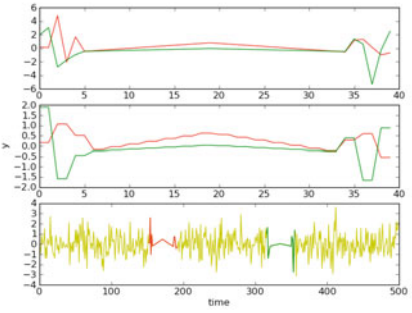


**Fig. 6.** Synthetic Amplitude Scaling: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Random walk data with a pair of inserted motifs that differ by a scaling factor of 3 and beginning at the original amplitude

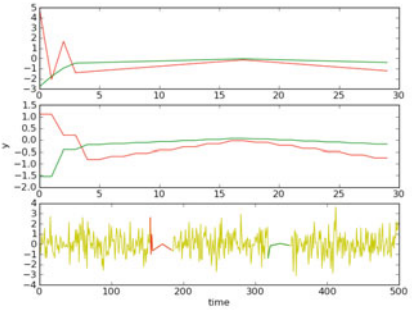


**Fig. 7.** Synthetic Amplitude Shifting and Scaling: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Random walk data with a pair of inserted motifs that differ by a a scaling factor of 3

### 5.2 Real-World Data with Known Motifs

The *winding* dataset is a multivariate industrial time series. We consider the univariate time series of 2500 samples of the angular velocity of a reel sampled at 10 Hz. This dataset is a baseline for identifying known motifs (see Figure 8 for one example). Additionally, as a low-cost proxy for more extensive on-body sensors, we recorded a collection of outdoor exercise routines using a Garmin Forerunner 305. The Forerunner 305 is a GPS-receiver training watch with a heart rate monitoring strap that samples data at 1 Hz. The device enabled us to record multi-variate time series data of a variety of activities. For example, the time series in Figure 10 is a plot of a run around the campus quad and three random instances of slowing to walk. The data points are speed over time and extremely noisy compared to other datasets. A smoothed version of the run dataset was also used to address the amount of noise in the original dataset.

### 5.3 Discord Discovery in Synthetic Data

The threshold for consideration as a time series motif focuses on those subsequences that have a number of collisions larger than two standard deviations above the mean. What then are the windows in the collision matrix that are at least two standard deviations below the mean? A time series *discord* is a subsequence that is least similar to all other subsequences in the time series [7]. We propose to utilize the collision matrix for motif discovery to find the discord(s) of the time series as well. Each position in the collision matrix corresponds to how frequently masked strings for two strings collide. The sum of positions in a row in the collision matrix corresponds to how frequently the string corresponding to

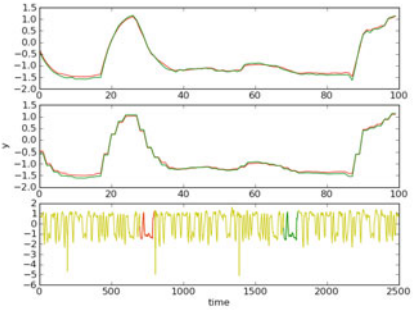


**Fig. 8.** Industrial Winding: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Industrial winding data time series of the angular velocity of reel two complete time series and highlighted motif subsections

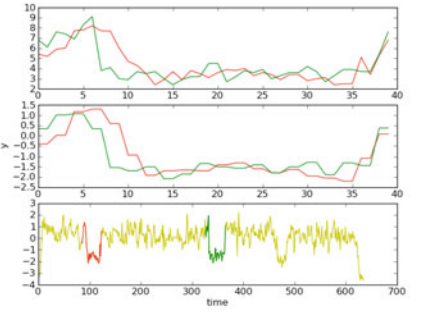


**Fig. 9.** Garmin Forerunner: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Complete Garmin Forerunner speed time series and highlighted motif subsections

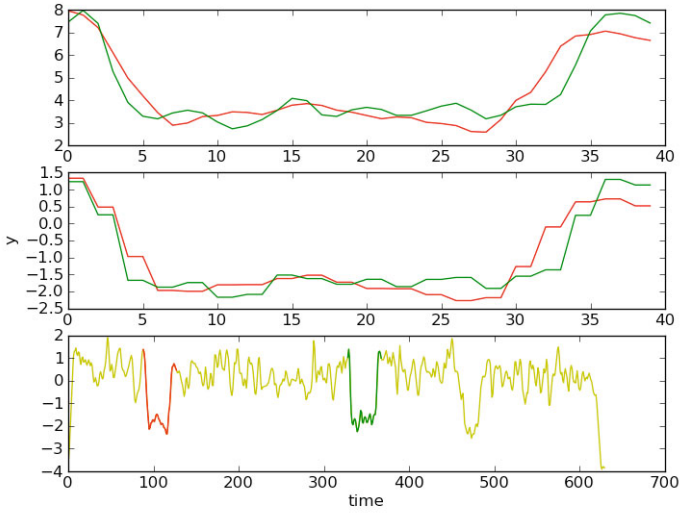


**Fig. 10.** Garmin Forerunner Smoothed: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Complete smoothed Garmin Forerunner speed time series and highlighted motif subsections

the row collides with all other strings. Considering all of the sums of rows in the collision matrix, larger values correspond to subsequences that are more similar to all subsequences. Smaller values correspond to subsequences of the time series that are the least similar to all other subsequences.

We generated a synthetic linear dataset of length 500 with an initial value of $-2.5$ and slope 0.01. At a random position, the data point was replaced with the value increased by 3.5. Using the method described on the collision matrix, only the windows overlapping the position containing the changed value resulted in a row sum at least two standard deviations below the mean of all row sums in the collision matrix.

## 6 Empirical Results and Discussion

The results of the MD-RP algorithm on the synthetic datasets with planted motifs and real-world datasets with known motifs are listed in Table 2. In the sanity check and winding data experiments, we expected both approaches to discover the same motifs, and they do. In the shifting, scaling, and combination of shifting and scaling, MD-RP outperforms RP at discovering the planted motifs and returns fewer spurious motifs than RP. Finally, in the Garmin Forerunner run data, MD-RP does not perform as well unless that data are smoothed.

**Table 2.** Experimental Results

| | | **Datasets** | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **Algorithm** | | Check | Shift | Scale | Scale/Shift | Winding | $\text{Run}_1$ | $\text{Run}_2$ |
| **MD-RP** | TP | **20** | **20** | **16** | **16** | **20** | 9 | **20** |
| | FP | 0 | 0 | 4 | 4 | 0 | 11 | 0 |
| **RP** | TP | **20** | 0 | 3 | 0 | **20** | **16** | **20** |
| | FP | 0 | 11 | 17 | 20 | 0 | 4 | 0 |

The results from our experiments indicate that augmented features allows for discovery of previously ruled out motif pair possibilities. We make two critical observations about our feature selection. First, there exists a many-to-one mapping between SAX string representations and additional feature vectors. That is, two equivalent SAX strings map to the same derived features, and additional unequal SAX strings also map to the same derived features.

This is useful for two reasons: 1) our augmented representation hashes to the same location the same number of times as the original representation; and 2) the multiple strings that map to the same additional features are those that we want to consider as motifs. Second, after empirical analysis, we determined that including the SAX string and the additional features together as the representation decreased the performance of the algorithm. Instead, we exclusively use the derived features and throw out the original SAX strings.

### 6.1 Spurious Motifs

Not all of the top results from the collision matrix contain actual motifs. In the random walk data, we discovered several spurious motifs. For example, in Figure 11, we see two sections of the time series which are random data. The motifs are clearly visible and the algorithm discovers them, but also claims that this pair is a motif.

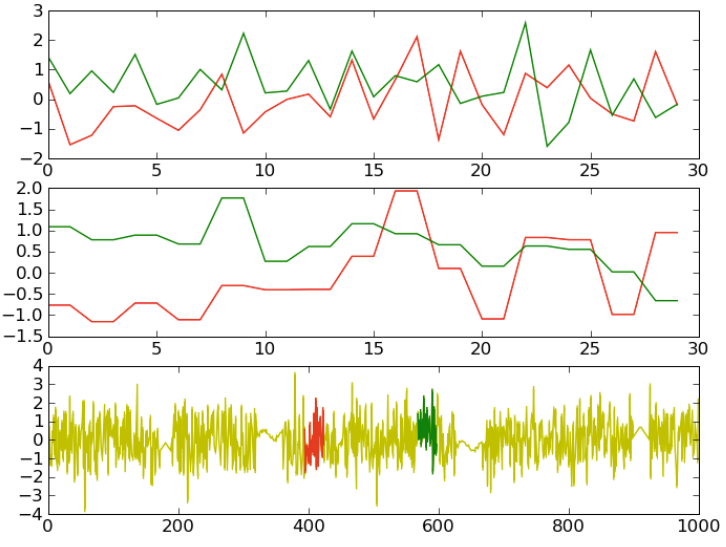


**Fig. 11.** Spurious Motifs: (Top) Motif subsequences overlaid from original dataset; (Middle) Motif subsequences overlaid from SAX representation of original dataset; and (Bottom) Complete random walk time series with inserted peaks and sine waves and highlighted spurious motif subsections

## 7 Conclusion and Future Work

In this paper, we presented motivation to augment traditional discretization of time series data with derived features used in other domains (e.g., speech processing). These additional features, combined with the strict ordering of the SAX alphabet capture useful time series dynamics. Our representation is robust with respect to shifts in amplitude and amplitude scaling.

We presented a survey of empirical results that provided: 1) a sanity check that our representation indeed discovers the same motifs as the prior approach and representation; 2) results on synthetic data demonstrating the advantages of our representation over the original; 3) an application of the representation to a domain (i.e., industrial winding data); and 4) an application to exercise data.

Future work will proceed in several directions. First, we evaluated our approach using univariate time series. We will focus on multivariate time series and addressing issues of dimensionality. Our representation choice is applicable

for use in the related work on subdimensional motif discovery [11] and uniform scaling in the time dimension [1] – both multivariate approaches. We will explore the impacts that our representation choices have on these multivariate time series motif-discovery algorithms. Second, the inspiration for the augmented features comes from speech waveforms – a potentially massive data source filled with interesting motifs. We will investigate how appropriate the SAX representation may be for speech and applications of motif-discovery algorithms to speech.

## References


1. Yankov, D., Keogh, E., Medina, J., Chiu, B., Zordan, V.: Detecting time series motifs under uniform scaling. In: KDD 2007: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 844–853. ACM, New York (2007)
2. Catalano, J., Armstrong, T., Oates, T.: Discovering patterns in real-valued time series. In: Fürnkranz, J., Scheffer, T., Spiliopoulou, M. (eds.) PKDD 2006. LNCS (LNAI), vol. 4213, pp. 462–469. Springer, Heidelberg (2006)
3. Sagot, M.: Spelling approximate repeated or common motifs using a suffix tree. In: Lucchesi, C.L., Moura, A.V. (eds.) LATIN 1998. LNCS, vol. 1380, pp. 374–390. Springer, Heidelberg (1998)
4. Pevzner, P., Sze, S.: Combinatorial approaches to finding subtle signals in DNA sequences. In: Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology, Citeseer, vol. 8, pp. 269–278 (2000)
5. Buhler, J., Tompa, M.: Finding motifs using random projections. Journal of Computational Biology 9(2), 225–242 (2002)
6. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: A symbolic representation of time series, with implications for streaming algorithms. In: DMKD 2003: Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pp. 2–11. ACM Press, New York (2003)
7. Shieh, J., Keogh, E.: Isax: indexing and mining terabyte sized time series. In: KDD 2008: Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 623–631. ACM, New York (2008)
8. Jurafsky, D., Martin, J.: Speech and language processing. Prentice-Hall, New York (2000)
9. Chiu, B., Keogh, E., Lonardi, S.: Probabilistic discovery of time series motifs. In: 9th International Conference on Knowledge Discovery and Data Mining (SIGKDD 2003), pp. 493–498 (2003)
10. Lin, J., Keogh, E., Lonardi, S., Patel, P.: Finding motifs in time series. In: Proceedings of the Second Workshop on Temporal Data Mining, Edmonton, Alberta, Canada (July 2002)
11. Minnen, D., Isbell, C., Essa, I., Starner, T.: Detecting subdimensional motifs: An efficient algorithm for generalized multivariate pattern discovery. In: IEEE Int. Conf. on Data Mining (ICDM), vol. 1 (2007)
12. Mohammad, Y., Nishida, T.: Constrained Motif Discovery in Time Series. New Generation Computing 27(4), 319–346 (2009)
13. Minnen, D., Starner, T., Essa, I., Isbell, C.: Improving activity discovery with automatic neighborhood estimation. In: International Joint Conference on Artificial Intelligence, pp. 6–12 (2007)

14. Vahdatpour, A., Amini, N., Sarrafzadeh, M.: Toward unsupervised activity discovery using multi-dimensional motif detection in time series. In: IJCAI, pp. 1261–1266 (2009)
15. Oates, T.: Identifying distinctive subsequences in multivariate time series by clustering. In: Chaudhuri, S., Madigan, D. (eds.) Fifth International Conference on Knowledge Discovery and Data Mining, pp. 322–326. ACM Press, San Diego (1999)
16. Oates, T., Schmill, M.D., Cohen, P.R.: A method for clustering the experiences of a mobile robot that accords with human judgments. In: AAAI/IAAI, pp. 846–851 (2000)