Journal of Classification 3? (201?)
DOI: 10.1007/s00357-

A Run Length Transformation for Discriminating Between Auto Regressive Time Series

Anthony Bagnall

University of East Anglia, United Kingdom

Gareth Janacek

University of East Anglia, United Kingdom

Abstract: We describe a simple time series transformation to detect differences in series that can be accurately modelled as stationary autoregressive (AR) processes. The transformation involves forming the histogram of above and below the mean run lengths. The run length (RL) transformation has the benefits of being very fast, compact and updatable for new data in constant time. Furthermore, it can be generated directly from data that has already been highly compressed. We first establish the theoretical asymptotic relationship between run length distributions and AR models through consideration of the zero crossing probability and the distribution of runs. We benchmark our transformation against two alternatives: the truncated Autocorrelation function (ACF) transform and the AR transformation, which involves the standard method of fitting the partial autocorrelation coefficients with the Durbin-Levinson recursions and using the Akaike Information Criterion stopping procedure. Whilst optimal in the idealized scenario, representing the data in these ways is time consuming and the representation cannot be updated online for new data. We show that for classification problems the accuracy obtained through using the run length distribution tends towards that obtained from using the full fitted models. We then propose three alternative distance measures for run length distributions based on Gower's general similarity coefficient, the likelihood ratio and dynamic time warping (DTW). Through simulated classification experiments we show that a nearest neighbour distance based on DTW converges to the optimal faster than classifiers based on Euclidean distance, Gower's coefficient and the likelihood ratio. We experiment with a variety of classifiers and demonstrate that although the RL transform requires more data than the best performing classifier to achieve the same accuracy as AR or ACF, this factor is at worst non-increasing with the series length, m, whereas the relative time taken to fit AR and ACF increases with m. We conclude that if the data is stationary and can be suitably modelled by an AR series, and if time is an important factor in reaching a discriminatory decision, then the run length distribution transform is a simple and effective transformation to use.

Keywords: Time series classification; Run length distribution, Auto regressive model approximation.

Authors' Addresses: A. Bagnall and G. Janacek, University of East Anglia, Norwich Research Park, Norwich NR4 7TJ, UK, email: Anthony.Bagnall@uea.ac.uk; G.Janacek@uea.ca.uk.

Published online

1. Introduction

At the heart of any time series data mining task is the requirement to measure the similarity between time series. There are essentially two approaches to modelling time series: models based on autocorrelation function (ACF) and models based on fitted curves. Auto regressive moving average (ARMA) models and the multitude of variants that are derived from the ACF have featured strongly in the statistical literature and have been shown to accurately model many real world data sets. However, the vast majority of the statistical literature focusses on either alternative model fitting methods and structures, or forecasting with models fitted to fairly short series. There has been very little consideration of how best to measure similarity between series in terms of time series objectives such as query, clustering, classification or anomaly detection. The few papers that do consider discriminating between ARMA series tend to concentrate on alternative modelling methods to account for factors such as co-integration (for example, see Maharaj 1996, 1999, 2000) or consider alternative ways of comparing model parameters (Corduas and Piccolo 2008; Liao 2005; Kalpakis, Gada and Puttagunta 2001; Piccolo 1990). Our overall aim is to apply some of the modelling techniques developed in statistics to data mining problems, with the objective of rapid, efficient discrimination. Our major focus is on how best to classify time series, although the techniques we describe translate easily to other problem domains such as clustering and query by content.

In Bagnall, Davis, Hills, and Lines (2012) it is argued that the easiest way to gain improvement on TSC problems is to transform into an alternative data space where the discriminatory features are more easily detected. One transform considered in Bagnall et al. (2012) was the ACF, and classifiers built on the ACF were found to outperform classifiers built in the time domain and the principal component space on several standard test problems. However, one potential problem with using the ACF is the time taken to perform the transform, and the requirement to repeat the transform in the presence of new data. We propose a run length transformation that can detect similarity in auto-correlation without the time consuming step of forming the ACF. A run of length k is a sequence of k contiguous observations that are all above or all below the mean. The run length transform is simply a histogram that counts the occurrences of runs of length $(1, 2, \ldots, s)$ for some predefined $s < m$, where m is the series length. The fundamental motivation behind this research is that the runs distribution varies with the correlation structure of the original series and hence can be used to discriminate between series from different models. The benefit of using the run length transform is that it is simple to perform, easy to understand, can be done in linear time and updated in constant time. Furthermore the run length histogram can be

calculated directly from data that has been already compressed into a binary series of below and above the mean (clipped). This means that the run length transform can be exactly calculated from data already compressed at a rate of between 32:1 to 256:1. In Section 2 we give some background into time series classification and auto regressive models. In Section 3 we describe the run length transform and show that asymptotically the auto correlation function and the run length distribution are theoretically linked. This shows that, with long enough series, class differences embedded in the ACF will also be expressed in differences in the run length histogram. In Section 4 we describe four alternative distance measures that can be used with run length histograms. In Section 5 we describe our algorithm for generating stationary auto regressive series and outline the classification experimental regime we use for Section 6, where we conduct a series of four sets of classification experiments with simulated data sets to evaluate the run length transformation. We perform a case study with real world data in Section 7 and summarize our findings in Section 8.

2. Background

2.1 Time Series Classification (TSC)

We define time series classification as the problem of building a classifier from a collection of labelled training time series. We limit our attention to problems where each time series has the same number of observations. We define a time series $\mathbf{x_i}$ as a set of ordered observations

$$\mathbf{x_i} =< x_{i1}, \ldots, x_{im} >$$

and an associated class label y_i. The training set is a set of n labelled pairs

$$D = \{(\mathbf{x_1}, y_1), \ldots, (\mathbf{x_n}, y_n)\}.$$

TSC problems have been addressed by researchers in a wide range of fields including, but not limited to, data mining, statistics, machine learning, signal processing, environmental sciences, computational biology, and chemometrics. For traditional classification problems, the order of the attributes is unimportant and the interaction between variables is considered independent of their relative positions. For time series data, the ordering of the variables is often crucial in finding the best discriminating features. There are three broad categories of TSC discriminating features which are described by three general approaches to measuring similarity between time series: similarity in time, similarity in shape and similarity in change.

Similarity in time can be quantified by measures such as Euclidean distance or correlation (Douzal-Chouakria, Diallo, and Giroud 2010; Abraham and Tan 2010). Similarity in time is characterized by the situation where the series from each class are observations of an underlying common curve in the time dimension. Variation around this underlying common shape is caused by noise in observation, and also by possible noise in indexing which may cause a slight phase shift. A classic example of this type of similarity is the Cylinder-Bell-Funnel artificial data set, where there is noise around the underlying shape, but also noise in the index of where the underlying shape transitions (see, for example Douzal-Chouakria and Amblard 2012). The majority of research into TSC has concentrated on this area. In data mining research, the commonly used benchmark classification algorithm is 1-NN with an elastic measure such as Dynamic Time Warping (DTW) to allow for small shifts in the time axis (Ding, Trajcevski, Scheuermann, Wang, and Keogh 2008). Alternative approaches used in other research fields include weighted dynamic time warping (Jeong, Jeong, and Omitaomu 2010), support vector machines built on variable intervals (Rodriguez and Alonso 2005), tree based approaches (Douzal-Chouakria and Amblard 2012; Deng, Runger, Tuv, and Vladimir 2011), and fusion of alternative distance measures (Buza 2011).

Similarity in shape describes the scenario where class membership is characterized by a common shape but the discriminatory shape is phase independent. If the common shape involves the whole series, but is phase shifted between instances of the same class, then transformation into the frequency domain is one valid approach (Agrawal, Faloutsos, and Swami 1993). If the common shape is local and embedded in confounding noise, then a subsequence techniques such as Shapelets can be employed (Ye and Keogh 2009; Lines et al. 2012).

Similarity in change relates to the situation where the relevant discriminatory features are related to the autocorrelation function of each series. The most common approach in this situation is to fit an ARMA model then base similarity on differences in model parameters (Corduas and Piccolo 2008; Liao 2005; Kalpakis, Gada and Puttagunta 2001; Piccolo 1990). Related problems include accounting for factors such as co-integration (for example, see Maharaj 1996, 1999, 2000) or using difference (van Wyk, van Wyk, and Qi 2009). The common element to this work is that similarity between series is not measured in the time domain.

Our sole concern lies in problems where classification is best achieved through measures of similarity in change. Techniques for similarity in time will not work for problems where similarity lies in the ACF function. We demonstrate this point in Section 6.5. However, the converse is also true. In Bagnall et al. (2012) we show that, for certain types of TSC problems, class

similarity is detectable in phase independent and/or auto correlation related feature spaces rather than in the time domain. Though complex classifiers may be able to reconstruct this similarity through the internal non-linear mapping they employ to construct the classifier, a far simpler and more intuitive approach is to transform the data into an alternative space and use a basic classifier. We propose a method for detecting differences in series that can be accurately modelled as stationary autoregressive (AR) processes that simple, faster than fitting the whole model and can be updated in constant time.

2.2 Auto Correlation and Auto Regressive Models

One of the most important basic transformations is the autocorrelation function, which measures the interdependence of terms in the time domain. The observed vector of ACF terms of time series $\mathbf{x}$, which we denote $\hat{\rho} =< \hat{\rho_1}, \hat{\rho_2}, \ldots, \hat{\rho_l} >$, is defined as

$$\hat{\rho_k} = \frac{\sum_{i=1}^{m-k}(x_i - \bar{x})(x_{i+k} - \bar{x})}{\sum_{i=1}^{m}(x_i - \bar{x})^2}.$$

where $l < m$ is the maximum lag to consider. The quantity $\hat{\rho_k}$ is called the sample autocorrelation coefficient at lag k and has range $[-1, 1]$. Our basic premise is that for a subset of TSC problems, the within class similarity is best detected from the ACF or some function of the ACF, rather than from the time domain data.

One feature of the ACF is that the terms are not independent. A low order autocorrelation will persist in higher order terms. The standard way for compensating for this is to find the Partial Autocorrelation Function (PACF) λ, where λ_i is the autocorrelation between terms X_i and X_{i+k} with the effect of all the intermediate variables $X_{i+1}, X_{i+2}, \ldots, X_{i+k-1}$ removed. The PACF is derived from the Yule Walker equations and usually calculated using the Durbin-Levinson recursions. For details see any standard Time Series book such as Box, Jenkins, and Reinsel (2008). The traditional use of the ACF and the PACF is in fitting an Autoregressive Moving Average (ARMA) model to the data. An ARMA model is of the form

$$X_t = c + \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

where c is a constant, and ϕ_i and θ_i are parameters which satisfy the invertibility and stationarity conditions (which require all solutions of the characteristic equations be in the range -1 and 1). The random variables ε_t are assumed to be independent, identically and normally distributed with zero

mean and constant variance. For TSC we assume that each series within the same class is a series of m observations from a common underlying ARMA model. Given an observed time series $\mathbf{x}$, the standard statistical methodology for fitting a model involves the following steps:

1. estimate the autocorrelation function (ACF) $\mathbf{r}$ from the data $\mathbf{x}$;
2. estimate the partial autocorrelations (PACF) λ from the ACF $\mathbf{r}$ using the Durbin-Levinson recursions;
3. choose the appropriate model that minimizes the Akaike Information Criterion (AIC).

Details on these steps can be found in Durbin (1960). Whilst clearly an appropriate approach if an ARMA model does in fact represent class similarity, there are two potential problems with these transforms. Firstly, whilst the ACF can be found in $O(nlog(n))$, finding the PACF is $O(n^2)$. Faster worst case $O(n\log^2(n))$ algorithms do exist (Bojanczyk, Brent, and De Hoog 1995) but they are complex and are on average not much faster than the standard approach. Secondly, if new data is observed, there is no easy way to update the PACF and the whole procedure must be repeated. Our objective is to find fast approximations of the fitted model that can be used to discriminate between different underlying generating models when there are time constraints on the process.

We have written basic filters in the machine learning Java toolkit WEKA (Witten et al.1999) to perform these transformations. All code is available for download from:

http://www.uea.ac.uk/computing/machine-learning/runlengths.

3. Run Length Transform

A run in a time series is a contiguous sequence of observations that are all either above or below the mean. The number of runs in a series is commonly used as a test of independence. The run length distribution is a sequence of counts of the number occurrences of runs of length $(1, 2, \ldots, s)$ for some predefined s. Figure 1 describes the run length transform and Figure 2 shows a small example.

Two further refinements are used. For comparing series of unequal length, we convert the run length distribution into a run length proportion, by simply dividing each data by the run sum. Secondly, for classification problems, we can remove the need for the parameter s by setting s equal to m, then removing the attributes for which all values are zero.

In Bagnall and Janacek (2005) we proposed using the clipping transformation for time series data mining. Clipping is simply a transformation

Input: A time series $\mathbf{x}$ of length m, the series mean μ and the max run length s.
Output: A run length histogram **r**.

Initialise **r** to all zeroes.
$l = 1$
for $i = 2$ **to** m **do**
  if $(x_{i-1} < \mu \wedge x_i < \mu) \vee (x_{i-1} > \mu \wedge x_i > \mu)$ **then**
    $l = l + 1$
  else
    if $l < s$ **then**
      $r_l = r_l + 1$
    else
      $r_s = r_s + 1$
    end if
    l=1
  end if
end for
return r

Figure 1. The run length transform algorithm.

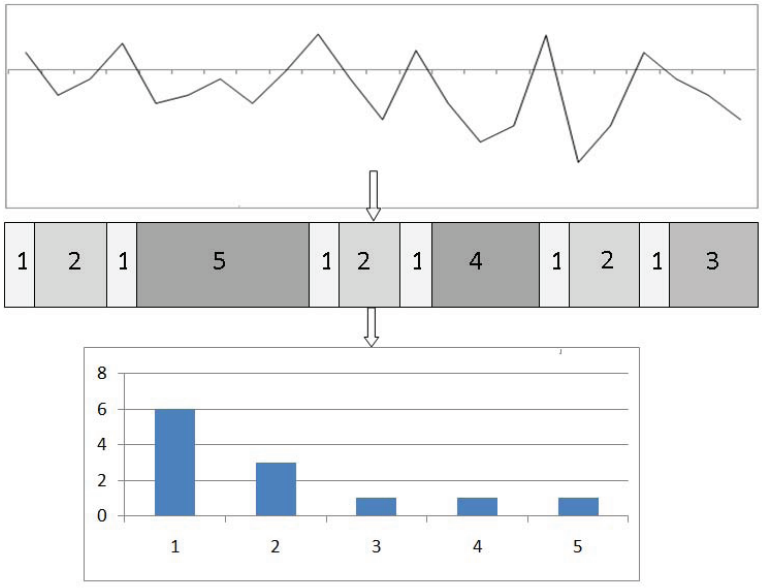

Figure 2. A small example of the run length transform.

of a real valued series into a binary series of above and below the average. Clipping the series allows for the data to be massively compressed and also offers the potential to use very fast bit operators. We showed that AR models derived from the compact clipped representation are theoretically related to models derived from the original data and that the discriminatory power was thus asymptotically equivalent. In Bagnall, Ratanamahatana, Deogh, Sonardi, and Janacek (2006) we showed that as a natural consequence of using clipped data we could generate run lengths, and that if we used a variable length run length encoding we could utilize the Kolomgorov compression to cluster shaped based series. This form of similarity will detect some forms of variation in the run length distributions, but it is not discriminatory enough to detect small differences between the generating model of series caused by differences in autocorrelation function. Hence we were lead to consider the relationship between the run length distribution and the autocorrelation function.

The probability of crossing the zero level by a series is called the *zero crossing probability* and is obviously related to the run lengths distribution. Suppose that D is the zero crossing count of a discrete time series of length m. If the original data has zero mean, D is simply the sum of the run length transform, $D = \sum_{i=1}^{s} r_i - 1$, (where r_i is the count of number of runs of length i), since each run representing a zero crossing. It can be shown (see Kedem 2008)

$$\lim_{m\to\infty} \frac{D}{m} = \frac{1}{\pi}\cos^{-1}(\rho(1)). \tag{1}$$

Where $\rho(1)$ is population autocorrelation of lag 1. This result shows there is a direct relationship between the distribution of the number of zero crossings and the autocorrelation lag $\rho(1)$ and hence if we measure the number of zero crossings we can, asymptotically at least, make observations about the variability of the autocorrelations of lag 1. This suggests that the more detailed information in the run length transform may inform us about the variability of the higher order autocorrelations. If we assume a series $\mathbf{x}$ is a set of observations from normally distributed random variables $\{X_t\}$, the probability of a zero crossing is

$$P[(X_{t-1} < 0 \textbf{ and } X_t > 0) \textbf{ or } (X_{t-1} > 0 \textbf{ and } X_t < 0)]$$

and

$$P[X_{t-1} < 0 \textbf{ and } X_t > 0] = \int_{-\infty}^{0}\int_{0}^{\infty} f(x_{t-1}, x_t)dx_{t-1}d_{x_t}$$

where f is the bivariate normal distribution. To compute the probability of a run of length i we need to find the probability

$$p[X_t < 0, X_{t+1} > 0, X_{t+2} > 0, \cdots, X_{t+i} > 0, X_{t+i+1} < 0]$$

and

$$p[X_t > 0, X_{t+1} < 0, X_{t+2} < 0, \cdots, X_{t+i} < 0, X_{t+i+1} > 0].$$

We can write these probabilities as $i+2$ fold integrals of multivariate normal distributions. Our evaluations of these probabilities thus depends on the evaluation of integrals of the form

$$\frac{1}{\sqrt{|\mathbf{\Sigma}|(2\pi)^i}} \int_\infty^0 \int_0^\infty \cdots \int_0^\infty \int_\infty^0 exp\left(-\frac{1}{2}\boldsymbol{x}^T \mathbf{\Sigma}^{-1} \boldsymbol{x}\right) d\boldsymbol{x}$$

where $\boldsymbol{x} = (x_t, x_{t+2}, \cdots, x_{t+i+1})^T$ and $\mathbf{\Sigma}$ is the covariance matrix constructed from the first $i+2$ autocorrelations. So although it is computationally impractical, given the autocorrelation function ρ we could in theory calculate the probability of observing a run of any length i. It thus follows that variability in the ACF expressed in the observed series will also express itself in the run length distribution, and hence, with long enough series, we should be able to discriminate between different series using the runs distribution alone.

4. Run Length Similarity

The majority of research into AR based methods uses Euclidean distance between fitted model parameters as a similarity metric (see, for example, Corduas and Piccolo 2008), and for the full fitted models we also adopt this approach. However, there are two basic potential problems with using Euclidean distance as a metric between two run length distributions. Firstly, Euclidean distance gives a much higher weighting proportionately to small differences in runs with large counts over runs with small counts. This may mean that subtle differences are overwhelmed by differences caused by random fluctuation. We overcome this problem by defining alternative pointwise distance measures. The second problem is that pointwise similarity ignores the fact that there is embedded adjacency information in the distribution; for example, a source series with a large count of runs length 2 is more similar to another series with a large run count of 3 than it is with one with a large run count of 30. To utilize this information we use a dynamic time warping between run length distributions as a distance measure.

Suppose we have two run length distributions, $\mathbf{a} = \{a_1, a_2, \ldots, a_s\}$ and $\mathbf{b} = \{b_1, b_2, \ldots, b_s\}$. A distance function between $\mathbf{a}$ and $\mathbf{b}$, denoted $d(\mathbf{a}, \mathbf{b})$, is effectively a test of whether $\mathbf{a}$ and $\mathbf{b}$ are drawn from the same population distribution. The obvious way to perform this test that does not over-weight to large values is to use the χ^2 statistic. However, since χ^2 is based on the ratio of observed and expected values, it has the opposite

problem to Euclidean distance: it can give excessive weight to small observations. It is common practice to merge observations with low counts (typically with expected value less than 5) to mitigate against this effect. This is computationally complex, and requires some subjective decisions.

Suppose, for example, we are comparing the distributions $\mathbf{a} = \{14, 0, 7, 3, 2, 1, 0, 0\}$ with $\mathbf{b} = \{7, 7, 7, 6, 0, 0, 0, 0\}$ (note for simplicity we are merging on observed values not expected, but the point still holds). Histogram $\mathbf{a}$ dips at a_2 and tails off, whereas $\mathbf{b}$ is flat and ends abruptly at b_4. To effectively use χ^2 we should merge elements with value less than 5. If we merge by tracking backwards from the end of the distribution we get $\mathbf{a} = \{14, 0, 7, 6, 0, 0, 0, 0\}$, but we are still left with the zero observation at a_2. Should this be merged with a_1 or a_3? Suppose we choose (arbitrarily) to merge it with a_1. Do we perform the same transform on $\mathbf{b}$? If so, we end up with the two identical series $\mathbf{a}' = \{14, 7, 6, 0, 0, 0, 0, 0\}$ with $\mathbf{b}' = \{14, 7, 6, 0, 0, 0, 0, 0\}$. If we had chosen to merge a_2 with a_3, $\mathbf{b}$ becomes $\mathbf{b}' = \{7, 14, 6, 0, 0, 0, 0, 0\}$ and is now very different to $\mathbf{a}$. If we chose not to alter $\mathbf{b}$ it becomes difficult to compare the distributions since the indexes are different. To avoid these problems, we propose two alternative pointwise distance measures commonly used to compare counts data: the Gower coefficient and the Likelihood ratio statistic.

4.1 The Gower Coefficient

We derive a statistic to measure similarity that is based on "General Similarity Coefficient" (Gower 1971). The Gower coefficient is designed to measure the similarity between two units from a sample of observations. For real valued attributes it involves normalising data by the range.

Suppose we have a data set $\mathbf{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ of n time series, which we convert into n run length distributions $\mathbf{R} = \{\mathbf{r_1}, \ldots, \mathbf{r_n}\}$, each of which has s run length counts $\mathbf{r_i} = \{r_{i1}, \ldots, r_{is}\}$. So r_{ij} denotes the count of runs of length j in the i^{th} time series. Let v_k denote the range of values for the k^{th} attribute, i.e.

$$v_k = \max_{i=1,\ldots,n}(r_{ik}) - \min_{i=1,\ldots,n}(r_{ik})$$

The Gower similarity coefficient for two series $\mathbf{r_i}$ and $\mathbf{r_j}$ is then

$$g(\mathbf{r_i}, \mathbf{r_j}) = \sum_k 1 - \frac{(r_{ik} - r_{jk})}{v_k}$$

Note most definitions include an attribute weighting term, which we ignore for simplicity. g is a measure of similarity. We redefine it for distance

as

$$d_g(\mathbf{r_i}, \mathbf{r_j}) = \sum_k \frac{(r_{ik} - r_{jk})}{r_k}$$

We refer to d_g as the Gower coefficient. Differences in variables with small ranges contribute more to d_g than those with large ranges. (See also Cox and Cox 2000). Whilst this mitigates against the problems with using Euclidean distance, it does require a set of observations. With many applications such as anomaly detection distances are calculated between subsequences of the same series and there is no fixed set of observations. Hence we define a second pointwise distance measure based on the likelihood ratio that can measure the distance between any two series.

4.2 Likelihood Ratio Distance

The likelihood ratio test is a commonly used alternative to the χ^2 test of distributions. Hence we can adapt the likelihood ratio test statistic for multinomial distributions to form a distance measure between histograms. For any two series $\mathbf{a} =< a_1, \ldots, a_s >, \mathbf{b} =< b_1, \ldots, b_s >$, let $m_a = \sum_{i=1}^{s} a_i$, $m_b = \sum_{i=1}^{s} b_i$. The observed frequency for each index i in $\mathbf{a}$ and $\mathbf{b}$ is then

$$p_{a_i} = \frac{a_i}{m_a}, \frac{b_i}{m_b}$$

and the overall observed frequency is

$$p_i = \frac{a_i + b_i}{m_a + m_b}.$$

The likelihood ratio test statistic, which we denote d_l as we are using it as a distance measure, is

$$d_l(\mathbf{a}, \mathbf{b}) = \sum p_{a_i} \log\left(\frac{p_{a_i}}{p_i}\right) + \sum p_{b_i} \log\left(\frac{p_{b_i}}{p_i}\right)$$

Unlike the Gower coefficient, the log likelihood does not require ranges over a set of series. It avoids the problem of zero counts, since they do not contribute to the sum. Neither statistic allow for the adjacency relationship present in runs histograms. To utilize this we employ a dynamic time warping measure.

4.3 Dynamic Time Warping

For similarity in shape, Dynamic Time Warping (DTW) is commonly used to mitigate against distortions in the time axis (Ratanamahatana and

Keogh 2005). Suppose we want to measure the distance between two run length distributions, $\mathbf{a} = \{a_1, a_2, \ldots, a_s\}$ and $\mathbf{b} = \{b_1, b_2, \ldots, b_s\}$. Let $M(\mathbf{a}, \mathbf{b})$ be the $m \times m$ pointwise distance matrix between $\mathbf{a}$ and $\mathbf{b}$, where $M_{i,j} = (a_i - b_j)^2$.

A warping path $W =< (e_1, f_1), (e_2, f_2), \ldots, (e_k, f_k) >$ is a set of points (i.e. pairs of indexes) that define a traversal of matrix M. So, for example, the Euclidean distance $d_E(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{k}(a_i - b_i)^2$ is the path along the diagonal of M, i.e.$W_e =< (1, 1, (2, 2), \ldots, (s, s) >$.

A valid warping path must satisfy the conditions $(e_1, f_1) = (1, 1)$ and $(e_k, f_k) = (m, m)$ and that $0 \leq e_{i+1} - e_i \leq 1$ and $0 \leq f_{i+1} - f_i \leq 1$ for all $i < m$.

The DTW distance between series is the path through M that minimizes the total distance, subject to constraints on the amount of warping allowed. Let $w_i = M_{a_{e_i}, b_{f_i}}$ be the distance between element at position e_i of $\mathbf{a}$ and at position f_i of $\mathbf{b}$ for the i^{th} pair of points in a proposed warping path P. The distance for any path P is

$$D_P(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{k} w_i.$$

If $\mathcal{W}$ is the space of all possible paths, the DTW path W is the path that has the minimum distance, i.e.

$$W = \min_{P \in \mathcal{W}}(D_P(\mathbf{a}, \mathbf{b})),$$

and hence the DTW distance between series is

$$D_W(\mathbf{a}, \mathbf{b}) = \sum_{i=1}^{k} w_i,$$

The optimal path W can be found exactly through a dynamic programming formulation. This can be a time consuming operation, and it is common to put a restriction on the amount of warping allowed. This restriction is equivalent to putting a maximum allowable distance between any pairs of indexes in a proposed path. If the warping window is h, then the optimal path is constrained so that $|e_i - f_i| \leq h \ \forall (e_i, f_i) \in W$. Since our histograms are short and we wish to set a small warping window (because we are only interested in compensating for small misalignments in the histograms), the DTW distance will not introduce much of a time overhead.

5. Experimental Setup

Our interest lies in discriminating between series emanating from autoregressive models. We are not claiming that the run lengths approach will

be necessarily useful for other sorts of data (in fact, we demonstrate it is not suitable for many problems in Section 6.5). We further restrict our attention to weakly stationary series (series with constant mean and variance). This is because our focus is on long series, and discriminating between long non-stationary series from different models will be best achieved in the time domain. If the class models are non-stationary, then the difference between instances of one class and another will be dominated by the trend rather than the the short term fluctuations that characterize stationary series.

The basic hypotheses we are testing is whether one transformation/ distance measure is better than another for a given class of generating data. Our definition of whether one algorithm is better than another is that it has a higher average classification accuracy over the population of all stationary AR models considered for a particular classification experimental set up. The set up we use for all classification experiments involves two randomly generating models representing two classes.

To compare two algorithms for any fixed series length m we randomly generate k model pairs using the algorithm described below. For each of these pairs we generate n train/test data sets and record the testing accuracy. We then average the accuracy over the test data sets for a particular model and over all the models used for that particular value of m. We repeat this for increasing values of m to demonstrate the asymptotic behaviour. Our bench mark performance is always the fitted AR model which, given the data is from an AR process, should on average be optimal, all other parameters being equal.

Generating random stationary models for classification is not particularly easy. We need to control two factors: firstly, the models must be stationary; and secondly, they must be both distinguishable from each other with enough data and non-trivial for smaller data set sizes. Given a set of random numbers between -1 and 1, it is possible to generate the stationary AR parameters by solving the system of linear equations that result from the stationary restriction on the the characteristic equation

$$1 - \sum_{i=1}^{p} \phi_i r^p = 0$$

using any standard algorithm such as Gaussian elimination via LU decomposition. The resulting solution gives the parameters of a stationary AR model of order p. The algo

1. For model 1.
    (a) Randomly choose a number of parameters p_1 between 1 and 10.
    (b) Generate p_1 random numbers between -0.9 and 0.9, $\mathbf{r_1} = \{r_1, \ldots, r_{p_1}\}$.

2. For model 2.
   (a) Perturb p_1 by a random number between -2 and 2 (inclusive) to find p_2.
   (b) Perturb the first $\min(p_1, p_2)$ terms of $\mathbf{r_1}$ by between -10% and 10% to obtain $\mathbf{r_2}$.
   (c) If $p_2 > p_1$ generate $p_2 - p_1$ random parameters and add to $\mathbf{r_2}$.
3. Solve the systems of linear equations to find the two sets of valid (i.e. stationary) AR parameters $\phi_\mathbf{1}$ and $\phi_\mathbf{2}$.
4. Generate train/test data sets of series length $m = 400$ and $m = 1000$.
5. Perform a 1-NN test classification on the fitted AR parameters on both small and large data sets.
6. If test accuracy on the smaller $m = 400$ data set is less than 60% or greater than 90% reject the model.
7. If test accuracy on the larger $m = 1000$ data set is 100% or less than that on the smaller set, reject the model.

The code for doing this is available from:
http://www.uea.ac.uk/computing/machine-learning/runlengths.

6. Results

The experimental series is designed to highlight the merits of the run length transformation. We demonstrate that as the length of the series increases the discriminatory power of the run length distribution approaches that of the full AR model (Section 6.1), then evaluate the alternative distance measures for run length distributions and conclude that dynamic time warping performs the best of those considered (Section 6.2), and that it outperforms the ACF and FFT (Section 6.3). The real benefit of using the run length is the speed of performing the transform, hence in Section 6.4 we show that if the amount of time to construct the model is constrained (meaning less data can be used to fit the full model) then the run length transform can produce more accurate classifiers than those found from the full model. For completeness, in Section 6.5 we demonstrate that for shape based similarity the AR and run length approach is not as effective as previously published alternatives. Finally, we conduct a case study on data relating to heart rates whilst meditating where the approach is appropriate.

6.1 Asymptotic Equivalence Between Run Lengths and the Full AR Model

The main result of this paper is that as series get longer, it is not necessary to fit the full AR model to data in order to discriminate between alterna-

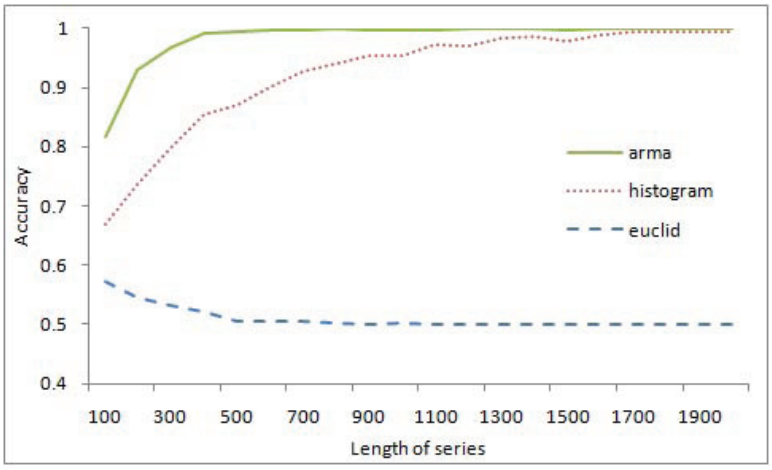

Figure 3. Testing classification accuracy for increasing m using Euclidean distance with a 1-Nearest Neighbour classifier on the raw data, the fitted ARMA parameters and the run length histogram. The two class generating models are both AR(1) with parameters $\phi = 0.5$ and $\phi = 0.7$ respectively. Each point is the average testing accuracy of ten runs, with a training set and testing set of 100 series of each class.

tive generating models. Figure 3 demonstrates this with a simple simulated classification problem. As the size of the series increases, the classification accuracy using the run lengths distribution approaches the accuracy of using the fitted AR parameters.

In Figure 3 we include Euclidean distance on the raw data to highlight that the type of series we are considering is not distinguishable by the commonly used distance measures used in the time series data mining literature. We do this to highlight the fact that we are only considering a certain class of problem.

6.2 Alternative Distance Measures for Run Lengths

This experiment assesses the four proposed distance metrics to use with run length distributions and compares them to results achieved using the full fitted AR model. Given the fact that the simulated data is generated by an AR model, discriminating with the estimated AR model fitted on the whole data set is clearly the optimal approach. Hence we wish to assess which of the four metrics best approximates the full model. The maximum run length stored was set to 10% of original data length. So for series length 1000, all runs over 100 were recorded as the same length. The maximum warping window was set to 10% of the size of the maximum run length.

Table 1. Average testing classification accuracy on test data sets for two class AR problems using a 1 Nearest Neighbour classifier and five different distance metrics. Results are for full AR model (AR+Euclidean) and with the run lengths distribution (RL+measure) with four alternative distance metrics. Each cell contains the mean (standard deviation) over 200 randomly generated models (100 models for $m = 5000$).

m	AR+Euclidean	RL + Euclidean	RL + Gower	RL + Likelihood	RL + DTW
200	57.24% (4.83)	52.27%(4.80)	52.68% (5.3)	52.70% (4.833)	54.00% (4.88)
1000	76.20% (9.56)	53.63% (4.88)	58.79% (7.98)	57.93% (7.21)	61.67% (9.23)
5000	78.18% (17.90)	59.86% (3.98)	70.30% (18.64)	68.76% (6.78)	77.84% (12.03)

Note these model parameters have been in no way optimized and it's highly likely we could achieve similar results with greater compression.

In Table 1, to compare two sets of results for any given m, we perform both a paired t test and a Kruskal-Wallis paired test. The first observation to make concerning these results is that for each of the three levels of m, there is a significant difference in mean accuracy between DTW distance and the Euclidean, Gower and Likelihood distances. Secondly, Gower and Likelihood are significantly better than RL+Euclidean, but not significantly different to each other. Finally, for $m = 200$ and $m = 1000$ AR+Euclidean is significantly more accurate than DTW. However, for $m = 5000$, there is no significant difference between AR+Euclidean and RL+DTW. This reinforces the theoretical result from Section 2 and the experimental result from Section 6.1 that there is a strong asymptotic relationship between the ACF and the runs length and that variations in underlying AR model express themselves more fully in the run lengths as m gets larger.

6.3 Alternative Transformations

In Section 6.1 we demonstrated that for large enough m dynamic time warping of the run length histogram gives an accuracy that is not significantly worse than that obtained using Euclidean distance on the fitted AR parameters. However, the obvious next question is how does the RL transform perform in comparison to other possible transformations. There are two popular transformations used in the data mining and machine learning community for classification/clustering data based on autoregressive structure. The first involves taking the Fourier transform and retaining a subset of the coefficients of Agrawal, Faloutsos, and Swami (1993), and Wu, Agrawal, and El Abbadi (2000). The Fourier terms are particularly useful because they can be used to differentiate between both shape and structure, depending on the distance measure used. The other approach often seen in the Machine Learning literature is to find the ACF and simply use it as a set of new features. It should be noted that although faster than finding the full

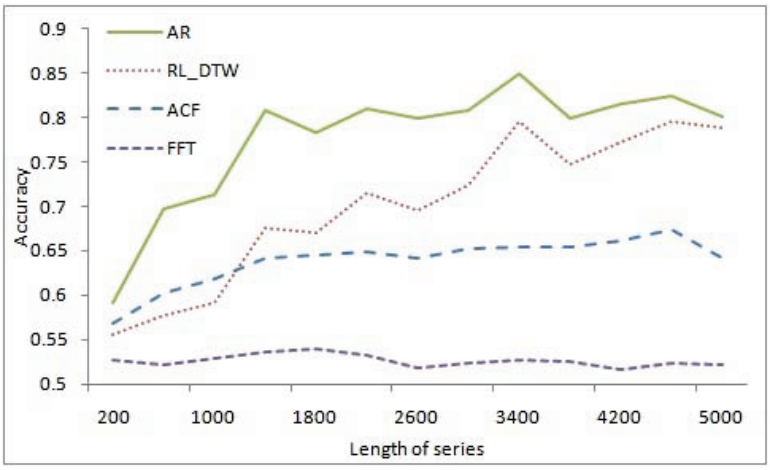

Figure 4. Testing classification accuracy for increasing m using 1-NN.

fitted model through the PACF, neither of these approaches is linear in m. Furthermore, neither can be updated in constant time. To assess whether the simpler and faster histogram approach compares to these alternatives, we ran classification experiments for varying values of m. The average testing accuracy is shown in Figure 4.

Firstly Figure 4 shows us that once the series are longer than approximately 1000, the RL transformation performs better than using the ACF. Secondly, whilst the RL is clearly tending towards the AR accuracy, the gap between AR and ACF is maintained (the ACF accuracy is approximately 80% of the ARMA accuracy for all sizes of m). This demonstrates that there are fundamental differences between the series that are not detectable in the ACF but which can be detected in the RL distribution, given enough data. Thirdly, the FFT approach is clearly not able to differentiate between the series. We include these results to demonstrate the importance of clarifying whether shape based or model based similarity is required. We used the standard Euclidean distance measure on the FFT, which is really just an approximation of Euclidean distance, and hence inappropriate. We also experimented with using DTW on the AR coefficients, the periodogram of the FFT and on the ACF, but the results were not significantly different to those obtained without warping. Table 2 presents the data from Figure 4 alongside the associated warping results.

6.4 Classification with Limited Time

The transformation we propose is appropriate when predictions are needed quickly, new data is constantly arriving or the data has already been

Table 2. 1-NN classification accuracy with both Euclidean and DTW distance.

m	AR(Euclid)	AR(DTW)	RL(Euclid)	RL(DTW)	FFT(Euclid)	FFT(DTW)	ACF(Euclid)	ACF(DTW)
200	59.15%	58.30%	53.88%	55.63%	52.65%	53.75%	56.90%	54.68%
600	69.73%	68.78%	53.20%	57.75%	52.13%	51.95%	60.30%	56.93%
1000	71.30%	71.70%	53.80%	59.18%	52.93%	55.08%	61.75%	53.05%
1400	80.77%	81.35%	58.35%	67.53%	53.50%	52.23%	64.10%	59.02%
1800	78.40%	77.87%	57.50%	66.98%	53.88%	51.25%	64.50%	55.83%
2200	81.07%	81.05%	57.53%	71.55%	53.18%	52.72%	64.93%	57.10%
2600	79.93%	80.25%	56.65%	69.60%	51.72%	50.93%	64.13%	55.82%
3000	80.88%	80.68%	57.15%	72.40%	52.40%	51.05%	65.27%	56.40%
3400	84.95%	85.47%	60.65%	79.57%	52.67%	52.25%	65.38%	60.15%
3800	79.93%	79.78%	58.42%	74.75%	52.58%	52.28%	65.40%	58.08%
4200	81.62%	81.85%	58.93%	77.30%	51.68%	52.97%	66.13%	60.32%
4600	82.45%	82.58%	61.90%	79.55%	52.40%	52.35%	67.40%	59.70%

transformed into clipped data. To simulate the scenario of limited time we examine what happens to the classification accuracy when we allow the RL method access to more data than the AR or the ACF transform. We compare DTW with run lengths against logistic regression for AR and ACF (in further experimentation not reported here, logistic regression was found to outperform 1-NN for AR and ACF). We fix the series length for AR and ACF to 500 but increase the length of series available to RL(DTW) and measure the associated testing accuracy. Figure 5 shows that RL outperforms the other two methods with approximately 3000 data.

Table 3 shows what happens when we vary the amount of data for AR and ACF. The second column gives the size of m at which RL(DTW) first outperforms both AR and ACF transforms. The third column shows that the multiple of data required by RL(DTW) is constant or possibly decreasing. Table 4 gives some timing results for increasing m and demonstrates that the RL transform is an order of magnitude faster than the other approaches.

The final columns in Tables 3 and 4 clearly demonstrate as m increases, the time taken to transform is increasing but the proportion of extra data required by RL is constant. Hence, if time is an issue and the time series are long, then better results will be achieved by using the run length transform for classification.

6.5 UCR Datasets

In Figure 3 we include Euclidean distance on the raw data to highlight that the type of series we are considering is not distinguishable by the commonly used distance measures used in the time series data mining literature. It is worth making the point that the converse is also true. ARMA models do not work well with shape based problems. Table 5 gives the classification accuracy for a selection of problems from the Time Series Data Mining

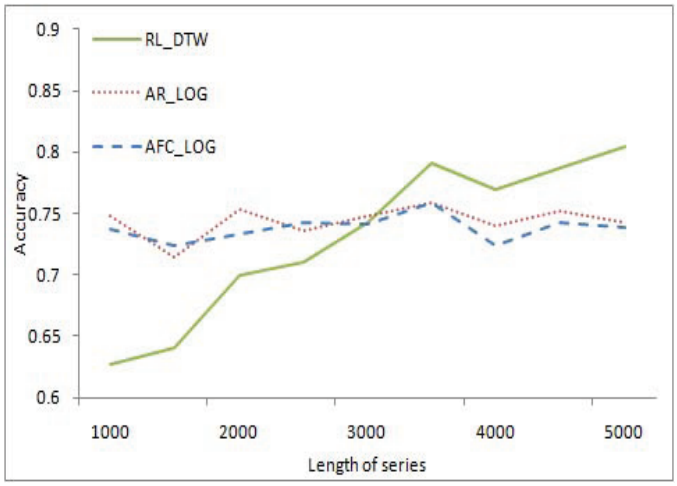

Figure 5. Accuracy of RL(DTW) for increasing m vs AR and ACF using logistic regression with $m = 500$.

Archive (Keogh and Folias). Given the run length distribution is designed for long series, we selected the twelve problems with the longest series, ranging in length from 500 to 2709. The classifier built in the time domain outperforms the auto regressive and the run length classifiers. We include these results to emphasize that we are considering only long series that can be well fitted by a stationary auto-regressive model, and that the techniques we propose do not necessarily generalize to all time series problems. In the following section we identify a problem where the approach is appropriate.

7. Meditation

Our final experiment is a real world case study which we include to highlight the advantages of using the run length distribution and to reinforce the conclusions drawn on simulated data. The data comes from the paper *"Exaggerated Heart Rate Oscillations During Two Meditation Techniques"* (Peng et al.1999) and was downloaded from PhysioNet. The study involved taking time series of measurements of heart rates approximately every second from a set of subjects with the objective of finding differences in the change in heart rate between normal subjects and those meditating using either Chi or Kundalini techniques. There were four groups of subject:

- *Chi meditation group*. There are two time series for each of the eight subjects, one before meditation, one during meditation. Each series is about one hour in duration.

Table 3. Size of data set (to the nearest 100) for RL(DTW) to outperform both AR(logistic) and ACF(logistic).

Fixed m for AR and ACF	m when RL(DTW) first wins	Estimated Ratio
100	300	3
200	1300	7
300	1500	5
400	2000	5
500	2800	6
600	2400	4
700	3200	5
800	3000	4
900	3700	4

Table 4. Time taken to transform 200 data sets of length m, averaged over 30 runs.

m	RL(DTW)	AR	ACF	Estimated Ratio (ACF/RL)
500	4.006	43.254	38.06	10
1000	9.328	164.94	146.75	16
1500	17.90	370.47	326.10	18
2000	28.74	657.33	575.050	20
2500	42.48	1043.32	899.593	21
3000	60.37	1545.31	1302.45	22
3500	77.08	2120.25	1746.04	23
4000	93.84	2756.55	2287.04	24
4500	119.52	3517.67	2899.59	24
5000	136.87	4418.22	3620.76	26

- *Kundalini Yoga meditation group*. As for the Chi group, there are pre-meditation and meditation series for each of the four subjects. Durations range from 17 to 47 minutes.
- *Spontaneous breathing group.* Eleven volunteers were recorded while sleeping. Durations are 6 hours each.
- *Metronomic breathing group.* Fourteen Volunteers were recorded while supine and breathing at a fixed rate of 0.25 Hz for 10 minutes.
- *Elite athletes.* Nine subjects participated in the Ironman Triathlon; the recordings were obtained during sleeping hours before the event. Durations range from 1 to 1.7 hours.

From a spectral analysis they observed that the amplitude of oscillations during meditation was significantly greater than in the non meditation group, and that this large variation contradicts the commonly held notion that meditation results in a quiescent state. In later work this has been termed the "meditation paradox" (Qairunnisa et al. 2012). The initial study focussed on the variation in amplitude of heart rates. The series are long, and vary greatly in length (the series range in length from 469 to 26,923), with readings taken approximately every second. To demonstrate the utility of the

Table 5. 1-NN classification accuracy of on test data sets from the long series UCR data sets.

Data Set	Raw Data+Euclidean	AR+Euclidean	Run Lengths+Euclidean	Run Lengths+DTW
FordA	68.18%	73.79%	76.52%	63.71%
FordB	59.14%	60.12%	63.21%	53.58%
Earthquakes	71.22%	71.94%	69.06%	69.06%
OliveOil	83.33%	40.00%	43.33%	40.00%
Lighting2	80.33%	55.74%	59.02%	47.54%
MALLAT	82.09%	61.45%	53.48%	34.07%
StarLightCurves	86.50%	81.85%	54.27%	76.32%
Haptics	39.29%	24.03%	21.10%	20.13%
CinCECGtorso	86.38%	40.07%	31.16%	46.38%
InlineSkate	31.45%	18.91%	16.91%	17.64%
HandOutlines	85.95%	55.41%	50.27%	66.76%

Table 6. Leave one out cross validation error and balanced error for 1-NN classifier with four distance measures on the meditation data set.

Distance Measure	Error	Balanced Error
Euclidean	10.34%	15.76%
Gower	13.79%	21.01%
Likelihood Ratio	8.62%	14.67%
DTW	8.62%	11.59%

Table 7. Contingency table for LR and DTW distance on the meditation data

	Likelihood Ratio Predicted		DTW Predicted	
Actual	Meditating	Not Meditating	Meditating	Not Meditating
Meditating	9	3	10	2
Not Meditating	2	44	3	43

run lengths transform, we frame the problem as time series classification. We label the 12 series of subjects performing Chi or Kundalini meditation as “meditating”, and the 46 series measured under other conditions as “not meditating”. We transform each series into the run length distribution, then convert to the run lengths proportion to compensate for the different lengths of the series. Table 6 shows the error rate and balanced error rate of a leave one out cross validation using a 1-NN classifier on the histogram distribution using four distance measures: Euclidean, Dynamic Time Warping, Gower and Likelihood ratio.

Table 7 shows the contingency tables for the likelihood ratio and DTW results. DTW performs best, misclassifying just two of the meditation series and three non meditation series. The two misclassified meditation cases were from the shorter Kundalini group. Although the data set is small and hence we should be careful on drawing too many conclusions,

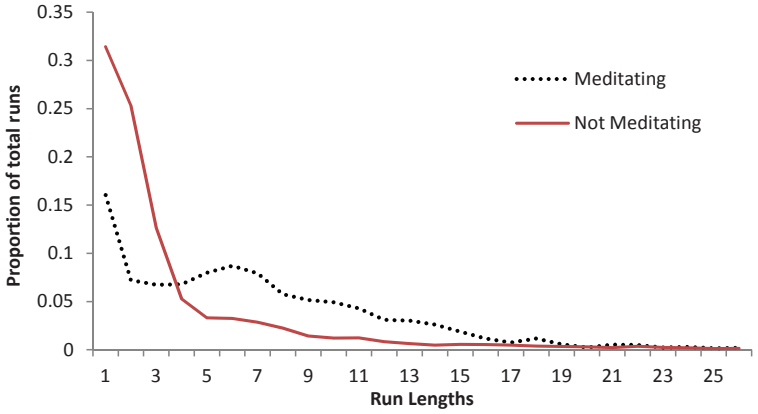

Figure 6. The average distribution of run lengths for meditating and non meditating subjects.

these results reinforce the conclusions drawn from simulated data that runs distributions can be used to detect differences in autocorrelation structure and that DTW is the best distance measure to use for this. Run lengths also offer a very simple way to analyse the data that can offer new insights. Figure 6 shows the average run length distributions for meditating and non-meditating subjects. Proportionately, meditating subjects have fewer runs of length 3 or less seconds, and more in the range 4 to 19 seconds. Hence the larger oscillation in heart rate values for meditators are apparently coupled with longer runs of above or below the mean. This may be of interest to domain experts since it may lead to insights into the connection between breathing rates and heart rate oscillations.

8. Conclusions

This paper has addressed the issue of how to classify data sets where each case is a long time series that can be accurately modelled by stationary autoregressive (AR) series. We propose using the run length distribution to discriminate between data from different stationary AR models. We show the theoretical link between run lengths and the autocorrelation function (ACF) and propose four possible distance metrics for comparing two run length histograms: Euclidean distance, the Gower statistic, likelihood ratio and dynamic time warping. We demonstrate through classification experiments with a 1 nearest neighbour classifier that dynamic time warping has the greatest discriminatory power. We then compare the run length transform (RL) with using the AR parameters fitted with the Durbin-Levinson recursions, the Fourier transform (FFT) and the ACF. We show that, when

using 1-NN, the accuracy gap between RL and AR/ACF decreases as m increases and eventually becomes insignificant. We then experiment further with alternative classifiers and find that ACF transform actually outperforms the AR transform, which is somewhat surprising. Finally we demonstrate the real benefits of using RL through a time controlled experiment. RL requires approximately 4 times as much data to perform as well as ACF and AR with the most effective classifier and this is at worst constant and possibly decreasing as m increases, whereas the relative time taken to fit ACF and AR is increasing with m.

This type of model is commonly used to model financial and engineering data. We have used classification experiments to test our algorithms, but a more natural application would be for detecting model change. Suppose you have constant readings from a sensor attached to a critical machine where variation is caused by internal noise and feedback and that the readings from such a machine could be accurately modelled by a stationary AR series. If the nature of this variation changes (for example, a spanner is thrown into the works) it may be of critical importance to intervene quickly. If one were to use a fitted AR model to detect a change of model based on a comparison of all historical data against recent observations it would require fitting models to the entire data and the new data. If run lengths are employed the previous fitted model can be quickly updated for the new data.

References

ABRAHAM, Z., and TAN, P. (2010), "An Integrated Framework for Simultaneous Classification and Regression of Time-Series Data", in *Proceedings of the SIAM International Conference on Data Mining, SDM 2010, Columbus, OH*, available from http://www.siam.org/ proceedings/datamining/2010/dm10_157_ abrahamz.pdf.

AGRAWAL, R., FALOUTSOS, C., and SWAMI, A. (1993), "Efficient Similarity Search in Sequence Databases", in *Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms, Chicago*, also in *Lecture Notes in Computer Science 730*, Springer Verlag, pp. 169–184.

BAGNALL, A., and JANACEK, G. (2004), "Clustering Time Series from ARMA Models with Clipped Data", *Machine Learning, 58(2)*, 151–178.

BAGNALL, A., RATANAMAHATANA, C., KEOGH, E., LONARDI, S., and JANACEK, G. (2006), "A Bit Level Representation for Time Series Data Mining with Shape Based Similarity", *Data Mining and Knowledge Discovery Journal, 13(1)*, 11–40.

BAGNALL, A., DAVIS, L., HILLS, J., and LINES, J. (2012), "Transformation Based Ensembles for Time Series Classification", in *Proceedings of the International Conference on Data Mining, SDM 2012, Anaheim CA*, available from http://siam.omnibooks online.com/2012datamining/.

BOJANCZYK, A.W., BRENT, R.P., and De HOOG, F.R., (1995), "A Weakly Stable algorithm for General Toeplitz Systems", *Numerical Algorithms, 10*, 225–244.

BOX, G., JENKINS, G., and REINSEL, G., (2008), *Time Series Analysis, Forecasting and Control"*, John Wiley and Sons.

BUZA, K., NANOPOULOS, A., and SCHMIDT-THIEME, L. (2011), "Fusion of Similarity Measures for Time Series Classification", in *Proceedings of the 6th International Conference on Hybrid Artificial Intelligent Systems*, also in *Lecture Notes in Computer Science 6679*, Springer Verlag, pp. 253–261.

CORDUAS, M., and PICCOLO, D. (2008), "Time Series Clustering and Classification by the Autoregressive Metric", *Computational Statistics and Data Analysis, 52(4)*, 1860–1872.

COX, T., and COX, M. (2000), "A General Weighted Two-Way Dissimilarity Coefficient", *Journal of Classification, 17(1)*, 101–121.

DENG, H., RUNGER, G., TUV, E., and VLADIMIR, M. (2013), "A Time Series Forest for Classification and Feature Extraction", *Information Sciences, 239*, 142–153.

DING, H., TRAJCEVSKI, G., SCHEUERMANN, P., WANG,X., and KEOGH, E. (2008), "Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measure", in *Proceedings of the 34th International Conference on Very Large Data Bases (VLDB)*, available from http://www.cs.ucr.edu/~eamonn/vldb_08 _Experimental_comparison_time_series.pdf.

DOUZAL-CHOUAKRIA, A., and AMBLARD, C. (2012), "Classification Trees for Time Series", *Pattern Recognition, 45(3),* 1076–1091.

DOUZAL-CHOUAKRIA, A., DIALLO, A., and GIROUD, F. (2010), "A Random-Periods Model for the Comparison of a Metrics Efficiency to Classify Cell-Cycle Expressed Genes", *Pattern Recognition Letters, 31(12)*, 1608–1617.

DURBIN, J. (1960), "The Fitting of Time Series Models", *Review of the International Statistical Institute, 28(4),* 233–243.

GOWER, J.C. (1971), "A General Coefficient of Similarity and Some of its Properties", *Biometrics, 27(4)*, 857–871.

HE, S., and KEDEM, B. (1990), "The Zero-Crossing Rate of Autoregressive Processes and Its Link to Unit Roots", *Journal of Time Series Analysis, 11(3)*, 201–213.

JEONG, Y., JEONG, M., and OMITAOMU, O. (2011), "Weighted Dynamic Time Warping for Time Series Classification", *Pattern Recognition, 44(9),* 2231–2240.

KALPAKIS, K., GADA, D., and PUTTANGUNTA, V. (2001), "Distance Measures for Effective Clustering of ARIMA Time-Series", in *IEEE Proceedings of the International Conference on Data Mining (ICDM), San Jose CA*, pp. 273–280.

KEOGH, E., and FOLIAS, T., "The UCR Time Series Data Mining Archive", http://www.cs.ucr.edu/eamonn/TSDMA/.

LIAO, T.W. (2005), "Clustering of Time Series Data: A Survey", *Pattern Recognition, 38(11)*, 1857–1874.

LINES, J., DAVIS, L., HILLS, J., and BAGNALL, A. (2012), "A Shapelet Transform for Time Series Classification", in *Proceedings of the 18th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Beijing, China*, available from http://dl.acm.org/citation.cfm?id=2339579.

MAHARAJ, E.A. (2000), Clusters of Time Series, *Journal of Classification, 17*, 297–314.

MAHARAJ, E.A. (1999), "Comparison and Classification of Stationary Multivariate Time Series", *Pattern Recognition, 32(7)*, 1129–1138.

MAHARAJ, E.A. (1996), "A Significance Test for Classifying ARMA Models", *Journal of Statistical Computation and Simulation, 54(4)*, 305–331.

PENG, C.K., MIETUS, J.E., LIU, Y., KHALSA, G., DOUGLAS, P.S., BENSON, H., and GOLDBERGER, A.L. (1999), "Exaggerated Heart Rate Oscillations During Two Meditation Techniques", *International Journal of Cardiology, 70(2)*, 101–107.

PHYSIONET, "PhysioBank's Data Archive", http://www.physionet.org/.

PICCOLO, D. (1990), "A Distance Measure for Classifying ARIMA Models", *Journal of Time Series Analysis, 11(2),* 153–164.

QAIRUNNISA, S., CHANDRASHEKAR, M., REVATHI, M., KONDAM, A., MADHURI, B.A., and SUESH, M. (2012), "A Study on Modulation on Cardiovascular Response to Yoga Training", *International Journal of Biological and Medical Research, 3(2)*, 1662–1666.

RATANAMAHATANA, C., and KEOGH, E. (2005), "Three Myths About Dynamic Time Warping Data Mining", in *Proceedings of the SIAM International Conference on Data Mining, DSM 2005*, available from www.siam.org/proceedings/datamining/2005/dm05_50ratanamahatanac.pdf.

RODRIGUEZ, J., ALONSO, C., and MAESTRO, J. (2005), "Support Vector Machines of Interval-Based Features for Time Series Classification", *Knowledge-Based Systems 18*, pp 171–178, available from http://www.sciencedirect.com/science/article/pii/S0950705105000432.

VAN WYK, B., VAN WKY, M., and QI, G. (2009), "Difference Histograms: A New Tool for Time Series Analysis Applied to Bearing Fault Diagnosis", *Pattern Recognition Letters, 30(6)*, 595–599.

WITTEN, I., FRANK, L., and HALL, M., (2011), *Data Mining: Practical Machine Learning Tools and Techniques* (3rd Ed.), San Francisco: Morgan Kaufmann.

WU, Y., AGRAWAL, D., and EL ABBADI, A. (2000), "A Comparison of DFT and DWT Based Similartiy Search in Time-Series Databases", in *Proceedings of the 9th International Conference on Information Knowledge Management, McLean VA*, New York: ACM Press, available from http://dl.acm.org/citation.cfm?id=354756.

YE, L., and KEOGH, E. (2009), "Time Series Shapelets: A New Primitive for Data Mining", in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France*, New York: ACM Press, available from http://dl.acm.org/citation.cfm?id=1557019&picked=prox&CFID=343051558&CFTOKEN=39520989.