Clustering time series from ARMA models with clipped data

A. J. Bagnall
School of Computing Sciences
University of East Anglia
Norwich, England
ajb@cmp.uea.ac.uk

G. J. Janacek
School of Computing Sciences
University of East Anglia
Norwich, England
gjj@cmp.uea.ac.uk

ABSTRACT

Clustering time series is a problem that has applications in a wide variety of fields, and has recently attracted a large amount of research. In this paper we focus on clustering data derived from Autoregressive Moving Average (ARMA) models using k-means and k-medoids algorithms with the Euclidean distance between estimated model parameters. We justify our choice of clustering technique and distance metric by reproducing results obtained in related research. Our research aim is to assess the affects of discretising data into binary sequences of above and below the median, a process known as clipping, on the clustering of time series. It is known that the fitted AR parameters of clipped data tend asymptotically to the parameters for unclipped data. We exploit this result to demonstrate that for long series the clustering accuracy when using clipped data from the class of ARMA models is not significantly different to that achieved with unclipped data. Next we show that if the data contains outliers then using clipped data produces significantly better clusterings. We then demonstrate that using clipped series requires much less memory and operations such as distance calculations can be much faster. Finally, we demonstrate these advantages on three real world data sets.

1. INTRODUCTION

The mining of time series in general and the clustering of time series in particular has attracted the interest of researchers from a wide range of fields, particularly from statistics [25], signal processing [10] and data mining [23]. An excellent survey of current research in the field has recently been published [21]. This growth in interest in time series clustering has resulted in the development of a wide variety of techniques designed to detect common underlying structural similarities in time dependent data. These techniques have been applied to data arising from many areas, for example: web mining [8]; finance and economics [27]; medicine [12]; meterology [5]; speech recognition [10]; gene expression analysis [4] and robotics [31].

This paper examines the effect of discretising time series on clustering accuracy. We transform real valued time series into binary series through the process of *clipping*. Clipping, or hard limiting, a time series is transforming a real valued time series Y into a binary series C where 1 represents above the population mean and 0 below, i.e. if μ is the population mean of series Y then

$$C(t) = \begin{cases} 1 & \text{if } Y(t) > \mu \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Clipped time series retain much of the underlying structure that characterises the real valued series. We demonstrate that clustering with clipped data provides the following benefits:

- if the series are long enough clustering with clipped data is not significantly less accurate than clustering with unclipped data;
- the clusters formed are significantly better with clipped data when there is at least a small probability of the data containing outliers;
- significant space and time complexity improvements can be achieved;
- algorithms developed for discrete or categorical data can be employed;
- clustering on clipped data can serve as a diagnostic method for outlier and model misspecification detection.

Our aim is to demonstrate these benefits over classes of alternative underlying models and from real world data. In [3] we assessed the effect of clipping on data originating from polynomial models on clusters found with the k-means algorithm. We found that, for data originating from a class of a mixture of two linear models, clipping does not significantly reduce the quality of the clusters when there are no outliers and does significantly improve the clustering accuracy in the presence of outliers, even when the probability of an outlier is small. In this paper we cluster using k-means, k-medoids and hierarchical methods with data from classes

of Autoregressive Moving Average (ARMA) model and real world data.

An ARMA(p, q) model has the form

$$\begin{aligned} Y(t) &= \phi_1 \cdot Y(t-1) + \ldots \phi_p \cdot Y(t-p) + \\ & \varepsilon(t) + \theta_1 \cdot \varepsilon(t-1) + \ldots + \theta_q \cdot \varepsilon(t-q) \end{aligned}$$

where $\varepsilon(t)$ are normally distributed random variables with variance σ^2. ARMA parameters can be estimated from data using the autocorrelations (see Section 3.2). It is shown in [18] that there is a relationship between the autocorrelation function of clipped and unclipped series and hence ARMA models fitted from clipped data tend asymptotically to models from unclipped data. This result, discussed in more detail in Section 3, forms the theoretical basis for using clipped data. The experimental evidence of the benefits of clipping with clustering is presented in Sections 4 to 6. We describe the space and time complexity improvements that clipping provides in Section 4. In Section 5, we cluster data from the simplest form of ARMA model, AR(1), with fixed parameters taken from related research [17, 35]. We show that our method of clustering unclipped data is at least as good as the alternative methods described in [17, 35]. This demonstrates that our choice of clustering algorithms and parameterisation is not the source of the variability observed in accuracy between clipped and unclipped data. We then demonstrate the benefits of clipping on models randomly sampled from the class of all stationary AR(1) models and show that, if the series are long enough, clipping does not decrease the accuracy of clustering and is better with outliers. In Section 6 we cluster data from fixed ARMA models used in [24, 25]. Results obtained are comparable to those published. We show that clipping produces more accurate clusters when there are outliers in the data. We cluster on models randomly sampled from the class of models given by all possible ARMA(1,0) to ARMA(2,2). We show that the benefits of clipping still exist in harder, multi-class, multi-model situations. In Section 7 we demonstrate the advantages of using clipped series on three real world data sets: ECG data and population data from [17] and reality check data set available from [20]. Finally, we summarise our findings and outline the future direction of this research in Section 8.

2. CLUSTERING TIME SERIES

The approaches to clustering time series can be categorised as model based or model free. Model based approaches (also called generative approaches) assume some form of the underlying generating process, estimate the model from each data then cluster based on similarity between model parameters. The most commonly assumed model forms are: polynomial mixture models (e.g. [11, 4, 3]); ARMA (e.g. [30, 24, 33, 17, 35]); Markov Chain and Hidden Markov models (MC and HMM) (e.g. [31, 8, 26, 32, 36]).

Reviews of methods used to cluster time series can be found in [3, 21]. Model free approaches involve the specification of a specific distance measure and/or a transformation of the data. Measures based on common subsequences (used in [9]), shape parameters [22] and correlation (e.g. [27]) have been proposed. Transformations used in conjunction with time series data mining include fast fourier transforms; dynamic time warping; wavelet transforms; and piecewise constant approximation. A large number of transformations are evaluated in [21].

One method that has similarities to our approach is the Symbolic Aggregate Approximation (SAX) transformation described in [23]. SAX is a transformation from real valued series into symbolic series through a two stage process of dimension reduction and discretisation. Clipping could be considered a specific type of SAX transformation with two classes and no dimension reduction. SAX may be of more general use in time series mining than clipping. However, the properties of clustering with clipped series are worthy of study because, firstly, binary data can be more compactly represented and efficiently manipulated and secondly, it is possible to assess their theoretical behaviour.

Other approaches use a combination of transformation and model fitting. For example, Kalpakis *et al* [17] fit ARMA models to the series then transform the fitted parameters into the associated Cepstrum. Our approach, described in detail in Section 3, is to transform the data then fit an ARMA model.

3. MODEL AND CLUSTERING METHODS

To test the affects of clipping we need to first specify:

1. a class of model from which the data in each cluster may come and the parameters of the model that require estimation (Section 3.1);
2. the fitting procedure to estimate the true model (Section 3.2);
3. the clustering method employed (Section 3.3);
4. the experimental procedure and method of evaluating performance (Section 3.4).

3.1 Generating Process

We assume that the series in each true underlying cluster is generated by an ARMA model. The model for cluster i is a random variable. A model without outliers is assumed to be of the form

$$\begin{aligned} X_i(t) &= \phi_1 \cdot X_i(t-1) + \ldots \phi_p \cdot X_i(t-p) + \\ & \varepsilon_i(t) + \theta_1 \cdot \varepsilon_i(t-1) + \ldots + \theta_q \cdot \varepsilon_i(t-q) \end{aligned} \quad (2)$$

where ε_i are $N(0, \sigma)$ and σ, p, q, ϕ_j and θ_j are constant for a model. One of the objectives of this paper is to demonstrate the robustness in the presence of outliers of using a clipped time series rather than the the raw data for clustering. Hence, we add a further term to Equation 2 to model the effect of outliers. A time series is assumed to be generated by a sequence of observations from the model

$$Y_i(t) = X_i(t) + r \quad (3)$$

where $r = s \cdot a \cdot b$. s is a constant, $a \in \{0, 1\}$ and $b \in \{-1, 1\}$ are observations of independent random variables, A and B, where A has density $f(a) = p^a(1-p)^{1-a}$ and B has density $f(b) = \frac{1}{2}$. r is a *random shock* effect that can occur with probability p, and if it occurs it has the effect of either adding or subtracting a constant s to the data (with equal probability). A time series y is a sequence of observations from a model Y, $y =< y(1), \ldots y(t), \ldots y(n) >$.

A binary data series is generated by clipping above and below the sample median. If ϕ_y is the sample median of the data series $y(t), t = 1, \ldots, n$, then the associated discretised time series, c, is defined as

$$c(t) = \begin{cases} 1 & \text{if} \quad y(t) > \phi_y \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

If several series are generated by the same model, we identify the generating process by a subscript i and the series index with a superscript j. Thus series y_i^j would be assumed to be the jth series from model Y_i.

A data set D is parameterised as follows: there are k models of the form given in Equation 3, each of which generates l time series, thus

$$D = \{y_1^1, \ldots, y_1^l, y_2^1, \ldots y_2^l, \ldots, y_k^l, \ldots y_k^l\}$$

each of the $l \cdot k$ time series is of length n and is sampled at the same points $t = 1, 2, \ldots, n$; σ defines the variability of static noise, s the level of random shocks and p the probability of shocks. Given a data set D, the problem is then to split the data into distinct sets or clusters, $D_1, D_2, \ldots, D_k$ such that,

$$\forall y_i^j \in D \quad y_i^j \in D_m \quad \text{if and only if} \quad i = m.$$

To form these clusters, we fit an ARMA model to each series then cluster on the basis of the similarity of the fitted parameters. The method of fitting and clustering is described in Section 3.2.

3.2 Model Estimation and Properties

Any invertible ARMA model can be represented as an infinite AR model,

$$\varepsilon(t) = \sum_{j=0}^{\infty} \pi_j X(t-j).$$

We estimate the AR model for each series by using a standard three stage procedure. Firstly the autocorrelations $\rho_1, \rho_2, \ldots$ up to a fixed maximum lag are calculated (set to 50 in experiments reported). Secondly the partial fitted models and hence the partial autocorrelations are found by solving the Durbin-Levinson recursion. These use the fact that the correlation matrix is Toeplitz to create a computationally efficient iteration. If the ith autoregressive term of the model fit to j parameters is denoted $\phi_{i,j}$ and $\phi_{1,1} = \rho_1$ then the Durbin-Levinson recursion is

$$\phi_{k,k} = \frac{\rho_k - \rho_{k-1}\phi_{1,k-1} - \rho_{k-2}\phi_{2,k-1} - \ldots - \rho_1\phi_{k-1,k-1}}{1 - \rho_{k-1}\phi_{k-1,k-1} - \rho_{k-2}\phi_{k-2,k-1} - \ldots - \rho_1\phi_{1,k-1}}$$

Finally the model that minimizes the Akaike information criteria (AIC) is chosen [1]. The AIC for a model with k AR terms fitted to a series of length n is defined as

$$AIC = \ln \hat{\sigma}^2 + 2 \cdot \frac{k}{n}$$

Further information about fitting models can be found in [15]. The choice of this methodology (DL/AIC) is made on pragmatic grounds. The Durbin Levinson approach is simple and efficient especially when used with AIC as a fitting criterion, since the AIC can be calculated directly from the partial autocorrelations $\phi_{i,j}$. In studies of model selection such as [28] AIC performs at least as well as other possibilities e.g. BIC [29]. The drawbacks of the procedure are well known (see [6]) and in our view minor. DL is effectively a least squares approach. The alternative is to use likelihood methods. These have useful asymptotic theory, but at the cost of increased complexity. Broerson [6] among others has pointed out that exact maximum likelihood solutions may not give a real advantage in some time series problems. Other possibilities are the Burg approach [7] and Godolphins method [13] or to estimate via the spectrum. For our purposes any advantages of these approaches are of less importance than the DL/AIC benefits of simplicity and efficiency.

In [2] we demonstrate that using the Durbin-Levison recursions do not adversely affect the efficiency of the estimates of the model parameters, by repeating the experiments described in Section 4 of [18].

Kedem, [18] and Kedem and Slud [19] derived several useful links between the original and the clipped series. For example, if we assume that the unclipped series y is both Gaussian and stationary to second order, we can use the bivariate normal distribution to compute probabilities. We assume there are no outliers in the data ($p = 0$), and we denote $\rho_Y(r)$ the autocorrelation of lag r for an unclipped model Y and $\rho_C(r)$ the autocorrelation of lag r for a clipped model C. It is not difficult to show that

$$P[C(r+s) = 1 \text{ and } C(s) = 1] = \frac{1}{4} + \frac{1}{2\pi}\sin^{-1}\{\rho_Y(r)\}.$$

It follows that the autocorrelations of the clipped series, $\rho_C(k)$, are given by

$$\rho_C(k) = \frac{2}{\pi}\sin^{-1}\rho_Y(k) \qquad k = \ldots -2, -1, 0, 1, 2, \ldots \tag{5}$$

Since the binary series gives the runs in the time series, we have, from equation 5, a link between the probability of a run and the correlation structure of the original process. Using a Taylor expansion we have

$$\rho_C(k) = \frac{2}{\pi}\left\{\rho_Y(k) + \frac{1}{6}\rho_Y^3(k) + \frac{3}{40}\rho_Y^5(k) + O(\rho_Y^7(k))\right\}$$

which implies that the two series will have similar correlations but that those from the clipped series will have smaller absolute values. In fact we can go further and can show that with the Gaussian assumptions above

$$E[C(t)Y(t+k)] = \frac{1}{2}E[Y(t+k)|C(t)] = \frac{1}{\sqrt{2\pi}}\rho_Y(k)$$

Now define

$$e(t) = C(t) - \frac{1}{\sqrt{2\pi}}Y(t)$$

since $E[e(t)Y(t+k)] = 0$ we have a process

$$C(t) = \frac{1}{\sqrt{2\pi}}Y(t) + e(t)$$

where $E[e(t)] = 0$. This implies that if the original series

can be written in autoregressive form, say

$$Y(t) + \phi_1 Y(t-1) + \phi_2 Y(t-2) + \ldots + \phi_p Y(t-p) = \varepsilon(t)$$

then the clipped series has the form

$$C(t) + \phi_1 C(t-1) + \ldots + \phi_p C(t-p) = \frac{\varepsilon(t)}{\sqrt{2\pi}} + e(t) - \phi_1 e(t-1) - \ldots - \phi_p e(t-p).$$

Here of course the errors are correlated. We can deduce that, given the covariance matrix of the errors $e(t)$, we can produce a linear transformation which will give us uncorrelated errors. Hence we deduce *a linear ARMA model for original series implies a linear ARMA model for the clipped series.*

3.3 Clustering Procedure

Given a set of parameters to an ARMA model fitted by the procedure described in Section 3.2 we cluster with k-means and k-medoids using the Euclidean distance between the fitted parameters. For any clustering experiment, we restart k-means and k-medoids and take the clustering that minimizes the within cluster distance. In Section 7.3 we also use hierarchical clustering methods.

More complex clustering mechanisms are used for similar data in [17, 35, 25]. In [21] it is observed that on many problems Euclidean distance performs as well or better than more complex measures. In Sections 5 and 6 we justify our choice of distance measure and clustering algorithm by demonstrating that Euclidean distance with k-means performs comparably with other published techniques on data from ARMA models.

3.4 Experimental Procedure

An experiment consists of the following steps: set the parameters; define a model space $\mathcal{M}$; randomly sample the model space to determine the k cluster models; for each cluster model, generate l series of length n; fit a model to each series using the method described in 3.2; cluster the series u times, taking as the best the clustering with the lowest within cluster distance from the centroid; evaluate the accuracy of the best clustering. Clustering performance is measured by the classification accuracy, i.e. the ratio of the percentage of the data in the final clustering that is in the correct cluster. Note we are measuring accuracy on the training data rather than applying the data to a separate testing data set. We do this because wish to measure the effects of outliers in the training data rather than assess the algorithm's ability to solve the clustering problem. We use this measure rather than some of the alternatives (see [14]) since we know the correct clustering.

For a given clustering we measure the accuracy by forming a $k \times k$ contingency matrix. Since the clustering label may not coincide with the actual labelling (e.g. all those series in cluster 1 may be labelled cluster 2 by the clustering algorithm) we evaluate the accuracy (number correctly classified divided by the total number of series) for all possible $k!$ permutations of the columns of the contingency table. The achieved accuracy is the maximum accuracy over all permutations.

4. SPACE AND TIME IMPROVEMENTS

One of the benefits of using clipped series is that the data can be packed efficiently in integer arrays and manipulated using bit operators. A series of doubles of length n can be stored in an array of $n/64$ integers, a worthwhile reduction particularly in applications that involve very long or very many series. For very large data sets packing the series may make the difference between being able to store the series in main memory and having to access the data from disk. Hence packing clipped series could provide a significant time improvement for clustering algorithms that require the recalculation of models directly from the data at each iteration of the clustering.

Less operations may also be required to perform calculations on binary series. One benefit of using clipped data comes in the calculation of the autocorrelations. An autocorrelation requires the calculation of the sum of the product of lagged series with the original, i.e.

$$R_i = \sum_{t=i}^{n} C(t) \cdot C(t-i).$$

The autocorrelation calculations for a clipped series represented as arrays of binaries requires fewer operations than that of an unclipped series (see [18] for details). We can get further speed up by packing the clipped series into integers then using logical bit operators and shift operators to find the multiplication. To find the sum R_i we shift a copy of clipped C left i places, logical *AND* the two series together to perform the multiplication, then sum the number of bits in the result to find R_i. This is much faster than looping through n times. We can also speed up the operation to sum the bits. Any algorithm to count the bits is $\Omega(n)$. We can however improve the constant terms in the time complexity function with the use of a lookup table for the number of bits in 8 bit integers. We use shift operators to evaluate the integer value of each 8 bit sequence then use a lookup table to find the number of bits. Figure 4 demonstrates that calculating the autocorrelations is then approximately three times faster with the clipped data even when the series are stored in main memory. Each point in Figure 4 gives the time taken (averaged over 30 series) to find the autocorrelations. The times include the procedure to clip the data. The median of each series is found using the quick select algorithm, hence clipping can be done in $O(n)$ time on average.

Packing the data also offers the potential for improving the performance of the clustering algorithm. For example, the mean calculation for k-means becomes the problem of counting bits in subseries, which can be done more efficiently than the equivalent calculation with the unclipped data using masks and bit count lookup tables. As demonstrated in Section 7.3 distance calculations can be made more efficient by similar mechanisms.

5. AR(1) MODELS

The first set of experiments are on data from AR(1) models used in [17, 35]. Kalpakis *et al* [17] and Xiong and Yeung [35] both include experiments with clustering data from two AR(1) models with parameter in the range $\phi = 0.3$ to $\phi = 0.6$. Kalpakis *et al* fit a model to the data using Linear Predictive Coding then transform the parameters into the associated Cepstrum. They cluster using the Euclidean

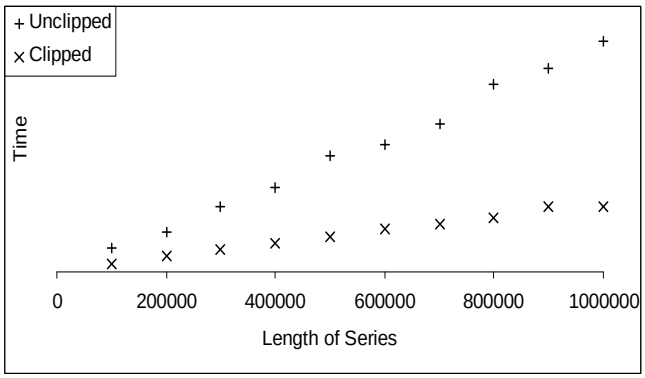

Figure 1: Average time taken to find autocorrelations

distance between Cepstrum as a distance metric.

Xiong and Yeung used an EM clustering algorithm based on the maximum likelihood estimates of the parameters of mixture ARMA models. The majority of the experimentation involved data from the same AR(1) models used by Kalpakis *et al.*

To demonstrate that clustering data from AR(1) models using k-means with Euclidean distance on the fitted AR parameters is approximately as accurate as the Cepstrum based technique used in [17] and the mixtures of AR models (MAR) method described in [35] we reproduce the experiments presented in Table 2 in [35] using the Cepstrum method of [17] and the Euclidean distance between parameters. Data sets of 15 series were generated from two AR(1) models. Data from one cluster is from a model with $\phi = 0.3 \pm 0.01$, data from the second cluster is from a model with ϕ as given in the first column of Table 1. It is assumed (but not stated in [35]) that the data series were of length 256, as in [17]. We perform two experiments to demonstrate how the fitting procedure affects the accuracy of the clustering.

Table 1 summarises all the results. Clustering on the fitted parameters and their Cepstrum was performed using restart k-means. The results for the unclipped data with no outliers and no AIC were found through fitting an AR model with a maximum of 20 terms without any check to measure the significance of the included variables. Clustering on Euclidean distance between the parameters is significantly worse than with Cepstrum at each treatment level. This is because of the noise introduced by the insignificant parameter estimates. The Cepstrum reduce the effect of this noise by smoothing and hence produce more accurate clusterings. We can achieve equivalent accuracy with Euclidean distance by introducing the standard model selection method of minimizing AIC described in Section 3.2. AIC nearly always reduces the model to 1 or 2 parameters. The second set of results in Table 1 show that the results are comparable to those given in Table 2 of [35]. This suggests that for this type of data k-means is as good as the MAR algorithm presented in [35]. Secondly, there is no significant difference between the mean accuracy at any treatment level between the Cepstrum clustering and the AIC clustering. This demonstrates

Table 1: Clustering accuracy of k-means on AR(1), averaged over 30 runs

	Accuracy (min/**av**/max/*st dev*)	
AR Coeff	CEP	Euclidean Dist
without AIC, unclipped data, no outliers		
0.60±0.01	(0.93/**0.99**/1.00/*0.02*)	(0.73/**0.97**/1.00/*0.05*)
0.50±0.01	(0.73/**0.94**/1.00/*0.04*)	(0.50/**0.78**/1.00/*0.12*)
0.40±0.01	(0.63/**0.76**/0.97/*0.07*)	(0.50/**0.58**/0.70/*0.07*)
with AIC, unclipped data, no outliers		
0.60±0.01	(0.93/**1.00**/1.00/*0.01*)	(0.90/**0.99**/1.00/*0.02*)
0.50±0.01	(0.73/**0.95**/1.00/*0.05*)	(0.87/**0.95**/1.00/*0.04*)
0.40±0.01	(0.60/**0.79**/0.97/*0.07*)	(0.53/**0.78**/0.93/*0.08*)
with AIC, clipped data, no outliers		
0.60±0.01	(0.73/**0.94**/1.00/*0.05*)	(0.80/**0.92**/1.00/*0.05*)
0.50±0.01	(0.53/**0.84**/0.97/*0.08*)	(0.53/**0.81**/0.97/*0.09*)
0.40±0.01	(0.50/**0.70**/0.90/*0.08*)	(0.50/**0.66**/0.87/*0.09*)
with AIC, unclipped data with outliers, $p = 0.01$		
0.60±0.01	(0.57/**0.83**/1.00/*0.09*)	(0.63/**0.85**/0.97/*0.08*)
0.50±0.01	(0.57/**0.73**/0.87/*0.07*)	(0.53/**0.76**/0.93/*0.08*)
0.40±0.01	(0.50/**0.63**/0.77/*0.07*)	(0.50/**0.61**/0.77/*0.07*)
with AIC, clipped data with outliers, $p = 0.01$		
0.60±0.01	(0.77/**0.94**/1.00/*0.05*)	(0.80/**0.95**/1.00/*0.05*)
0.50±0.01	(0.67/**0.84**/0.97/*0.07*)	(0.57/**0.81**/0.97/*0.08*)
0.40±0.01	(0.50/**0.69**/0.87/*0.09*)	(0.50/**0.67**/0.87/*0.08*)

that the LPC Cepstrum based clustering described in [17] is not better than the Euclidean based clustering if a standard method is used to reduce the noise by removing redundant parameter estimates.

The third part of Table 1 shows that clustering with clipped data does seem to degrade clustering performance for this model with $n = 256$. The averages are lower for all models. However, the maximum of each run is approximately the same. The poor average figure is caused by occasional very bad run (as shown by the wide differences in minimum values) and the short data length.

Obviously the LPC Cepstrum and MAR techniques may be superior for data from other classes of model. However, our interest focuses on the benefits of clipping the data with possible outliers. There is no evidence to suggest that more complex algorithms provide significantly more accurate clustering for data from AR(1) models. Hence we use the simplest clustering methods available (k-means and k-medoids) and suggest that it is reasonable to assume that the benefits of clipping observed would also be seen if the more complex methods described in [35, 17] were used instead.

To demonstrate the benefits of clipping we repeated the experiments with a small probability of an outlier, as defined in Section 3.1. The results in the fourth and fifth part of Table 1 clearly show that the extra noise in the data means that the clustering on the clipped data produces more accurate clusterings. The mean difference between the clipped average and the unclipped average is significantly greater than zero at every level.

Table 2 gives the accuracy of the k-means and k-medoids

Table 2: Accuracy using k-means and k-medoids on clipped data with $\phi_2 = 0.50$

	k-means	k-medoids
unclipped, p=0	0.95	0.93
clipped, p=0	0.81	0.80
unclipped, p=0.01	0.57	0.58
unclipped, p=0.01	0.81	0.81

for the situations considered in previous experiments. The results in Table 2 demonstrate that k-means clusters as accurately as k-medoids. There is no significant difference between the means of each combination. We use k-means for the majority of experimentation because it is faster.

To reinforce the observation that clustering with clipped data is better when the data contains outliers, we experiment with the parameter ϕ chosen randomly on an interval $[-0.9, 0.9]$ for each cluster. The wider the range the easier, on average, the clustering task. If the series are of a reasonable length the clustering algorithms should be able to find the correct clusterings in the majority of cases. Figure 5 shows the mean and median accuracy difference averaged over 30 runs for random models. Two observations can be made about these graphs. Firstly, the mean accuracy is lower with the clipped data when there are no outliers. This is caused by the occasional very bad clustering, a point illustrated by the fact that the median values are the same. This effect is reduced as the size of the series increases: with $n = 2000$ and $p = 0$ the observed mean difference in accuracy was -0.0013; and with $n = 10000$ we observed a positive mean difference (0.0007). In both cases there was no significant difference between the medians (using a Wilcoxon test for paired samples).

Secondly, for low probability of outliers, the average clustering accuracy using unclipped data is significantly worse. As the probability of a data being an outlier increases the accuracy on the unclipped data decreases more rapidly than the accuracy on the clipped data, demonstrated by the positive slope of both graphs in Figure 5.

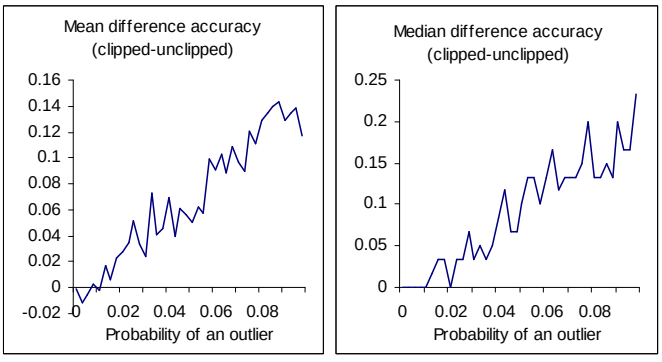

Figure 2: Difference in clustering accuracy between the clipped and unclipped series for varying values of p

Thus for two classes of data from AR(1) models, k-means clustering based on the Euclidean distance between fitted parameters and $n = 256$, clipping the data does not significantly reduce the median accuracy of clustering when there are no outliers and significantly increases the accuracy even when the probability of an outlier is small. To demonstrate that this result is not an artifact of the clustering algorithm we repeat the experiments using the k-medoids algorithm. Table 3 shows the results for clipped and unclipped data. The pattern of performance is the same: when there are no outliers the mean for unclipped data is higher but the medians are approximately the same; when the probability of a data being an outlier is 0.01 the average (both mean and median) clustering is significantly better with the clipped data.

Table 3: Accuracy using k-medoids to cluster clipped data from random AR(1). (mean/median/*standard deviation*)

	Unclipped	Clipped
No Outliers	(0.91/**1.00**/*0.164*)	(0.87/**0.97**/*0.176*)
Outlier p=0.01	(0.80/**0.83**/*0.21*)	(0.86/**0.97**/*0.182*)

To demonstrate that the result is observable in harder clustering problems, the experiments were repeated with larger values of k. Figure 5 shows the median accuracy for clipped and unclipped data for $p = 0$ on the left hand graph and $p = 0.01$ on the right. For each cluster we are generating 15 series of length 256 (i.e. $l = 15$ and $n = 256$). Since we keep l and n constant, the accuracy of clustering decreases for both clipped and unclipped as k increases. This achieves the desired result of presenting progressively harder clustering problems. Figure 5 demonstrates that clustering on the clipped data improves the accuracy in the presence of outliers. With no outliers, clustering with clipped data is less accurate, but this gap in accuracy reduces as n increases. From these experiments we can conclude that if

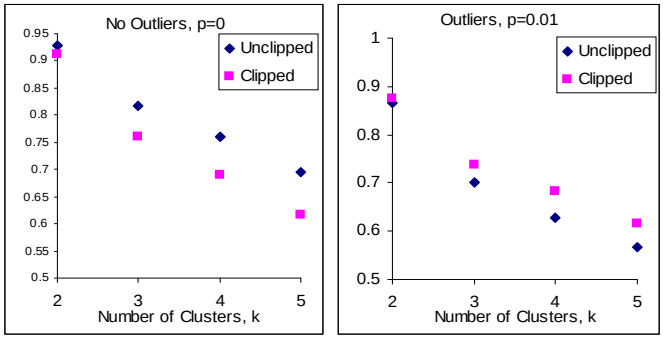

Figure 3: Clustering accuracy for alternative values of k, with no outliers (left) and a small probability of outlier (right)

a set of time series are derived from a mixture model of AR(1) processes, and if each series is of adequate length, then clustering (using k-means or k-medoids) with clipped data rather than unclipped data: significantly speeds up the

clustering process; does not significantly decrease the accuracy of clustering; and significantly increases the clustering accuracy when the data contains undetected outliers.

6. ARMA MODELS

Maharaj has investigated clustering time series from ARMA models in [24, 25]. She fits AR series to data from a variety of ARMA models, then clusters using a greedy algorithm with a distance measure derived from the p-value of a significance test. The first experiment involved five generating models ($k = 5$) each generating four series ($l = 4$) of length 200 ($n = 200$). The models used were AR(1), MA(1), AR(2), MA(2), ARMA(1,1). The second experiment was designed to be easier. The parameters were the same except k was changed to four and the models used were AR(0), AR(1), MA(1), ARMA(1,1). The results for k-means clustering using clipped and unclipped data with no outliers is shown in Table 4. Maharaj assesses the quality of a clustering by a measure based on the number of clusters exactly correct. This measure is not directly comparable to the accuracy measure we use. A simple conversion can be performed to estimate the accuracy by our definition from the measure used in [24]. If m clusters are exactly correct then $m \cdot l$ data are correct. Of the remaining $(k-m) \cdot l$ data, if we assume each is equally likely to be in any of the remaining clusters, and we assume that there are $k+1$ clusters, then we can estimate the overall accuracy. The estimated accuracy of the results reported in [24] is 0.60 for Experiment 1 and 0.95 for Experiment 2. Unclipped clustering with k-means performs better, although of course we have the advantage of knowing the number of clusters, k, *a priori*. The clipped series do not cluster as well on the easier problem (Experiment 2). The relatively large number of clusters means the data is not long enough to mitigate against the amount of information that is lost through clipping. However, if we add some outliers to the data, the situation is reversed. Table 5 shows that the outliers effect the accuracy of clusters on the clipped series far less than that on the unclipped.

Table 4: Clustering data from fixed ARMA models with no outliers

	Experiment 1 (k=5, l=4, n=200)				
	Mean	Median	Min	Max	SD
Unclipped	0.79	0.8	0.5	1	0.11
Clipped	0.61	0.6	0.35	0.95	0.084
	Experiment 2 (k=4,l=4,n=200)				
Unclipped	0.98	1	0.63	1	0.048
Clipped	0.75	0.75	0.5	1	0.117

For the class of mixture ARMA generating models we wish to demonstrate that if there are no outliers in the data and the series are long enough then clustering with clipped data does not significantly decrease the accuracy and if there are a small number of outliers, then clipping the data gives a significant increase in accuracy. To test these hypotheses we generated mixture models for two classes randomly structured from AR(1), MA(1) to ARMA(2,2) and with random parameters. We are primarily interested in clustering stationary ARMA models. This is because for long series non stationary models quickly increase or decrease to a point where numeric errors can occur. To make sure the random model generated is stationary we perform a simple test where a series is discarded if a value above or below a certain threshold is observed.

Table 5: Clustering data from fixed ARMA models with probability of an outlier equal to 0.01

	Experiment 1 (k=5, l=4, n=200)				
	Mean	Median	Min	Max	SD
Unclipped	0 .57	0.55	0.4	0.85	0.08
Clipped	0 .60	0.6	0.35	0.85	0.09
	Experiment 2 (k=4, l=4, n=200)				
Unclipped	0.67	0.69	0.44	1.00	0.11
Clipped	0.74	0.75	0.44	1.00	0.12

10 series of length 1000 were generated from each class. The mean classification accuracy for clipped and unclipped data with and without outliers is shown in Table 6, averaged over 200 experiments. For the data with no outliers, we are unable to reject the null hypothesis that the population mean difference is zero using a paired two sample t-test on the mean difference, hence we conclude there is no evidence to suggest that clustering accuracy is worse with clipped data for data from this model. When performing the test with the data containing outliers we are able to reject the null hypothesis at the 1% level and conclude that there is evidence to suggest the average accuracy is higher with clipped data when the probability of a data being an outlier is 0.01.

Table 6: Clustering accuracy on random ARMA models, $k = 2$

	Unclipped	Clipped
No Outliers	0.9675	0.9668
Outlier $p = 0.01$	0.94225	0.9570

On the majority of cases both methods find the correct clustering. This is because of the wide range of possible models sampled make this a fairly easy problem. The number of times one algorithm outperforms another is given in Table 7. It shows that when there are no outliers, unclipped outperformed clipped on 25 models and clipped outperformed unclipped on 21. With outliers, clipped did better on 29 occasions, unclipped on only 12.

Table 7: Number of times one method outperformed the other on random ARMA models, $k = 2$

	Unclipped	Clipped
No Outliers	25	21
Outlier $p = 0.01$	12	29

Figure 6 illustrates how the accuracy on the unclipped data decreases as the frequency of outlier increases when the series are of length $n = 200$.

A more complex clustering problem involves random data from ARMA models with more clusters. To test whether the benefits of clipping are still present on harder problems

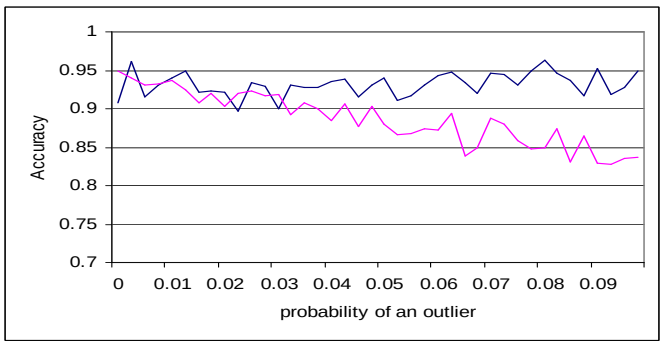

Figure 4: Accuracy for increasing probability of an outlier

we generated data as before but with k from 2 to 9. Figure 5 shows the accuracy for clipped and unclipped data with $n = 400$ and no outliers. Figure 6 shows the results from the same experiment with the probability of an outliers equal to 0.01.

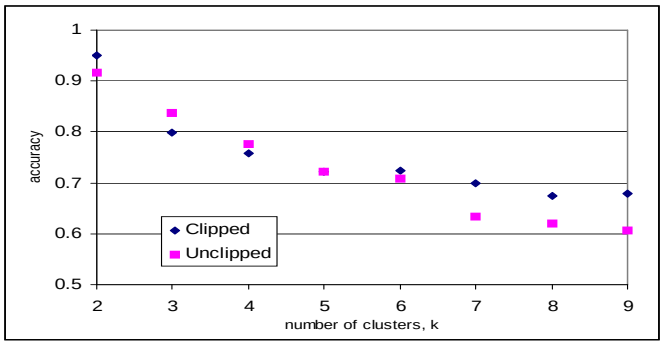

Figure 5: Mean accuracy for data with no outliers

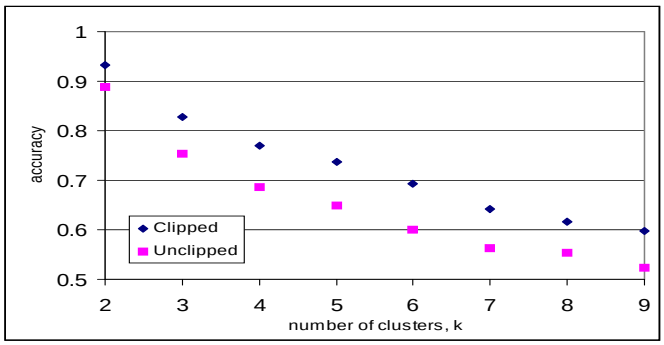

Figure 6: Mean accuracy for data with probability of an outlier 0.01

Figures 5 and 6 demonstrate that: the clustering problem becomes harder with the number of clusters; with no outliers clipping does not decrease the accuracy compared to unclipped data even for harder problems up to six clusters; and if outliers are present, clipping increases the accuracy.

Clipping is less accurate for 7, 8 and 9 clusters. It is consistent with previous observations to suppose that the clipped series would be as accurate for larger number of clusters if the series were longer and that the unclipped data would not reach the level of accuracy of clipped data if outliers were present, however long the series.

7. OTHER DATA SETS

Kalpakis *et al* [17] and Xioung and Yeung [35] also test their algorithms on four real data sets, available from [16]. We use two of the data sets, an ElectroCardioGram (ECG) data set and a population data set. We also evaluate the technique of clipping on a data set from [20], the Reality Check data set.

7.1 ECG data set

The ECG data consists of 70 series of ECG measurements. 22 of these patients suffered from malignant ventricular arrhythmia and are denoted Group 1 or venarh. Group 2, or normal, is made up of measurements of 13 healthy people. Group 3 (suprav) contains data on 35 people with the condition superventricular arrhythmia. The ECG data is characterised by large peaks and troughs and a high degree of periodicity which would normally indicate that autoregressive models are unsuitable [15]. However, both [17, 35] fitted ARMA models. Kalpakis *et al* normalise the data, smooth each series with a period 3 moving average, apply the difference operator three times and fit AR(2) models to the data (i.e. fit an ARIMA(2,3,0) to normalised, smoothed data). They then perform two clustering experiments. Experiment 1 involves clustering data groups 1 and 2 (venarh and normal). Experiment 2 cluster involves clustering data groups 2 and 3 (normal and suprav). The results for restart

Table 8: Results for clustering ECG data

	Unclipped	Clipped
Experiment 1	0.625	0.604
Experiment 2	0.625	0.825

k-means shown in Table 8 are within the range of values for the techniques used in [17]. Clipping the data actually leads to a higher level of accuracy in Experiment 2 than the best reported in [17], even without outliers in the data. This improvement can be explained by the unsuitability of an ARMA model. The peaks in the data set have a similar effect to the estimators as outliers and clipping removes their unwanted influence. This highlights another advantage of clustering with clipped data. A large difference in accuracy between clipped and unclipped data could indicate the presence of outliers, but it could also serve as a means of detecting model misspecification. A model based on estimates of frequency and amplitude may be more suitable for ECG data.

7.2 Population data set

The population data set consists of annual population data for 20 states of the US for the period 1900-1999. The states are grouped into two clusters, the first with an exponential increasing trend, the second with a stabilizing trend. Following Kalpakis, we fit ARMA(1,1,0) to each series and

cluster on the parameters. We also added noise to the data (probability of an outlier $p = 0.02$). The results are given in Table 9. The results for k-means with no noise are within

Table 9: Results for clustering population data

	unclipped	clipped
No Noise	0.684	0.58
Noise p=0.02	0.53	0.58

the range of those reported by Kalpakis for autocorellation based techniques. Clipping results in a significant degradation in performance. The most obvious explanation for this is that the series are too short (100 data). The benefits of clipping can still be observed when noise is added. The outliers do not effect the accuracy of the clusters formed with clipped data, but they significantly degrade the accuracy of the unclipped data.

7.3 Reality Check

The Reality Check data is a dataset for which the intuitive clustering can be found with Euclidean distance metrics. It serves a useful purpose of providing a basis for alternative distance metrics. We use the data to demonstrate that clipping does not massively alter the structure of the dendrogram that results from clustering using hierarchical clustering methods. Figures 7 and 8 show the dendrograms with clipped and unclipped data using average linkage hierarchical clustering. The graphs were produced with Clustan software [34]. The only difference in the dendrograms is the order groups (9,10) and (6,7,8) are linked into largest clusters. The fact that this is not a major difference is demonstrated by the observation that the dendrograms for unclipped data using nearest and furthest linkage show a similar amount of deviation from the average linkage graph to the difference between Figures 7 and 8.

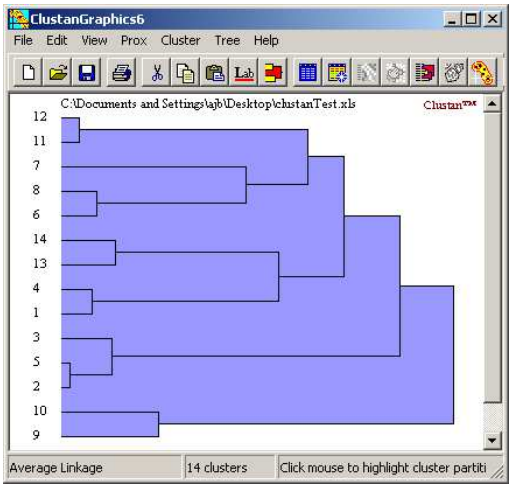

Figure 7: Average linkage dendrogram for unclipped data

Hierarchical algorithms require the calculation of all distances prior to the clustering process. Figure 9 demonstrates

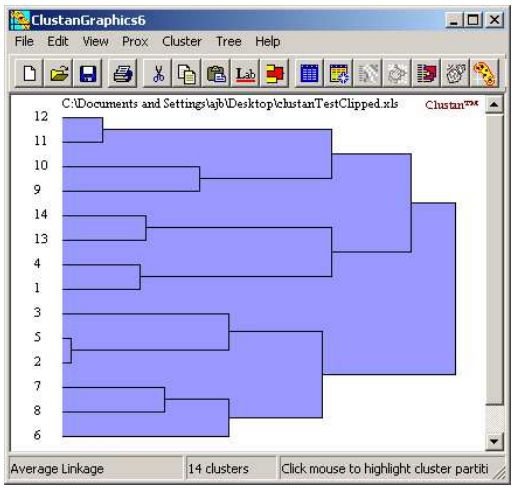

Figure 8: Average linkage dendrogram for clipped data

that the distance calculation with clipped data is approximately 5 times faster than with unclipped data. This speed up, coupled with the reduced space requirements of clipped data, could provide a justification for clipping in its own right.

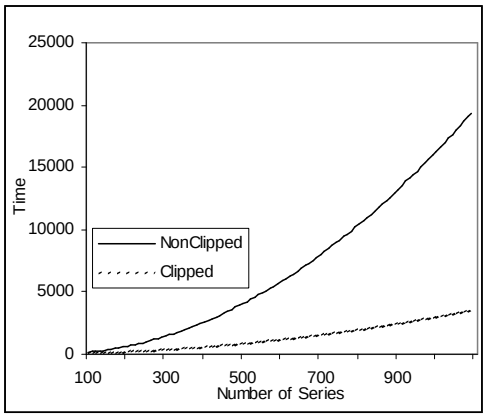

Figure 9: Average time taken to find distances between series length 1000

8. CONCLUSIONS

This paper discusses the benefits of clipping data when clustering time series. For data from the class of ARMA models we show that, theoretically, the models fitted to clipped data asymptotically approach the model fitted from the unclipped data. We demonstrate the application of this result in clustering through a series of clustering experiments with k-means and k-medoids using Euclidean distance on the fitted parameters as a distance metric. We justify our choice of clustering algorithm and distance metric by repeating published experiments and reproducing comparable results. We then demonstrate how the property of clipped series can be used in clustering by randomly sampling AR(1) and ARMA models. Over this class of model we show that: calculating

the autocorrelations is faster with clipped data; if the data series are long enough clustering accuracy on clipped data is not significantly less than clustering accuracy on unclipped data; and if the data contains outliers, the clustering accuracy on clipped data is significantly better.

Clustering clipped data may provide a sufficiently good clustering in its own right, particularly when data sets are massive and the clustering algorithm is slow. It can also serve as a means of outlier detection or identification of the use of an inappropriate model, as demonstrated in Section 7.1.

Our advice to researchers wishing to cluster time series would be to start with clipped data, then examine any results from more sophisticated transformations in relation to the results obtained after clipping, particularly if the series are long and time and space are an important considerations.

The next step in this research will be to evaluate the effects of clipping on data originating from Hidden Markov Models and on more of the real data sets available at [20] using a wider range of clustering algorithms.

9. REFERENCES

[1] H. Akaike. Likelihood of a model and information criteria. *Journal of Econometrics*, 16:3–14, 1981.

[2] A. J. Bagnall and G. Janacek. Clustering time series from ARMA models with clipped data. Technical Report CMP-C04-01, School of Computing Sciences, University of East Anglia, 2004.

[3] A. J. Bagnall, G. Janacek, B. d. Iglesia, and M. Zhang. Clustering time series from mixture polynomial models with discretised data. In *Proceedings of the second Australasian Data Mining Workshop*, pages 105–120, 2003.

[4] Z. Bar-Joseph, G. Gerber, D. Gifford, T. Jaakkola, and I. Simon. A new approach to analyzing gene expression time series data. In *Proceedings of The Sixth Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, pages 39–48, 2002.

[5] R. Blender, K. Fraedrich, and F. Lunkeit. Identification of cyclone-track regimes in the north atlantic. *Quart J. Royal Meteor. Soc.*, (123):727–741, 1997.

[6] P. Broerson and S. de Waele. Empirical time series and maximum likelihood estimation. In *Proc 2nd IEEE Benelux Signal Processing Symposium*, 2000.

[7] J. P. Burg. Maximum entropy spectral analysis. presented at 37th meeting of the Society of Exploration Geophysicists, Oklahoma City, 1967.

[8] I. V. Cadez, D. Heckerman, C. Meek, P. Smyth, and S. White. Visualization of navigation patterns on a web site using model-based clustering. In *Knowledge Discovery and Data Mining*, pages 280–284, 2000.

[9] G. Das, D. Gunopulos, and H. Mannila. Finding similar time series. In *Principles of Data Mining and Knowledge Discovery*, pages 88–100, 1997.

[10] E. Dermatas and G. Kokkinakis. Algorithm for clustering continuous density HMM by recognition error. *IEEE Tr. On Speech and Audio Processing*, 4(3):231–234, 1996.

[11] S. Gaffney and P. Smyth. Curve clustering with random effects regression mixtures. In C. M. Bishop and B. J. Frey, editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.

[12] A. B. Geva and D. H. Kerem. *Fuzzy and Neuro-Fuzzy Systems in Medicine*, chapter 3. Brain state identification and forecasting of acute pathology using unsupervised fuzzy clustering of EEG temporal patterns. CRC Press, 1998.

[13] E. J. Godolphin. A direct representation for the large-sample maximum likelihood estimator of a gaussian autoregressive-moving average process. *Biometrika*, 71(2):281–289, 1984.

[14] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. On clustering validation techniques. *Journal of Intelligent Information Systems*, 17(2-3):107–145, 2001.

[15] G. J. Janacek. *Practical Time Series*. Ellis Horwood, 2001.

[16] K. Kalpakis. Distance measures for clustering time series. `http://www.csee.umbc.edu/ kalpakis`.

[17] K. Kalpakis, D. Gada, and V. Puttagunta. Distance measures for effective clustering of ARIMA time-series. In *Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01)*, pages 273–280, 2001.

[18] B. Kedem. Estimation of the parameters in stationary autoregressive processes after hard limiting. *Journal of the American Statistical Association*, 75:146–153, 1980.

[19] B. Kedem and E. Slud. On goodness of fit of time series models: An application of higher order crossings. *Biometrika*, 68:551–556, 1991.

[20] E. Keogh and T. Folias. The ucr time series data mining archive. `http://www.cs.ucr.edu/ eamonn/TSDMA/`.

[21] E. Keogh and S. Kasetty. On the need for time series data mining benchmarks: A survey and empirical demonstration. *Data Mining and Knowledge Discovery*, 7(4):349–371, 2003.

[22] K. Kosmelj and V. Batagelj. Cross-sectional approach for clustering time varying data. *Journal of Classification*, 7:99–109, 1990.

[23] J. Lin, E. Keogh, S. Lonardi, and B. Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11. ACM Press, 2003.

[24] E. A. Maharaj. A significance test for classifying.

[25] E. A. Maharaj. Clusters of time series. *Journal of Classification*, 17:297–314, 2000.

[26] T. Oates, L. Firoiu, and P. R. Cohen. Using dynamic time warping to bootstrap HMM-based clustering of time series. *Lecture Notes in Computer Science*, 1828:35–52, 2001.

[27] P. Ormerod and C. Mounfield. Localised structures in the temporal evolution of asset prices. In *New Approaches to Financial Economics*. Santa Fe Conference, 2000.

[28] E. Parzen. Some recent advances in time series modelling. *IEEE Transactions on Automatic Control*, 19(6):723–730, 1974.

[29] D. K. Pauler. The.

[30] D. Piccolo. A distance measure for classifying ARIMA models. *Journal of Time Series Analysis*, 11(2):153–164, 1990.

[31] M. Ramoni, P. Sebastiani, and P. Cohen. Bayesian clustering by dynamics. *Machine Learning*, 47(1):91–121, 2002.

[32] P. Smyth. Clustering sequences with hidden markov models. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 648. The MIT Press, 1997.

[33] P. Tong and H. Dabas. Cluster of time series models: An example. *Journal of Applied Statistics*, 17:187–198, 1990.

[34] D. Wishart. Clustan clustering software. `http://www.clustan.com/`.

[35] Y. Xiong and D.-Y. Yeung. Mixtures of.

[36] S. Zhong and J. Ghosh. Scalable, balanced model-based clustering. In *Proceedings of SIAM Int. Conf. on Data Mining*, 2003.