Published version: Journal of New Music Research 40(4): 325-336, 2011
Open URL of article: http://www.tandfonline.com/openurl?genre=article&issn=1744-5027&volume=40&issue=4&spage=325

Learning timbre analogies from unlabelled data by multivariate tree regression

Dan Stowell and Mark D. Plumbley

October 2010

Abstract

Applications such as concatenative synthesis (audio mosaicing) and query-by-example require the ability to search a database using a sound which is qualitatively different from the actual desired result – for example when using vocal queries to retrieve nonvocal sound. Standard query techniques such as nearest neighbours do not account for this difference between source and target; they perform retrieval but do not learn to make timbral analogies. This paper addresses this issue by considering timbral query as a multivariate regression problem from one timbre distribution onto another. We develop a novel variant of multivariate tree regression: given only a set of unlabelled and unpaired samples from two distributions on the same space, the regression learns a cross-associative mapping which assumes general similarities in structure of the two distributions, yet can accommodate differences in shape at various scales. We demonstrate the technique with a synthetic example and with a concatenative synthesiser.

1 Introduction

Many musical applications of machine learning can be described as automatic classification tasks, where the class labels may for example indicate genre, key or instrumentation [Orio, 2006]. Less attention has been paid to regression-type tasks in music processing, yet they too may be facilitated or automated by machine learning. In this paper we consider one such task: automatically inferring timbral analogies from one type of music audio to another, where timbre is treated as a multivariate continuous attribute, and to infer a timbral analogy means to take a timbre value (or series of values) in one context and

infer which value(s) from some other context are the best match, a decision influenced by differences between the contexts.

Timbral analogies are needed in systems which use one type of sound source to search a database of some other type of sound source, such as query-by-humming [Wold et al., 1996] or concatenative synthesis from templates (audio mosaicing) [Schwarz, 2004, Part III][Sturm, 2006]. The basic problem is to map a point drawn from one timbral distribution onto its corresponding location in another distribution, accounting for global differences in the two timbre distributions.

In this paper we first consider the question of why analogies might be needed in the timbral domain, and discuss current approaches to timbre-based search in this light. We then consider the search for timbral analogies as a regression problem, and develop a variant of a non-linear multivariate regression technique for our purpose. We apply this technique to a small synthetic example in order to demonstrate the algorithm's performance, and evaluate in a more realistic situation through a concantenative synthesis of timbral audio samples.

2 Timbre search

In order to map the timbre of one sound onto that of another, we will require a timbre analysis of the signal. An issue that affects our choice of search strategy is whether the timbral analysis should best be treated as absolute and context-independent, or whether it should be treated as relative – for example, relative to the range of the sound source which produced it. Given a particular timbral "coordinate", should we treat it differently if we knew that it was produced by a clarinet or by a violin? Would such information imply a difference in the expressive purpose of the sound?

The common definition of timbre describes it as that attribute which enables a listener to differentiate sounds which are equal in pitch and loudness [ANSI, 1960]. It therefore does not demand that timbre be an absolute or context-invariant attribute of a sound. Research into music timbre perception has taken a similar stance, basing experiments on comparisons among sets of sound examples [Grey and Gordon, 1978, McAdams et al., 1995, Caclin et al., 2005, Burgoyne and McAdams, 2009]. Such studies often explain results in part through acoustic features derived from the examples, which can imply a context-independent notion of timbre inherent in the signal. However Grey

[1978] finds evidence for context-dependence of timbre perception in musical patterns. Lakatos [2000] offers some consideration of contextual effects by investigating sets of harmonic and percussive sounds both separately and combined. He presents evidence supporting the existence of two broadly context-independent timbre dimensions but also for some degree of contextual influence on timbre judgements.

Musical applications of timbre analysis often use acoustic features taken from the signal (e.g. Aucouturier and Pachet [2004], Schwarz [2004, Chapter 16]), implicitly treating timbre as absolute. This will certainly be appropriate in situations where the timbre data contains strong semantic "anchors" – a clear example of this occurs in human speech, where vowels are largely characterised by the absolute positions of the main resonances (formants) on the frequency scale [Deterding, 1997]. However, the evidence of context-dependence in musical timbre suggests this may not always be the case. Consider a system which synthesises or retrieves sounds based on timbral examples produced by voice (e.g. Schwarz [2004, Part III]): the human voice is naturally constrained to its own timbre range, yet we may well wish to induce the system to produce sounds outside this range. In fact we consider this to be a basic requirement, since such ability to extend our timbral range is one of the main appeals of such technologies.

2.1 Timbre lookup strategies

The most basic form of timbral search is perhaps a nearest-neighbour (NN) search, often using Euclidean distance. Since timbre features in general have quite different ranges, their ranges may be standardised before search, or a scale-invariant metric such as Mahalanobis distance may be used [Wouters and Macon, 1998]. For example, Schwarz [Schwarz, 2004, Chapter 16] uses the Euclidean distance normalised over the entire database of sounds. This normalisation accounts for differences between the ranges of the features, but not for differences between the timbral range of the different sound sources included in the database. Note that timbral distance search is but one criterion used in a concatenative synthesiser such as this, which uses a constraint-satisfaction framework to combine criteria related to duration, pitch and other considerations.

Large database search systems often do not store the raw timbral co-ordinates needed for NN search, but parametrically model the timbre of a recording

(e.g. using Gaussian Mixture Models) and store the model parameters [Aucouturier and Pachet, 2004]. Timbral search can then be performed by finding the parameter-set which maximises the likelihood of query data.

Whether search is performed by instance-based methods such as NN or model-based methods such as Gaussian Mixture Model likelihood, the difference in timbral ranges of different sound sources is often neglected, reflecting an approach to timbre as absolute rather than relative. One way to accommodate for some differences could be to standardise the mean and variance of timbre features separately for each type of sound source, or for each recorded audio excerpt, which would accommodate the large-scale differences. However it would fail to account properly for multidimensional interactions in the data such as the movement of one region relative to the rest of the distribution.

Rather than pursuing the idea of a normalisation scheme as a precursor to search, in this paper we develop an integrated method which automatically learns to map from one data distribution to another, assuming similarities in the orientation of the datasets in timbre space but allowing for differences in the distributions at large and small scales. Tree methods are attractive in this context because recursive partitioning provides a generic approach to dividing multidimensional distributions into regions of interest at multiple scales. We next describe the method, before applying it in two experiments which will illustrate its usefulness for timbral queries.

3 Multivariate regression trees

The framework of Classification And Regression Trees (CART) [Breiman et al., 1984] was developed as a computationally efficient nonparametric way to analyse structure in a multivariate dataset, with a class label or a continuous-valued response to be predicted by the independent variables. The core concept is to recursively partition the dataset, at each step splitting it into two subsets using a threshold on one of the independent variables (i.e. a splitting hyperplane orthogonal to one axis). The choice of split at each step is made to minimise an "impurity" criterion for the value of the response variable in the subsets. When the full tree has been grown it is likely to overfit the distribution, so it is then pruned by merging branches according to a cross-validation criterion to produce an optimally-sized tree.

CART methods have found application in a variety of disciplines and have

spawned many variants [Murthy, 1998]. Classification trees are perhaps more commonly used than regression trees; here we focus on the latter. Note that tree-based methods are not restricted to datasets with an underlying hierarchical structure, rather they provide an efficient approach to general nonparametric modelling of the variation and structure within a dataset.

The standard CART is univariate in two senses: at each step only one variable is used to define the splitting threshold; and the response variable is univariate. The term "multivariate" has been used in the literature to refer to variants which are multivariate in one or other of these senses: for example Questier et al. [2005] regress a multivariate response variable, while Brodley and Utgoff [1995] use multivariate splits in constructing a classification tree; Gama [2004] considers both types of multivariate extension. In the following we will refer to "multivariate-response" or "multivariate-splits" variants as appropriate. Multivariate-splits variants can produce trees with reduced error, although the trees will usually be harder to interpret since the splitting planes are more conceptually complex.

We next consider a particular type of regression tree which was proposed for the unsupervised case, i.e. it does not learn to predict a class label or response variable, rather the structure in the data itself. We will extend this tree to include multivariate splits, before considering the cross-associative case.

3.1 Auto-associative MRT

Regression trees are studied in a feature-selection context by Questier et al. [2005], including their application in the unsupervised case, where there is no response variable for the independent variables to predict. The authors propose in that case to use the independent variables also as the response variables, yielding a regression tree task with a multivariate response which will learn the structure in the dataset. In their feature-selection application, this allows them to produce an estimate of the variables that are "most responsible" for the structure in the dataset. However the strategy is quite general and could allow for regression trees to be used on unlabelled data for a variety of purposes. It is related to other data-dependent recursive partitioning schemes, used for example in estimation of densities [Lugosi and Nobel, 1996] or information-theoretic quantities [Stowell and Plumbley, 2009].

3.1.1 Splitting criterion

In constructing a regression tree, a choice of split must be made at each step. The split is chosen which minimises the sum of the "impurity" of the two resulting subsets, typically represented by the mean squared error [Breiman et al., 1984, Section 8.3]. For multivariate responses this is:

$$\text{impurity}(\alpha) = \sum_{i=1}^{n_\alpha} \sum_{j=1}^{p} (y_{ij} - \bar{y}_j)^2 \tag{1}$$

where n_α is the number of data points in the subset α under consideration, and $\bar{y}$ is the mean of the p-dimensional response variable y_i for the points in α. In the auto-associative case the y_{ij} are the same as the x_{ij}, the variables by which the splitting planes will be defined.

The impurity measure (1) is equivalent to the sum of variances in the subsets, up to a multiplication factor which we can disregard for the purposes of minimisation. By the law of total variance (see e.g. Searle et al. [2006, Appendix S]), minimising the total variance within the subsets is the same as maximising the variance of the centroids; therefore the impurity criterion selects the split which gives the largest difference of the centroids of the response variable in the resulting subsets.

In the feature-selection task of Questier et al. [2005] splits are univariate: each splitting plane is perpendicular to one axis. However, we are not performing feature-selection but characterising the data distributions; as explored by Gama [2004] it may be advantageous to allow multivariate splits to reduce error. We therefore extend the AAMRT approach by allowing multivariate splits. Since the hyperplane which splits a dataset into two subsets with the furthest-separated centroids is simply the hyperplane perpendicular to the first principal component in the centred data, the multivariate-splits AAMRT is implemented simply by using the first principal component to define the splitting plane.

Partitioning using the first principal component has been considered by previous authors such as Boley [1998]. It allows for efficient implementation since the leading principal component in a dataset can be calculated quickly e.g. by expectation-maximisation. We next introduce a novel extension of this approach specifically for the task of learning analogies between two datasets.

3.2 Cross-associative MRT

Auto-associative MRT may be useful for discovering structure in an unlabelled dataset [Questier et al., 2005]. Here we wish to adapt it such that it can be used to analyse structural commonalities between two unlabelled datasets, and learn associations between the two. Therefore we now develop a variant that is cross-associative rather than auto-associative; we will refer to it as cross-associative MRT or *XAMRT*.

Our assumptions will be that the two datasets are i.i.d. samples from two distributions which have broad commonalities in structure and orientation in the measurement space, but that there may be differences in location of regions between the distributions. These may be broad differences such as the location (centroid) or dispersion (variance) along one or many dimensions, or smaller-scale differences such as the movement of a small region of the distribution relative to the rest of the distribution. These assumptions are relevant for timbral datasets as will be illustrated in Section 4.

The AAMRT approach is adaptable to the case of two data distributions simply by considering the distributions simultaneously while partitioning. In our scheme with multivariate splits this means determining the splitting plane using the principal component of the concatenation of the datasets (or of subsets therefrom). However, given that we allow the two distributions to have differences in location we perform centring *separately* on each distribution, before combining them for the purpose of finding a common principal component. We perform this centring at each level of the recursion, which creates an algorithm which allows for differences in location both overall and in smaller subregions of the distributions. This is illustrated schematically in Figure 1.

[Figure 1 about here.]

If the datasets contain unequal numbers of data points then the larger set will tend to dominate over the smaller in calculating the principal component. To eliminate this issue we weight the calculation so as to give equal emphasis to each of the datasets, equivalent to finding the principal component of the concatenation J of weighted datasets:

$$J = [n_Y(X - C_X), n_X(Y - C_Y)] \tag{2}$$

where X and Y represent the data (sub)sets, C_X and C_Y their centroids, and n_X and n_Y the number of points they contain.

By recursively partitioning in this way, the two datasets are simultaneously partitioned in a way that reflects both the general commonalities in structure (using splitting hyperplanes with a common orientation) and their differences in location (the position of the hyperplanes, passing through the centroids of subsets of each dataset). The tree structure defines two different partitions of the space, approximating the densities of the two distributions, and pairing regions of the two distributions.

The tree thus produced is similar to a standard (i.e. neither auto-associative nor cross-associative) multivariate-response regression tree, in that it can predict a multivariate response from multivariate input. For example an input vector x_i can be classified into a node, and the corresponding expected value $\hat{y}_i$ given as the centroid of the Y data associated with that node. However the tree treats the two distributions symmetrically, allowing projection from either dataset onto the other. Unlike the AAMRT it does not require the input data to be the same as the response data.

3.2.1 Pruning criterion

Allowing a regression tree to proceed to the maximum level of partitioning will tend to overfit the dataset. Criteria may be used to terminate branching, but a generally better strategy (although more computationally intensive) is to grow the full tree and then prune it back by merging together branches [Breiman et al., 1984, Chapter 3]. In the CART framework, the standard measure for pruning both classification and regression trees is *crossvalidation error* within a branch: a normalised average over all datapoints of the error that results from estimating the label of each datum from the other data labels [Breiman et al., 1984, Chapters 3 and 8]. Branches which exhibit high crossvalidation error are merged into leaf nodes, so as to improve the stability and generality of the tree.

In our case this approach cannot be applied directly because we consider the unsupervised case, i.e. without labels. In Questier et al. [2005] the unlabelled data are used to predict themselves, meaning that the tree algorithm does in fact see (multivariate) labels attached to the data and the crossvalidation measure can be used. We wish to associate two separate distributions whose data points are not paired, and so such a strategy is not available to us.

Instead, we propose to apply the crossvalidation principle to the splitting hyperplanes themselves, producing a measure of the "stability" of a multivariate split with respect to the sampled data. This would penalise splitting hyperplanes

which were only weakly justified by the data, and so produce a pruned tree whose splits were relatively robust to outliers and noise. Our crossvalidation measure is calculated using a leave-one-out ("jackknife") procedure as follows: given a set of n_α data points whose first principal component p_α has been calculated to give the proposed splitting plane, we calculate

$$R_\alpha = \frac{1}{n_\alpha} \sum_{i=1}^{n_\alpha} \mathrm{abs}(p_\alpha \cdot \hat{p}_{\alpha i}) \tag{3}$$

where $\hat{p}_{\alpha i}$ is the first principal component calculated after excluding datum i. The abs has been introduced so that our measure considers the orientation but not the direction of the principal component vectors (a measured principal component may be flipped by 180° yet define the same splitting hyperplane; cf. Gaile and Burt [1980]). Both p_α and $\hat{p}_{\alpha i}$ are unit vectors, so R is the average cosine distance between the principal component and its jackknife estimates.

As with the standard CART, we then simply apply a threshold, merging a given branch if its value of R is below some fixed value. Our measure ranges between 0 and 1, where 1 is perfect stability (meaning the principal component is unchanged when any one data point is excluded from the calculation). In this work we use manually-specified thresholds when applying our algorithm, as in CART. Alternatively one could derive thresholds from explicit hypothesis tests by modelling the distribution of the jackknife principal components on the hypersphere [Figueiredo, 2007].

3.2.2 Summary of algorithm

The algorithm is summarised as pseudocode in Figure 2. Given two datasets X and Y, both taking values in $\mathcal{X} = \mathbb{R}^D$, the recursive function GROW creates the regression tree from X and Y, and the recursive function PRUNE prunes the tree given a user-specified stability threshold. An open-source implementation of the algorithm in Python is available.[1]

[Figure 2 about here.]

[1] `http://www.elec.qmul.ac.uk/digitalmusic/downloads/xamrt/`

4 Experiments

In this section we present two experiments exploring the application of the proposed algorithm to audio examples. The ultimate evaluation of musical synthesis techniques should typically include user listening tests and the like; our focus here is on the behaviour of the algorithm in comparison against standard techniques, for which it is particularly helpful to use numerical and graphical analysis of the output of different mapping techniques. Hence we focus on objective measures.

We first introduce the evaluation measure we will use, before presenting the two experiments.

4.1 Evaluation measure

It is natural to expect that a good mapping will produce a good coverage of the timbre distribution onto which we are mapping. For example, in concatenative synthesis this means making wide use of the "alphabet" of available sound grains, so as to generate a rich as possible output from the limited alphabet, avoiding too much repetition of grains. Here we develop this notion into an information-theoretic evaluation measure.

Communication through finite discrete alphabets has been well studied in information theory [Arndt, 2001]. A key information-theoretic quantity is the (Shannon) *entropy*, defined for a discrete random variable X taking values from an alphabet $\mathcal{A}$ as

$$H(X) = -\sum_{i=1}^{|\mathcal{A}|} p_i \log p_i \tag{4}$$

where p_i is the probability that $X = \mathcal{A}_i$ and $|\mathcal{A}|$ is the number of elements in $\mathcal{A}$. The entropy $H(X)$ is a measure of the information content of X, and has the range

$$0 \leq H(X) \leq \log |\mathcal{A}| \tag{5}$$

with the maximum achieved iff X is uniformly distributed.

If the alphabet size is known then we can define a normalised version of the entropy called the *efficiency*

$$\text{Efficiency}(X) = \frac{H(X)}{\log |\mathcal{A}|} \tag{6}$$

which indicates the information content relative to some optimised alphabet giving a uniform distribution. This can be used for example when X is a quantisation of a continuous variable, indicating the appropriateness of the quantisation scheme to the data distribution.

We can apply such an analysis to a unit-selection system such as concatenative synthesis, since it fits straightforwardly into this framework: timbral expression is measured using a set of continuous acoustic features, and then "quantised" by selecting one grain from an alphabet to be output. It does not deductively follow that a scheme which produces a higher entropy produces the most pleasing audio results. However, a scheme which produces a low entropy will tend to be one which has an uneven probability distribution over the grains, and therefore is likely to sound relatively impoverished—for example, some grains will tend to be repeated more often than in a high-entropy scheme. Therefore the efficiency measure is useful in combination with the resynthesised audio results for evaluating the efficacy of a grain selection scheme.

[Figure 3 about here.]

4.2 Experiment 1: Synthetic tones

Our first experiment uses two synthetic test tones, for the purpose of comparing the behaviour of techniques applied to data with a specific known mapping. We designed two synthetic sound recordings which each represented a trajectory through timbre space over 20 seconds. In order to enable visual and numerical assessment of the quality of the analysis, the timbral trajectories were designed to be piecewise linear and monotonic – in fact strictly increasing along each of the timbral axes – yet different in each of the two recordings. Thus the expected mapping from one to the other would be a mapping which matched units in the same time-order as they were generated.

The synthesisers used timbrally different sources – one a saw wave and one a square wave – each with a control for pitch and a control for the depth of modulation by white noise, thus allowing to vary smoothly between a purely harmonic tone and a noisy tone. The two sound recordings can be heard online.[1]

The two sound recordings were analysed into a 3D representation using standard measures of pitch and timbre: an autocorrelation-based pitch tracker [McLeod and Wyvill, 2005], the *spectral flatness* measure of signal-to-noise ratio and the *spectral centroid* measure of timbral brightness [Peeters, 2004]. Audio was generated at 44.1 kHz, and analysis was performed in frames of size 1024

with 50% overlap. Hann windowing was applied to audio frames before FFT analysis. The features were multiplied by fixed ratios to standardise their variances to similar ranges.

The 3D analysis co-ordinates from the two signals are depicted in Figure 3. The designed timbral trajectories are evident in the figure, but with some measurement noise which manifests as a broadening of the main trajectory trace.

Given the two sequences of 3D co-ordinates, we then performed a remapping from one signal to another: for each co-ordinate measured from one signal, we performed a query to retrieve the corresponding co-ordinate in the other signal, and stored the temporal index of the result. This was performed to map the saw-wave signal onto the square-wave signal, and vice versa, for three different types of search:

- Nearest neighbour (NN),
- Nearest neighbour (NN) after normalising the mean and variance within each of the two sets of co-cordinates,
- and our XAMRT algorithm with a pruning threshold of 0.99.

This gives a total of six remapping tasks. We did not perform any resynthesis in this first experiment.

Note that the time-series ordering information is not available to the search algorithms, but is used for evaluation. Since the signals were designed to be the same length and have monotonically increasing values for the co-ordinates, the most desirable mapping is a simple identity mapping that recovers the same time sequence as the input. The pure identity mapping is unlikely to be retrieved in this test because of the measurement noise in the acoustic features, but the algorithms can be compared to determine how close each of them is to this ideal.

[Figure 4 about here.]

[Table 1 about here.]

Results are depicted in Figure 4 and Table 1. All of the remapping algorithms produce a strong correlation between input and output indices – the normalised version of the NN search performing better than the non-normalised version, but the XAMRT search yielding a stronger correlation than either of them. However, the plots demonstrate some important features of the mappings which are not evident from the correlation values alone, and demonstrate visually

why the efficiency results show a stronger advantage for our method than is revealed by the correlations. The plots for both forms of nearest-neighbour lookup appear strongly striated, indicating that significant clusters of query indices are being mapped to the same target index. Conversely, many gaps are visible, indicating target indices which are never returned from the query. The plots for the XAMRT mapping show no such striation or gaps, offering a much smoother mapping for these data.

These results indicate that the XAMRT algorithm does perform as intended, learning to draw analogies from one dataset to another by adapting to the distributions taken by the two trajectories; and further, learning a *rich* mapping which selects widely from the available data points, rather than neglecting large selections of points as do the NN searches.

4.3 Experiment 2: Concatenative synthesis

Our second application example concerns concatenative synthesis or audio mosaicing, in which the timbral trajectory of one sound can be used to create new musical sequences by concatenating appropriately-selected segments of existing recordings. These brief segments (on the order of 100 ms duration, henceforth called *grains*) are stored in large numbers in a database and are not individually annotated.

As discussed earlier, the timbre features measured on the controlling signal and on the source material may well often occupy different regions of the timbral space since they have different timbral ranges (see also Figure 5). Range normalisation could be used to align the source and target timbre spaces, but would be unable to account for differences in the shapes of the distributions, and so is only a partial solution. We propose that our method could be used in such systems to perform the timbral search in a way which takes account of the differences in timbral distributions, as was already demonstrated for the synthetic examples in Section 4.2.

Concatenative synthesisers typically operate not only on timbre, but use pitch and duration as well as temporal continuity constraints in their search strategy, and then modify the selected grains further to improve the match [Maestre et al., 2009]. While recognising the importance of these aspects in a full concatenative synthesis system, we designed an experiment in which the role of pitch, duration and temporal continuity were minimised, by excluding such factors from grain construction/analysis/resynthesis, and also by selecting

audio excerpts whose variation is primarily timbral. In a full concatenative synthesiser it may be desirable to use strongly-pruned trees which would return a large number of candidate grains for a timbral query, and then to apply other criteria to select among the candidates; we leave this for future work.

We first describe the audio excerpts we used and how timbre was analysed, then the concatenative synthesiser. We then give results of the information-theoretic evaluation. Note that graphical/correlation-based analysis is not appropriate here (as it was in the previous experiment) because the timbre trajectories are not manually designed, so we cannot specify an expected sequence for the units chosen.

4.3.1 Audio data

[Table 2 about here.]

In order to focus on the timbral aspect, we selected a set of audio excerpts in which the interesting variation is primarily timbral and pitch is less relevant. The five excerpts—two musical (percussive) and three non-musical—are listed in Table 2 and are also available online.[1] The excerpts are 44.1 kHz mono recordings. This dataset is small, but as we will see is sufficient to yield statistically significant results in this case.

The excerpts are quite heterogeneous, in both sound source and duration (some differ by an order of magnitude). They each contain various amounts/types of audio event.

[Figure 5 about here.]

4.3.2 Timbre features

For this experiment we chose a set of 10 common acoustic timbre features: spectral power, spectral power ratio in 5 log-spaced subbands (50–400, 400–800, 800–1600, 1600–3200, and 3200–6400 Hz), spectral centroid, spectral 95- and 25-percentiles and zero-crossing rate (for definitions see Peeters [2004]). This is a richer timbral analysis than in the synthetic experiment of Section 4.2, and more reflective of the type of timbre space that might be used in a concatenative synthesiser.

Analysis was performed on audio grains of fixed 100ms duration taken from the audio excerpt every 100ms (i.e. with no overlap). Each grain was analysed by segmenting into frames of 1024 samples (at 44.1 kHz sampling rate) with

50% overlap, then measuring the feature values for each frame and recording the mean value of each feature for the grain. Grains with a very low spectral power (< 0.002) were treated as silences and discarded. The timbre features of the remaining grains were normalised to zero mean and unit variance within each excerpt. Analysis was performed in SuperCollider 3.3.1 [McCartney, 2002].

Figure 5 plots a PCA projection of the grain timbre data for two of the sound excerpts.

4.3.3 Timbral concatenative synthesiser

We designed a simple concatenative synthesiser using only timbral matching, either by a nearest-neighbour (NN) search (after normalising features by mean and variance) or by our XAMRT algorithm. Given two excerpts—one which is the source of grains to be played back, and one which is the control excerpt determining the order of playback—and the timbral metadata for the grains in the two excerpts, the synthesis procedure works as follows:

For each grain in the control excerpt, if the grain is silent (power < 0.002) then we replace it with silence. Otherwise we replace it with a grain selected from the other excerpt by performing a lookup of the timbre features—either a NN search or the XAMRT tree regression (without pruning). For numerical evaluation, the choice of grain is recorded. For audio resynthesis, the new set of grains is output with a 50ms linear crossfade between grains.

4.3.4 Results

[Table 3 about here.]

We applied the concatenative synthesis of Section 4.3.3 to each of the 20 pairwise combinations of the 5 audio excerpts (excluding self-to-self combinations, which are always 100% efficient) using each of the two lookup methods (NN and XAMRT). We then measured the information-theoretic efficiency (6) of each run. Table 3 summarises the efficiencies for each lookup method. Our method is seen to be significantly better than the normalised NN search, improving efficiency by over 13 percentage points.

Audio examples of the output from the system are available online.[1] Note that the reconstructed audio examples sound rather unnatural because the experiment is not conducted in a full concatenative synthesis framework. In particular we use a uniform grain duration of 100ms and impose no temporal

constraints, whereas a full concatenative synthesis system typically segments sounds using detected onsets and includes temporal constraints for continuity, and therefore is able to synthesise much more natural attack/sustain dynamics [Maestre et al., 2009]. Our method shows promise as the timbral component of a multi-attribute search which could potentially be used in concatenative synthesis, as well as other applications requiring timbral search from audio examples (e.g. query-by-example [Wold et al., 1996]).

5 Conclusions

We have introduced an unsupervised algorithm which is able to learn associations between two unlabelled datasets, on the assumption that the underlying distributions have some common structure. The purpose is to address a specific issue arising from timbral database queries: the need for systems to learn to make *analogies* when the query is qualitatively different from the intended target sound. Our algorithm is a variant on the multivariate regression tree, and has a simple implementation with a novel pruning criterion based on the stability of the multivariate splits. We have presented two experiments which demonstrate the benefits of our algorithm over nearest-neighbour-type searches.

As future work we intend to incorporate the XAMRT algorithm into a full concatenative synthesis framework, and perform subjective evaluations including listening tests. We are also investigating other application domains for the XAMRT algorithm, e.g. comparing vowel formants in corpora of different speakers. More broadly, we hope to have motivated the need to be able to search for analogies in timbral queries, and look forward to further developments in this area.

Acknowledgments

DS is supported by the EPSRC under a Doctoral Training Account studentship and grant EP/I001832/1. MP is supported by an EPSRC Leadership Fellowship (EP/G007144/1).

References

ANSI. *Acoustical Terminology.* Number S1.1-1960. American National Stan-

dards Institute, New York, 1960.

C. Arndt. *Information Measures*. Springer, 2001.

J.-J. Aucouturier and F. Pachet. Improving timbre similarity: how high's the sky? *Journal of Negative Results in Speech and Audio Sciences*, 1(1), 2004.

D. Boley. Principal direction divisive partitioning. *Data mining and knowledge discovery*, 2(4):325–344, 1998. doi: 10.1023/A:1009740529316.

L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth Statistics/Probability Series. Wadsworth Inc, 1984.

C. E. Brodley and P. E. Utgoff. Multivariate decision trees. *Machine Learning*, 19(1):45–77, 1995. doi: 10.1023/A:1022607123649.

J. A. Burgoyne and S. McAdams. A meta-analysis of timbre perception using nonlinear extensions to CLASCAL. In R. Kronland-Martinet, S. Ystad, and K. Jensen, editors, *Sense of Sounds*, volume 4969/2009 of *Lecture Notes in Computer Science*, chapter 12, pages 181–202. Springer, Berlin, 2009. doi: 10.1007/978-3-540-85035-9_12.

A. Caclin, S. McAdams, B. K. Smith, and S. Winsberg. Acoustic correlates of timbre space dimensions: a confirmatory study using synthetic tones. *Journal of the Acoustical Society of America*, 118(1):471–482, 2005. doi: 10.1121/1.1929229.

D. Deterding. The formants of monophthong vowels in Standard Southern British English pronunciation. *Journal of the International Phonetic Association*, 27(1):47–55, 1997. doi: 10.1017/S0025100300005417.

A. Figueiredo. Comparison of tests of uniformity defined on the hypersphere. *Statistics and Probability Letters*, 77(3):329–334, 2007. doi: 10.1016/j.spl.2006.07.012.

G. L. Gaile and J. E. Burt. *Directional Statistics*, volume 25 of *Concepts and Techniques in Modern Geography*. Geo Abstracts Ltd., 1980.

J. Gama. Functional trees. *Machine Learning*, 55(3):219–250, 2004. doi: 10.1023/B:MACH.0000027782.67192.13.

J. M. Grey. Timbre discrimination in musical patterns. *Journal of the Acoustical Society of America*, 64(2):467–472, 1978. doi: 10.1121/1.382018.

J. M. Grey and J. W. Gordon. Perceptual effects of spectral modifications on musical timbres. *Journal of the Acoustical Society of America*, 63(5):1493–1500, 1978. doi: 10.1121/1.381843.

S. Lakatos. A common perceptual space for harmonic and percussive timbres. *Perception & Psychophysics*, 62(7):1426–1439, 2000.

G. Lugosi and A. Nobel. Consistency of data-driven histogram methods for density estimation and classification. *Annals of Statistics*, 24(2):687–706, 1996. doi: 10.1214/aos/1032894460.

E. Maestre, R. Ramírez, S. Kersten, and X. Serra. Expressive concatenative synthesis by reusing samples from real performance recordings. *Computer Music Journal*, 33(4):23–42, 2009. doi: 10.1162/comj.2009.33.4.23.

S. McAdams, S. Winsberg, S. Donnadieu, G. De Soete, and J. Krimphoff. Perceptual scaling of synthesized musical timbres: common dimensions, specificities, and latent subject classes. *Psychological Research*, 58(3):177–192, 1995. doi: 10.1007/BF00419633.

J. McCartney. Rethinking the computer music language: SuperCollider. *Computer Music Journal*, 26(4):61–68, 2002. doi: 10.1162/014892602320991383.

P. McLeod and G. Wyvill. A smarter way to find pitch. In *Proceedings of the International Computer Music Conference (ICMC'05)*, pages 138–141, Barcelona, Spain, 2005.

S. K. Murthy. Automatic construction of decision trees from data: a multi-disciplinary survey. *Data Mining and Knowledge Discovery*, 2(4):345–389, 1998. doi: 10.1023/A:1009744630224.

N. Orio. Music retrieval: a tutorial and review. *Foundations and Trends in Information Retrieval*, 1(1):1–90, Nov 2006. doi: 10.1561/1500000002.

G. Peeters. A large set of audio features for sound description. Technical report, IRCAM, 2004.

F. Questier, R. Put, D. Coomans, B. Walczak, and Y. Vander Heyden. The use of CART and multivariate regression trees for supervised and unsupervised

feature selection. *Chemometrics and Intelligent Laboratory Systems*, 76(1): 45–54, 2005. ISSN 0169-7439. doi: 10.1016/j.chemolab.2004.09.003.

D. Schwarz. *Data-Driven Concatenative Sound Synthesis*. PhD thesis, IRCAM, Paris, France, Jan 2004. URL `http://recherche.ircam.fr/equipes/analyse-synthese/schwarz/thesis/`.

S. R. Searle, G. Casella, and C. E. McCulloch. *Variance Components*. Wiley-Interscience, online edition, 2006. ISBN 978-0470316856. doi: 10.1002/9780470316856.

D. Stowell and M. D. Plumbley. Fast multidimensional entropy estimation by k-d partitioning. *IEEE Signal Processing Letters*, 16(6):537–540, Jun 2009. doi: 10.1109/LSP.2009.2017346.

B. L. Sturm. Adaptive concatenative sound synthesis and its application to micromontage composition. *Computer Music Journal*, 30(4):46–66, 2006. doi: 10.1162/comj.2006.30.4.46.

E. Wold, T. Blum, D. Keislar, and J. Wheaten. Content-based classification, search, and retrieval of audio. *IEEE multimedia*, 3(3):27–36, 1996. doi: 10.1109/93.556537.

J. Wouters and M. W. Macon. A perceptual evaluation of distance measures for concatenative speech synthesis. In *Proceedings of ICSLP'98*, volume 6, pages 2747–2750, Sydney, Nov 1998. doi: 10.1109/ICASSP.2001.941045.

List of Figures

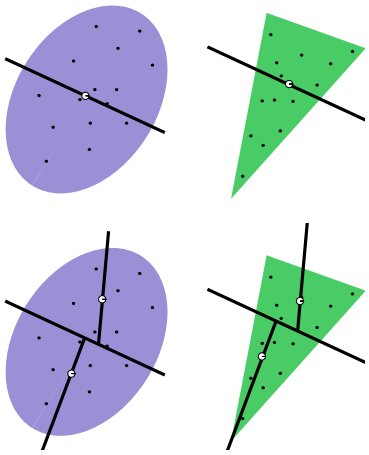


Figure 1: Schematic representation of the first two steps in the recursion. In the first step (top), the centroids of each dataset are calculated separately, and then a splitting plane with a common orientation is chosen. The second step (bottom) is the same but performed separately on each of the partitions produced in the first step.

GROW(X, Y)

$C_X \leftarrow$ centroid of X
$C_Y \leftarrow$ centroid of Y
$J \leftarrow$ result of equation (2)
$p \leftarrow$ principal component of J
$X_l \leftarrow X \cap ((X - C_X) \cdot p > 0)$
$X_r \leftarrow X \cap ((X - C_X) \cdot p \leq 0)$
$Y_l \leftarrow Y \cap ((Y - C_Y) \cdot p > 0)$
$Y_r \leftarrow Y \cap ((Y - C_Y) \cdot p \leq 0)$
if X_l is singular or Y_l is singular
  then $L = [X_l, Y_l]$
  else $L =$ GROW(X_l, Y_l)
if X_r is singular or Y_r is singular
  then $R = [X_r, Y_r]$
  else $R =$ GROW(X_r, Y_r)
return [L, R]

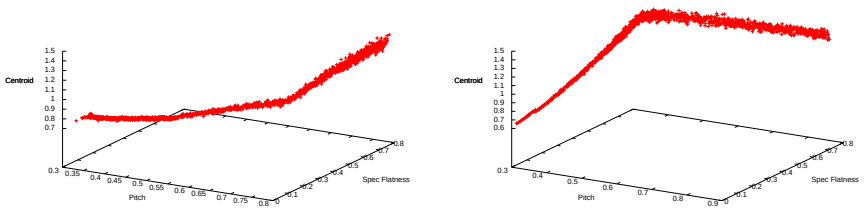
PRUNE(*tree*, *threshold*)

PRUNE(left child, *threshold*)
PRUNE(right child, *threshold*)
if children of left child are both leaf nodes
  then PRUNEONE(left child, *threshold*)
if children of right child are both leaf nodes
  then PRUNEONE(right child, *threshold*)

PRUNEONE(*tree*, *threshold*)

$R \leftarrow$ result of equation (3)
if $R <$ *threshold*
  then merge child nodes into a single node

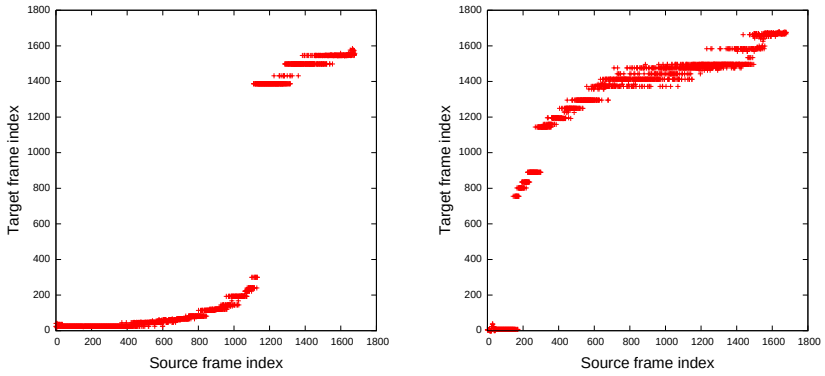
Figure 2: The XAMRT algorithm. X and Y are the two sets of vectors between which associations will be inferred.



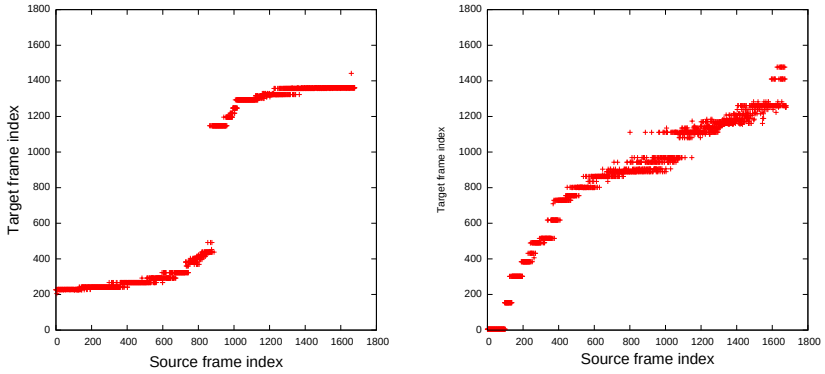
(a) Saw-wave signal

(b) Square-wave signal

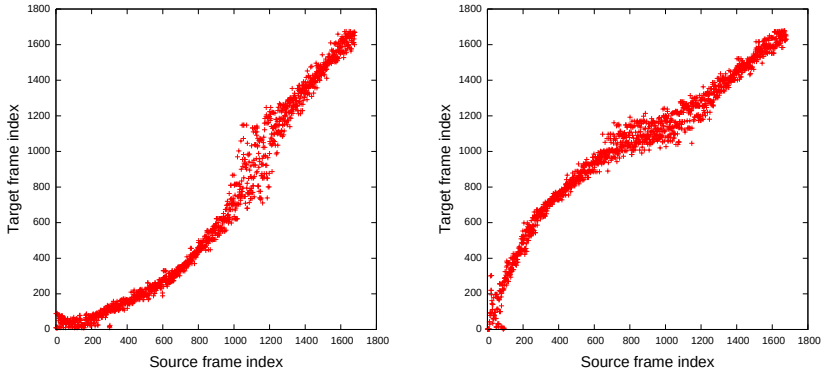
Figure 3: 3D pitch and timbre features measured on the two 20-second synthesised sounds after segmenting into 1024-sample frames. Axes are variance-normalised, hence units are not given.



(a) Nearest neighbour (NN)



(b) Nearest neighbour (NN), normalised



(c) XAMRT

Figure 4: A-to-B index mappings for a pair of synthesised signals. For each mapping technique, two plots are shown: one mapping the saw-wave signal to the square-wave signal (left), and one doing the reverse (right). Because the synthetic signals both have a monotonically increasing timbral progression through time, the ideal mapping here is a smooth and unbroken monotonically increasing mapping.

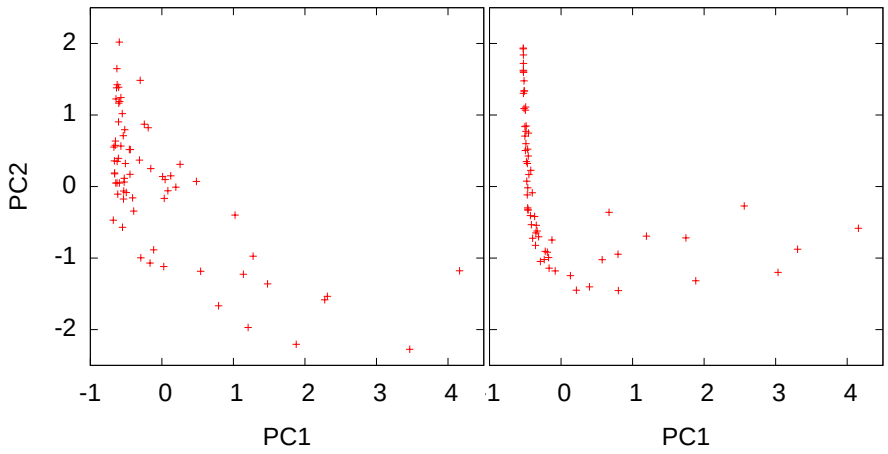


Figure 5: Two-dimensional PCA projections of timbre co-ordinates derived from analysis of the *Amen breakbeat* (left) and *thunder* (right) sound excerpts (described in Section 4.3.1). The timbre distributions have broad similarities in structure as well as differences: both show a non-linear interaction between the two axes yielding a curved profile; yet the lower plot exhibits a sharper bend and a narrower distribution in the upper-left region. The projection was calculated by applying PCA to the balanced concatenation of the separately-standardised datasets (Equation (2)).

List of Tables

	Correlation	Efficiency (%)
Saw→square		
NN	0.86	36.6
NN, normalised	0.91	37.9
XAMRT	0.97	83.0
Square→saw		
NN	0.80	41.5
NN, normalised	0.94	47.3
XAMRT	0.97	83.7

Table 1: Pearson correlations and efficiencies for the synthetic sound experiment, corresponding to the results of Figure 4.

Description	Duration (sec)	No. of grains
Amen breakbeat	7	69
Beatboxing	93	882
Fireworks	16	163
Kitchen sounds	49	355
Thunder	8	65

Table 2: Audio excerpts used in timbre experiment. "No. of grains" is the number of 100ms grains segmented and analysed from the audio (excluding silent frames)—see text for details.

Query type	Efficiency (%)
Nearest neighbour	70.8 ± 4.4
XAMRT	84.5 ± 4.8

Table 3: Experimental values for the information-theoretic efficiency of the lookup methods. Means and 95% confidence intervals are given. The improvement is significant at the $p < 0.000001$ level (paired t-test, two-tailed, 19 degrees of freedom, $t = 12.47$).