Feature Based Classification of Astronomical Transients

PETER ASHWELL
SID: 307168697

Supervisor: Dr. Tara Murphy

This thesis is submitted in partial fulfillment of
the requirements for the degree of
Bachelor of Information Technology (Honours)

School of Information Technologies
The University of Sydney
Australia

8 November 2011

Student Plagiarism: Compliance Statement

I certify that:

I have read and understood the University of Sydney Student Plagiarism: Coursework Policy and Procedure;

I understand that failure to comply with the Student Plagiarism: Coursework Policy and Procedure can lead to the University commencing proceedings against me for potential student misconduct under Chapter 8 of the University of Sydney By-Law 1999 (as amended);

This Work is substantially my own, and to the extent that any part of this Work is not my own I have indicated that it is not my own by Acknowledging the Source of that part or those parts of the Work.

Name: Peter Ashwell

Signature: **Date**:

Abstract

Transient classification is the problem of identifying and classifying temporary phenomena in astronomical data. These phenomena are caused by extreme physical processes such as supernova explosions unfolding in the Universe. Hence, automating transient discovery and classification has great scientific value for astronomers. This thesis contributes in a number of areas to solving the problem of the early classification of transient events from noisy and sparsely sampled astronomical data. In it I frame the problem of astronomical transient classification in the computer science field of time series classification. I also implement a classifier using the Random Forest supervised classification algorithm with wavelet transforms, and statistical properties and shapelet representations of time series as features. I assess the effectiveness of this classifier on a number of simulated transient classes with singular and combined distortions such as noise, missing data and power law distributions applied. The results of this evaluation demonstrate that supervised classification holds some promise for solving the transient classification problem but is not yet suitable for the VAST pipeline in terms of classification. I propose using data preprocessing techniques such as regression and noise smoothing and filtering to improve classification accuracy.

Acknowledgements

Thanks first to my family for providing me the opportunity to study Computer Science. Thanks also to Kitty for some stimulating discussions throughout the year and for producing the model transients I used in my experiments. Many thanks to Tara for her diligent supervision and encouragement. And finally thanks to Cat for her love and support and for patiently awaiting my return from the voyage of thesis writing.

CONTENTS

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Problem Context

Astronomy is by nature an observational, data-driven science. During the last few decades the field was transformed by technology with software and databases becoming an integral part of data gathering and analysis. As the sophistication of observational equipment has improved the volume of observational data available to researchers has increased. The ASKAP[1] telescope array under construction in Western Australia will produce a gigabyte of processed data per second . A topic of growing interest has been to automate the analysis of these large data volumes. The subject of this research is to develop algorithms for use in an automatic data analysis pipeline for detecting astronomical transient events as part of the VAST[2] project.

1.2 Transients and Time Series

An astronomical transient event is a structural change in the observation of a stellar object such as a star or galaxy, referred to as *sources* by astronomers. These observations take the form of *time series*, or to astronomers, *light curves*. Time series represent the intensity of an observed source in potentially multiple frequencies over some time indices. An explicit definition for time series is:

$$\{(t_i, \mathbf{x}_i) \quad i \in \{t_1 \dots t_D\} \subset \mathbb{R} \quad \mathbf{x}_i \in \mathbb{R}^D\}$$

a mapping from unique time indices t_i to a D dimensional vector $\mathbf{x}_i$. The t_i values may represent any real unit of time - seconds, days, years. The sequence of increasing t_i is not necessarily evenly distributed. A typical astronomical time series we are looking for is a supernova. A supernova occurs when very large

[1] http://www.atnf.csiro.au/SKA/
[2] http://www.physics.usyd.edu.au/sifa/vast/index.php

stars reach the end of their life cycle resulting in a catastrophic explosion. These explosions release large amounts of radiation that can be detected from great distances. Variability in astronomical time series

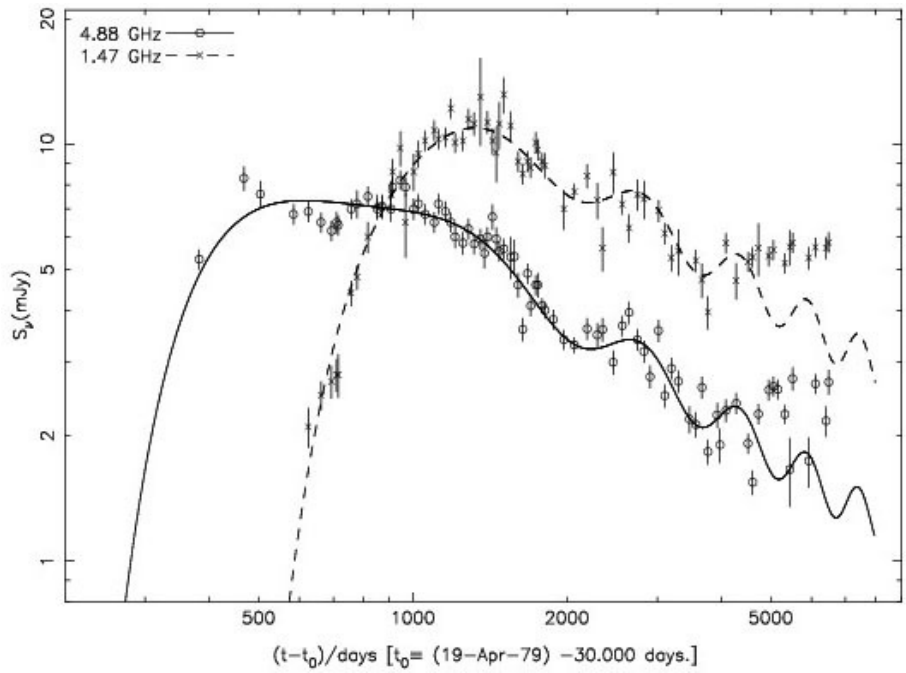


FIGURE 1.1: Supernova light curve. The height of the observations on the y-axis indicates the observed intensity at the given time position

is not always intrinsic to the source, that is, the result of some change in the source itself such as in a supernova event. Some scintillation will result from the light interacting with the interstellar medium on its journey to earth. This is called extrinsic variation and will complicate the process of analysing the time series.

1.3 The Problem

The precise problem to be explored in this research is the development of an algorithm for the classification of streaming time series data. The algorithm will receive data every 5 seconds. At each time step any new transients will be reported and along with what class they belong to with a measure of confidence. It is possible that a transient may belong to an undetermined class and this also needs to be taken into account. The detection of transients should be done as early as possible but should not have too many false positives.

1.4 Time Series Analysis

This problem is primarily one of *time series analysis*. Fortunately, in recent years this field has received a lot of attention in various domains: speech recognition (Sakoe and Chiba, 1978), handwriting analysis (Bahlmann et al., 2002), even image outlines can be represented as time series and classified (Ye and Keogh, 2009). However, the well-developed techniques from these other areas do not extend immediately to our astronomical data. Table 1.1 outlines the most serious difficulties inherent to this task.

incompleteness	Data arrives as a stream, and classification must be done without full knowledge of the curve structure. Most developed techniques assume full structure data is available.
distortions	Astronomical time series have distortions in terms of amplitude scaling, time warping, noise and missing data.
precision	Classifications must have very high precision. Too many false positives will waste astronomer time and make the system useless.
real-time	There are very large data volumes arriving in a stream - 1 GB/s, so classification must be time and space efficient and at least near-real time.
redundant	Most data is not relevant to event detection. The start and end of interesting structures must also be determined by the program.
periodic	Some data will come from periodic sources and this will confuse algorithms searching for 'one-off' events. Periodic time series may need to be identified and handled separately.

TABLE 1.1: Difficulties inherent to classifying transients from astronomical data

This set of problems is not trivial, and no individual piece of research at present addresses them all. Literature from several domains: machine learning, time series analysis (in astronomy and other fields) and statistics is reviewed and discussed in the following sections with the aim of addressing these issues.

1.5 Coping with Distortions in Astronomical Time Series

Below is a list of subproblems associated with the issue of **distortions** in our astronomical time series.

- **noisy observations**. It can be assumed that every point in a light curve has added noise. For simplicity this noise is assumed to be Gaussian distributed which is a reasonable approximation of real conditions in astronomical data.
- **amplitude scaling** - the same astronomical event will have a different intensity when observed at different distances
- **missing data** - streaming data is not continuous. This may be due to bad weather or shared telescope responsibilities.
- **time warping** - events may have different durations or unfold slightly differently, but still have very similar structures

1.6 Our contributions

This thesis contributed in several ways to solving the challenge of transient classification. I phrased the problem as a time series classification problem and gave an extensive review of existing literature from a variety of application domains. I proposed a classification framework with variety of simulated transient types and implemented software to apply the distortions present in astronomical data to them. Finally I develop a feature based supervised transient classification approach and evaluate a number of features including wavelet transforms, statistical properties of light curves and the shapelet time series feature representation. I concluded that these features hold promise for transient classification but are not yet ready for application in the VAST pipeline, and proposed data preprocessing and modifications to the feature implementations that will improve classification performance. This thesis is an important first step in developing effective transient classification for VAST.

CHAPTER 2

Literature Review

2.1 Distance Measures for Time Series

2.1.1 Overview

This literature review explores a number of approaches to transient classification. These include distance measures, Gaussian Processes, feature based classification with wavelet transforms, shapelet and motif finding, Support Vector Machines with periodic kernels, Temporal Grammars, and statistical properties of flux histograms of light curves. It concludes that feature based classification with wavelet transforms and statistical features is the most likely approach for effective classification of transients due to its ability to include many features that capture a wide variety of transient properties.

2.1.2 Introduction

The most primitive approach to analysing a time series is to overlay it onto another time series with known class and see how well it fits by calculating the Euclidean *distance* between the two curves. A low distance indicates a high similarity. By comparing the test curve to a number of training samples the most likely class is determined as that having the lowest distance (or lowest distance below some threshold). This approach, called a *Nearest Neighbour* classification, can be very effective if the time series are uniform within their class.

Unfortunately, astronomical time series do not have that property. Although they have the same form in terms of peaks, throughs, plateaus, lines, wobbles and so on, the actual magnitude and time over which astronomical transients unfold is not necessarily the same. These issues are called amplitude scaling and time warping. These two distortions, compounded with the lack of complete data make Euclidean distance useless.

Overcoming this distance measure issue may yield in itself a solution to our problem. Additionally, distance measures are likely to be integral to data preprocessing or to the application of other approaches such as temporal grammars, support vector machines and shapelets discussed later. The following section is devoted to an exploration of more flexible distance measures than the Euclidean distance.

2.1.3 Dynamic Time Warping

Dynamic Time Warping (DTW) is a technique first introduced in (Sakoe and Chiba, 1978) and was popularised in (Berndt and Clifford, 1994) with successful application to speech signal classification. The algorithm is a dynamic programming approach to that allows the matching to 'skip' parts of either time series in order to align them better. The distance of two time series under DTW is then the minimum across all possible matchings. Figure 2.1 shows dynamic time warping finding a better alignment for two sequences than Euclidean distance. Dynamic time warping deals very well with changes in the way transients unfold but is not guaranteed to find a good match in the presence of amplitude scaling. It also seems hard to extend this algorithm to matching subsequences as required by our problem.
Should dynamic time warping or some modification be found useful to us, an extension of the computation that allows on-line updating would be very attractive for dealing with the **real-time** complication. Such an extension is given in (Capitani and Ciaccia, 2007), which gives an algorithm with constant time updating of DTW distance for a time series stream. The distance measure so produced is a close approximation of the full DTW distance.

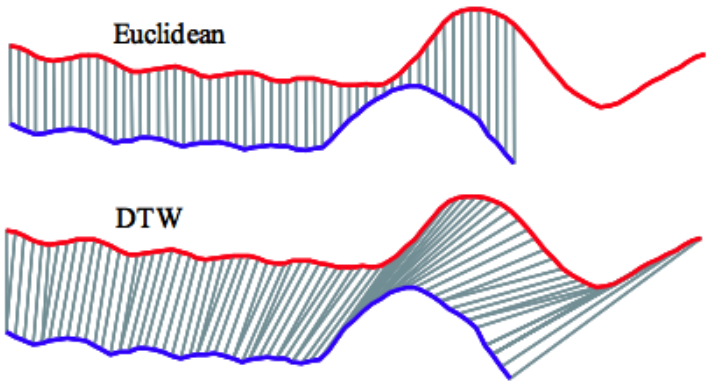


FIGURE 2.1: Dynamic time warping finding a superior alignment for two time series than Euclidean distance. Alignment is indicated where a grey line joins two points of the series.

2.1.4 Longest Common Subsequence for Time Series

A similar distance measure to DTW is the Longest Common SubSequence (LCSS). LCSS differs from DTW mainly in that all components of both series do not need to be included in the matching. The most similar components of each series are compared in the distance measure only. This approach will find be able to match subsequences to series and cope with time warping at the same time. In (Vlachos et al., 2002), an implementation of LCSS that allows translations (not scaling) in space and is fast to compute is given. The translations are incorporated into the dynamic programming algorithm as another dimension to search through. In the paper it is applied to accurately recognise human gestures presented as multivariable time series in the presence of time warping and translations in space.

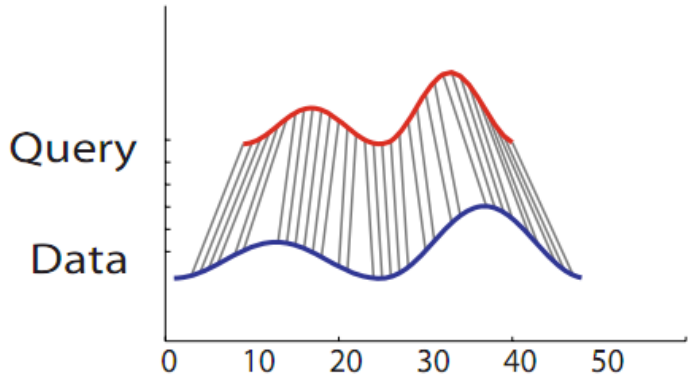


FIGURE 2.2: Longest Common Subsequence Distance match between a test (query) curve and a training sample (data)

2.1.5 Complexity distance

An interesting recent paper of note is (Batista et al., 2011). This paper attempts to produce a distance measure based on the abstract notion of complexity, the relative smoothness or bumpiness of a curve. A simple approach to this suggested in the paper is to factor into an existing distance measure (for example, euclidean distance) the relative length of the two time series. For example, if A and B are two time series, C(A) and C(B) are the lengths of A and B, and E(A, B) is their euclidean distance, then a new distance measure would be:

a

$$D(A,B) = \frac{\max(C(A), C(B))}{\min(C(A), C(B))} E(A,B)$$

The intuition here is that the length of the time series roughly corresponds to its variance over time - the closer two curves are in length the closer they are in complexity. The paper gives good results for a complex time series representing leaf outlines. A similar idea may help to improve classification accuracy for astronomical time series.

2.2 Gaussian Processes for regression and classification

2.2.1 Introduction to Gaussian Processes (GPs)

A Gaussian Process (GP) is a statistical model of data that can be used for regression, noise-filtering, classification and prediction. In this section a discussion of the regression and noise-filtering abilities will be presented in the hopes of addressing the **distortions** issue. A thorough introduction and exploration of gaussian processes can be found in (Rasmussen and Williams, 2006). A brief overview for the purposes of discussion is provided here.

A GP consists of a multivariate gaussian distribution, where each dimension of the distribution corresponds to an index (in this case, time index) of an input point, say x. Gaussian distributions are defined by a mean and a covariance matrix. GPs are more general in that the entries of the covariance matrix are determined by a covariance function k. An evaluation of GPs for regression is carried out in (Rasmussen, 1996), demonstrating that GPs are competitive with neural networks on nonlinear regression tasks, even performing slightly better when large amounts of noise are present.

Covariance functions determine the influence that the points in the distribution have on each other. They are used to control the amount of flexibility and smoothness in the function that the distribution represents. This is done both through the choice of function (popular choices are the squared exponential or the Matern functions), and *hyperparameters* to the chosen function. Some commonly used hyperparameters are lengthscale, noise variance and signal variance These correspond respectively to the expected influence of points based on their distance apart in the index, the expected fluctuation in

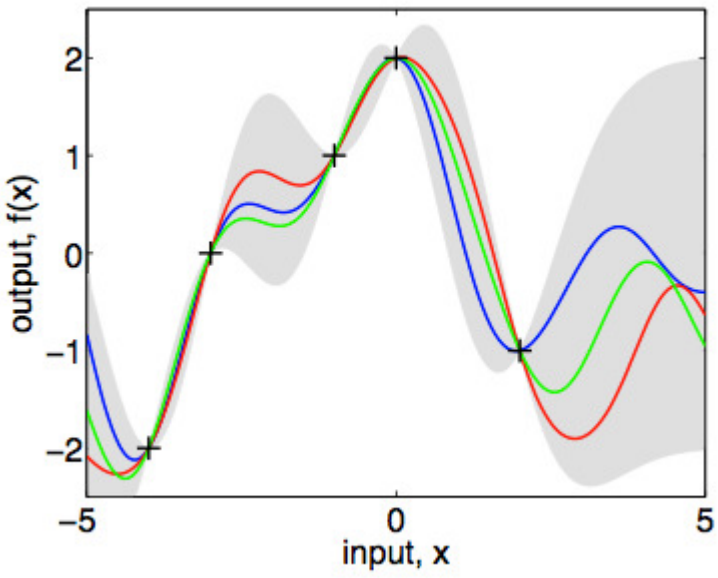


FIGURE 2.3: A Gaussian Processes doing Non-Linear Regression on a Time Series. The crosses indicate observations, while the grey bands indicate uncertainty. Some possible underlying functions drawn from processes are shown in green, blue and red. Taken from (Rasmussen and Williams, 2006)

height and expected noise in the data. Optimising the hyperparameters for the dataset is key to getting good regression, prediction and noise filtering.

2.2.2 Sparse Gaussian Processes

The time and space complexity of the fundamental GPs is prohibitive for large volumes of data. In recent years several versions of GPs with approximations to covariance functions have been developed to cope with these constraints. These improvements give a time complexity of $O(mn)$ for training time and $O(m^2)$ for prediction where m is the number of basis functions and $m << n$. The most recent versions of sparse GPs are Sparse Spectrum Gaussian Processes (SSGPs) in (Lázaro-Gredilla et al., 2010) which use periodic basis functions in the approximation. Sparse Multiscale GPs (Walder et al., 2008) and Fully Independent Training Conditional (FITC) (Snelson and Ghahramani, 2005) comprise the state of the art. Despite a sensitivity to overfitting on one highly non-linear dataset, SSGP otherwise outperforms FITC and SPGP. With the excepting of the overfitted dataset, all implementations approach the error of a full GP as the number of basis functions (m) in the approximation is sufficiently large.

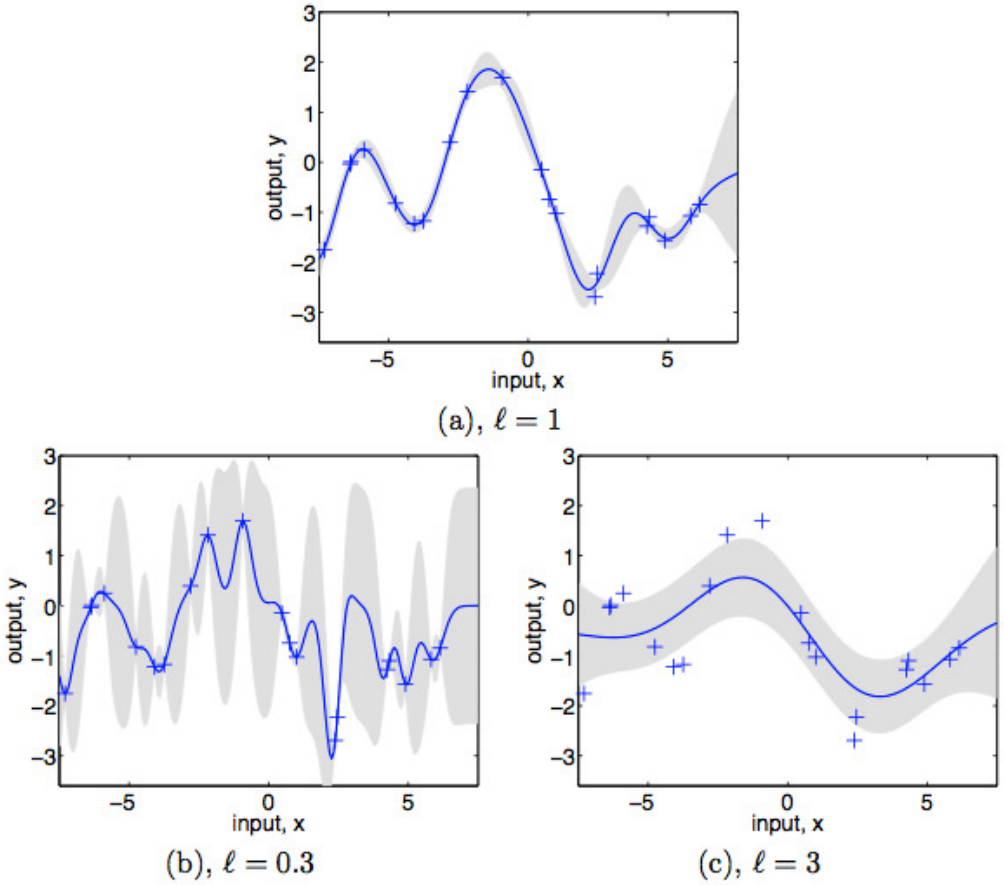


FIGURE 2.4: 3 Gaussian Process interpretations of the same data with varying length-scales (l). The variance bands of the three plots demonstrate that choosing the right hyperparameters is important for accurate regression. Taken from (Rasmussen and Williams, 2006)

2.2.3 Online Gaussian Processes

Standard GPs can be altered to allow for on-line updates of the training variables, see (Osborne and Roberts, 2007). Recently, sparse models have also been implemented that also allow for fast on-line updating. In (Ranganathan et al., 2011), a sparse GP is presented giving an $O(n)$ update time per addition of an additional point. These GPs have full predictive power and outperform state-of-the-art sparse GPs on non-linear data sets. There are limitations on this algorithm however, most importantly that optimising the hyperparameters to new data is a costly $O(n^2)$ step. This is not ideal since we do not know anything about the structure of our unfolding time series and some tuning is necessary for good results. However, if this issue can be overcome, these are an attractive option for solving the **distortions** problem.

2.2.4 Summary

Gaussian processes are a promising tool for regression and noise filtering in astronomical data. They have a high time complexity and their running time in practice will need to be tested by experiment. Gaussian processes are unsuitable for transient classification because they require a known start and end point of a transient event and any shifting of the time series will lead to poor classification. The boundaries of the transient event will not be available in the VAST classification pipeline.

2.3 Approaches to Time Series Classification

This section gives a general discussion of techniques developed for analysing and classifying time series. Some of the methods will be more useful as *features* for generic machine learners. Falling under this category are wavelets, temporal grammars and periodograms. Others are more directly useful for classification, such as shapelets, support vector machines (SVMs), and phase invariant kernels. Each subsection will aim to discuss the technique in terms of the relevant complications in our problem, namely the **real-time**, **precision** and **periodic** issues.

2.4 Frequency Domain Approaches

2.4.1 Introduction

Frequency domain analysis is the most well explored technique for studying astronomical time series. In itself it is highly effective for identification and classification of periodic stars - one category of astronomical time series our system needs to deal with. Additionally, the outputs of the various techniques can be used to extract features for time series without periodic structures. These features can then be used in generic feature based machine learning classifiers. Worth noting is that frequency metrics are less sensitive to noise and time warping than time domain analysis. A brief survey of the techniques and how they may be applied to our problem follows.

2.4.2 Discrete Fourier Transforms and the Lomb-Scargle Periodogram

A Fourier transformation is decomposition of a continuous function into sinusoids. The strength of the peak for each component of the decomposition indicates the strength of that component in the original

signal. There is a version of the Fourier transform that works for discrete data such as ours, but unfor-

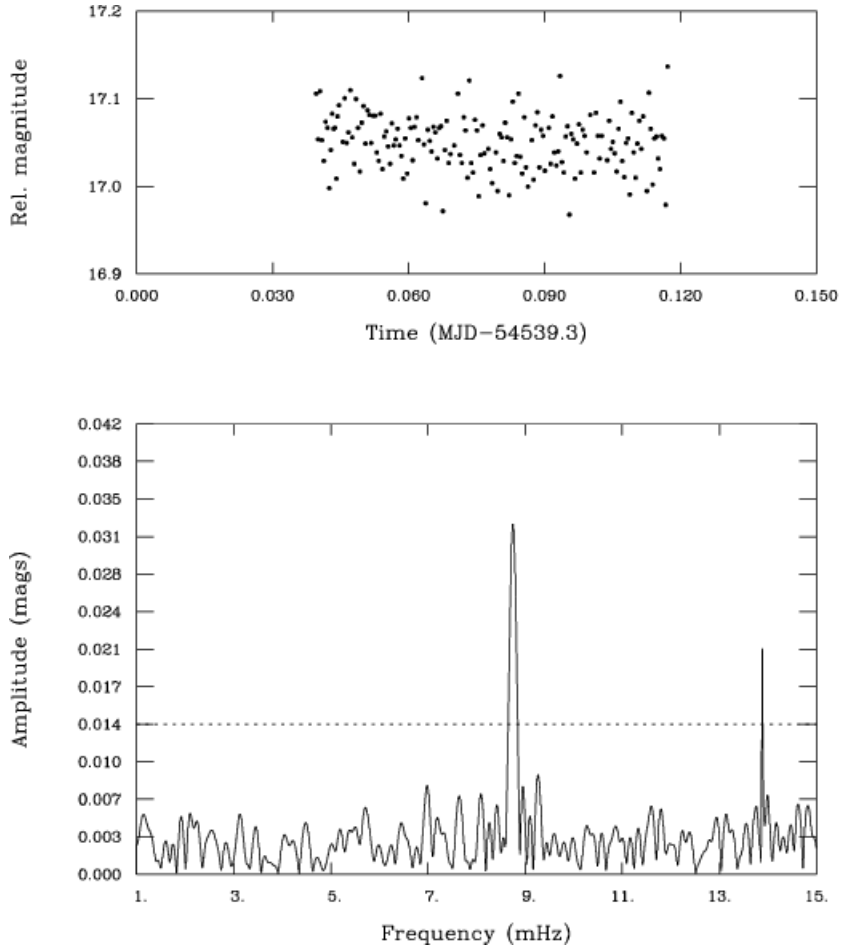


FIGURE 2.5: Fourier Transform of an astronomical time series. The peaks represent the most significant periodic components. In this case the signal has two clear periodicities.

tunately is sensitive to discontinuous regions. The Lomb Scargle Periodogram, introduced in (Lomb, 1976) and (Scargle, 1982), is a spectral decomposition that copes with this issue. The method involves fitting a number of sinusoidal basis functions onto a dataset using least squares regression. The output is a spectral decomposition of weighted sinusoids like the Fourier transform.

2.4.3 Wavelets

A wavelet decomposition for a time series is a transformation into a number of basis waveforms. The Fourier transformation presented above is one such decomposition, but many other forms exist with useful properties such as the Haar wavelet transform.

The Haar wavelet decomposition produces a set weighted component rectangular shaped waves of decreasing granularity. The output of this process on a time series is given in 2.7.

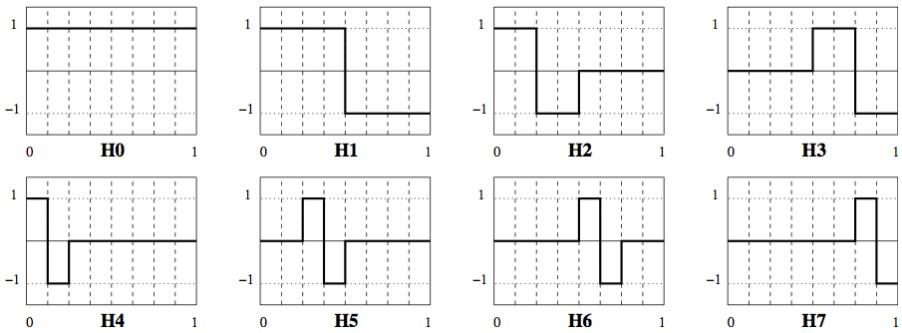


FIGURE 2.6: The first 8 Haar basis wavelets.

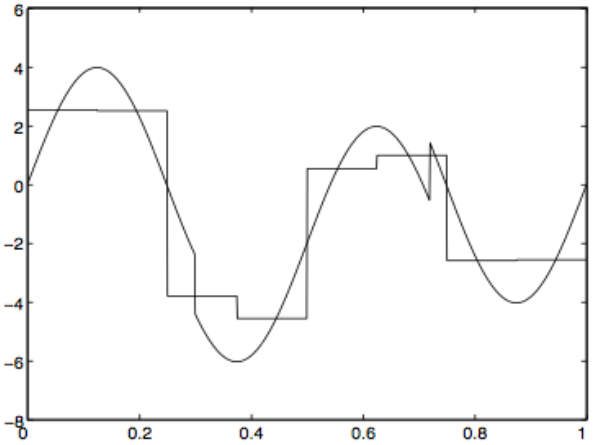


FIGURE 2.7: A time series reconstructed from its Haar wavelet decomposition.

This transformation takes linear time in the length of the time series. As explored in (Popivanov and Miller, 2002), The structure of the rectangular Haar wavelet makes similarity comparison for time series very fast, an appealing property for the large data volumes involved in this research. Unfortunately, Haar Wavelets are still inflexible under astronomical time series **distortions**, but could possibly be utilised for similarity search with a non-Euclidean distance measure.

2.4.4 Phase Invarant Kernels

Besides spectrum analysis, an approach to classifying periodic stars is to use a phase-invariant distance measure. Phase invariant means that no matter what translation the time index is under, the distance

between the two light curves is the same.

In (Wachman et al., 2009) a phase invariant Kernel is proposed for periodic astronomical time series. It is immediately suitable for the nearest neighbour algorithm. The first proposed kernel is computed for two time series x and y as:

$$K(x, y) = \sum_{i=1}^{n} e^{x \cdot y_{+i}}$$

That is, the exponential of the dot products for all possible discrete alignments of the two time series. This measure gives much higher scores to those time series for which there exists some very close alignment.

More interestingly, with some modification this kernel is also suitable for use in a support vector machine. With this approach the authors get excellent results a dataset of real light curves with accuracies of greater than 99%. The kernel score is fast to compute with a computational bound of $O(n \log n)$. The authors do not address the issue of amplitude scaling and classify with full light curves, so there are still some complications for our problem that this approach does not cover. Nevertheless, the use of handcrafted kernels for time series is an interesting one, explored further in Section 2.5.1.

2.5 Time Domain Analysis Approaches

Time series analysis which works directly with the time indexed data is called *time domain* analysis. The simplest possible classification method, nearest neighbour classification, was already mentioned in Section 2.1 when discussing distance measures for time series. Wavelets and shapelets can be used as features for generic learners such as support vector machines and neural networks. Additionally, those same machine learners can be applied directly to the data for classification.

2.5.1 Support Vector Machines

Support Vector Machines are machine learning tools that work by computing a so-called maximum margin that separates two classes in a feature space. A simple example of a support vector machine working in euclidean space is presented below. Multi class classification is possible by feeding a test

case into an ensemble of binary classifiers, facing them off against each other in a tournament style. The winner of the tournament is chosen as the classification for that test case.

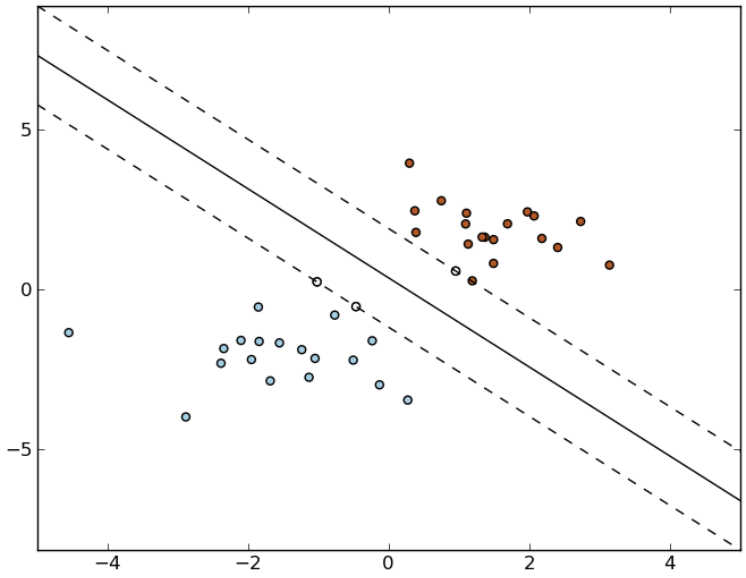


FIGURE 2.8: Separation margin for two classes (blue and red dots) produced by a support vector machine

A simple suggestion to use one of these SVM ensembles in our task would be to represent a time series with n time indices as a vector in n dimensional Euclidean space. A test class would be fed into the ensemble and whichever time series class it most closely resembles in Euclidean distance would be chosen as the label. Of course, this approach has all the problems that the Euclidean distance measure has as outlined in 2.1, But classification would be faster (and possibly more accurate) than the nearest neighbour algorithm.

A modification of the SVM algorithm that is very relevant to our task is outlined in (Shimodaira et al., 2002) and (Bahlmann et al., 2002). In this paper a kernel is developed which uses the dynamic time warping distance of two time series. Kernels can be used to transform time series into points in a new feature space. The SVM algorithm can modified to separate these points if the kernel has certain properties. The proposed kernel unfortunately is not completely suitable (it is not positive semidefinite) and cannot be expected to work properly, but it nevertheless gives comparable performance to Hidden Markov Models on speech and handwriting classification datasets in these two papers. This notion of a kernel designed explicitly to have the distance measure properties we need for our time series is worth exploring further.

2.6 Temporal Grammars

2.6.1 Introduction

Astronomical time series have forms which make them distinct from each other and from background noise. Peaks, grades of slopes, valleys, bumps and other local features characterise each class. Humans are good at discerning these forms, but to train a machine learning classifier requires some kind of language to express the substructures - a temporal grammar. This Section discusses feature extraction methods for temporal grammars that are both robust for classification and are human interpretable - they can be easily adjusted and reapplied. There are limitations to this technique for our problem in that any features extracted must be invariant to the in-class distortions of astronomical time series. This may be possible if the features are sufficiently abstract.

2.6.2 Early Temporal Grammars and Basic Approach

Early work in this area consists of approximating a pattern using simple shapes, for example, straight line segments as in as in (Keogh and Pazzani, 1998). One attempt at a generalisation of temporal grammars is found in (Olszewski, 2001), and this is a good introduction to the end-to-end approach. This paper utilises a grammar of {constant, straight, triangular, trapezoid, sinusoid, exponential} to the task of pattern representation. This approach is very similar to the shapelet approach except that these features are more abstract, allowing potentially for modifications that cope with stretching and scaling more easily than shapelets. Additionally, the simplified nature of grammar components means that comparing features amongst time series is much faster. Dynamic programming is used to decide on optimal partitions of the pattern by finding the minimal error choices of substructure pieces. These substructures are then represented as numerical feature vectors which are fed into a standard machine learning classifier. The implementation was run on complex non-linear time series datasets including ECG (Electrocardiogram) data.

The paper implements and trials a feature extractor, comparing it with feature extraction methods based on wavelets and fourier transforms. The temporal grammar approach was competitive, and better in many cases. Unfortunately, extraction of these features is a slow process and this approach will unlikely be feasible for time series data streams.

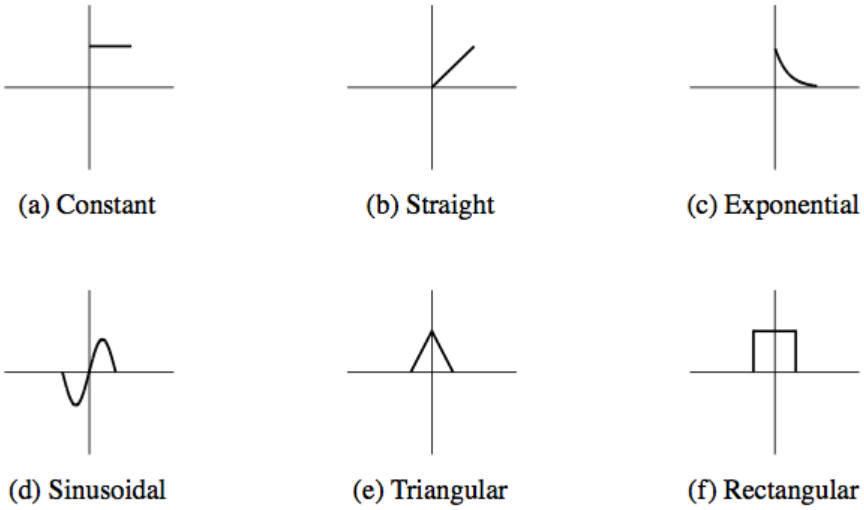


FIGURE 2.9: Components of temporal grammars used in the Olszewski paper

2.6.3 Recent Improvements and Distortion-Invariant Forms

In (Kadous and Sammut, 2005) a more abstract temporal grammar is proposed. In this case, not based on parameterised curve fitting but on more general pattern substructures including plateaus, increasing and decreasing sections and local maxima and minima. The grammar can be extended but it already quite powerful with those features alone. A classifier is built by constructing a decision tree from features extracted from training samples.

The Kadous temporal grammar is applied to similar datasets as in Olszewski: time series and a temporal representation of sign language expressions. Both datasets are highly nonlinear. Accuracy around the average for professional cardiologists is achieved, notably without any expert background knowledge introduced into the model. The model also outperforms a Hidden Markov Model implementation slightly, with the added bonus that the rules produced are human interpretable (HMM training states are not).

Since the features are more abstract than the parameterised curves of the previous section, proposals for dealing with amplitude scaling and warping seem feasible. For an example, one could look at the distributions of local maxima and minima to find likely matches, then search for constant factor differences in their amplitude for confirmation, similarly for plateau length and height. If an algorithm for

rapid extraction of features from time series streams were developed, a similar approach could be used in solving this problem.

2.7 Motifs and Shapelets

2.7.1 Introduction

Motifs are defined across the domain of pattern matching as subsequences that occur frequently within either a singular sequence, or a collection of sequence objects. Shapelets (Ye and Keogh, 2009) are a related concept to motifs, and are defined as subsequences found in a collection of sequence objects that discriminate best between the classes in that collection.

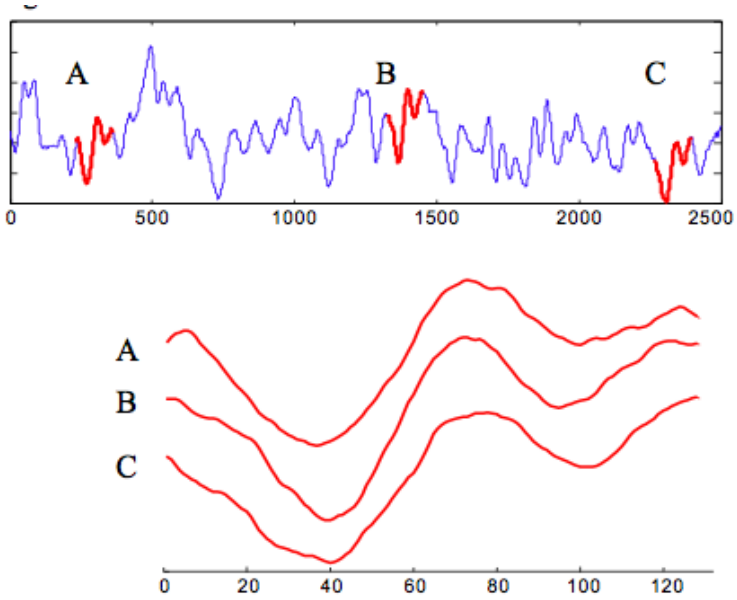


FIGURE 2.10: Figure of a motif in a time series taken from Lin et al. (2002)

Motifs and shapelets are of interest to addressing the problem posed in this thesis because they contain information about an astronomical transient without any reference to a start or end point for the event, a complication encapsulated in the **subsequence** distortion. Attempting to match a full time series from our training data to a fraction of a complete event light curve suffers from a variety of problems. Distance measures such as the Euclidean and Dynamic Time Warping distance (see 2.1 suffer greatly in the presence of noise and missing data. They are also not well defined when the test case is an incomplete version of the training curve. Classification methods such as Gaussian processes, support

vector machines as well as many potentially useful features such as the Haar wavelet transform require advanced knowledge of the start and end points of an event.

Additionally, because subsequences which are best at discriminating a particular class from others are also those which are less likely to be totally corrupted by added noise, we can expect an improvement in accuracy over standard distance measures when noise distortion is present.

Finally, although not a distortion explicitly examined within our experimental framework, in a real-world context both motifs and shapelets will be helpful in coping with local variations in training data. Finding regions of light curves that are consistent in structure within a class as opposed to regions of high variability decreases inter-class confusion in the classifier. Local variations are exhibited by some of the light curves in our experimental dataset, in XRBs and both kinds of flare stars. Strict definitions of motifs in the context of time series lead to algorithms for their extraction and use in classification.

2.7.2 Motifs

Algorithms for motif extraction are well developed in the field of Bioinformatics, where motifs are defined as frequently repeated strings in long sequences of the discrete symbols ‘G’, ‘A’, ‘T’ and ‘C’ representing DNA. The maturity of these algorithms influences the approach in Lin et al. (2002) which utilises the Piecewise Aggregate Approximation to transform a time series to a symbolic representation. Once a sequence of discrete symbols the aforementioned algorithms are made applicable for motif extraction. There are complications to this approach however, in that the PAA transformation is not always a good representation of the original time series. This is especially true in the presence of noise. An additional issue is that no consideration is given to the meaning of adjacency between symbols. Adjacency is meaningless for DNA, but very important for time series data.

Motifs are useful in a context where large amounts of a sequence are highly variable within or independent of the class of event producing the sequence (as in DNA). In real-world astronomical data this may well be the case, but in the context of our simulated data the light curves are very generic within a given class. Under these definitions of similarity and frequently occurring subsequences then, it is very likely a motif finding algorithm would declare the entire sequence as a motif for a class! What is needed is a slightly different idea: a subsequence which not only occurs frequently within a particular class, but also one which does not appear in any *other* classes within a multi-class classification context. The concept of a shapelet (see below) meets our requirements.

2.7.3 Shapelets

The idea of shapelets, first presented in Ye and Keogh (2009), is to find subsequences in time series that are maximally discriminative amongst the classes of the dataset. The algorithm proposed in this paper gives a brute force approach to extracting shapelets from a dataset, using entropy as a natural choice of measuring inter-class discriminative power. Once extracted, shapelets can be incorporated into classification in a variety of ways. The Ye paper proposes a decision tree approach where logical rules using the (Euclidean) distances of each of the shapelets extracted from the dataset to a test case are used to determine its class label. Another obvious approach is to use the nearest neighbuor algorithm combined with any sensible distance measure such as subsequence Euclidean distance or constrained Dynamic Time Warping (see Section 2.1).

FIGURE 2.11: Figure of time series shapelets extracted from time series representing arrowheads, taken from Ye and Keogh (2009)

Shapelets are extracted as follows: For every single subsequence of any length in the dataset, the subsequence distance (see section 2.1) is computed to every element of the dataset. This collection of distances is split into two subsets D_{left} and D_{right} by user specified threshold τ. The τ parameter determines the variance allowed for a particular shapelet being extracted from the dataset. The expressiveness of the shapelet within a class is then measured by the information gain:

$$I(s,\tau) = E(D) - \frac{N_1}{N}E(D_{left}) - \frac{N_2}{N}E(D_{right}) \tag{2.1}$$

where $N_1 = |D_{left}|$, $N_2 = |D_{right}|$, and $E(D)$, the entropy of a dataset, is defined as:

$$E(D) = -\sum_{i=1}^{C} \frac{n_i}{N} log(\frac{n_i}{N}) \tag{2.2}$$

where n_i is the number of time series labeled with class i and is C the number of class labels in the dataset.

Choosing good shapelets for the purposes of classification simply involves choosing those which have the highest information gain. Ties (or close ties) can be resolved by looking at the actual distances across the split of D_{left} and D_{right}. The Ye paper proposes a suitable method for doing this. The threshold τ is context-specific. A too tight or too loose choice of this parameter will give no sensible outputs. For our purposes visual inspection of the output will serve fine, but if necessary good thresholds could be determined automatically by searching for values that give shapelets meeting a predefined minimum information gain.

Unfortunately since our dataset has very high intra-class uniformity, this approach will, like motif finding algorithms, most likely return an entire training sequence as the most discriminative subsequence. Short, expressive subsequences are more likely to be robust to noise and are important for on-line and subsequence classification. Extending the Ye shapelet extraction algorithm to have a range of lengths is trivial, but what lengths are useful for the purposes of classification will need to be found by experiment.

The work in Ye and Keogh (2009) is extended in Mueen et al. (2011), giving a faster algorithm for finding shapelets. The improvement comes both from the caching of distance computations and early abandonment of shapelet evaluation using a theoretical limit on information gain difference for adjacent subsequences (subsequences that are mostly overlapping). Taking advantage of this improved performance, the authors also propose an extension of the classification component of the Ye paper. This extension involves using combinations of shapelets linked by the *and* and *or* logical operators, meaning respectively both or either shapelet provides good discrimination to a class (see figure 2.12). Finding these logical combinations fits naturally into the shapelet extraction algorithm by evaluating the *and* and *or* respectively as the maximum and minimum of the information gain of the individual components in the expression. Whether or not this modified approach will improve classification on our dataset depends on how distinct each shapelet is to each class, and will need to be tested by experiment.

2.7.4 Shapelets and streams

Although this thesis involves classification of static data objects mimicking a sliding window approach, the ASKAP telescope array in will involve a data stream. It is worth including into this literature review then the work in Xing et al. (2011). This paper takes the definition of a shapelet outlined in Ye and Keogh (2009) and gives a measure of quality in *earliness of classification*.

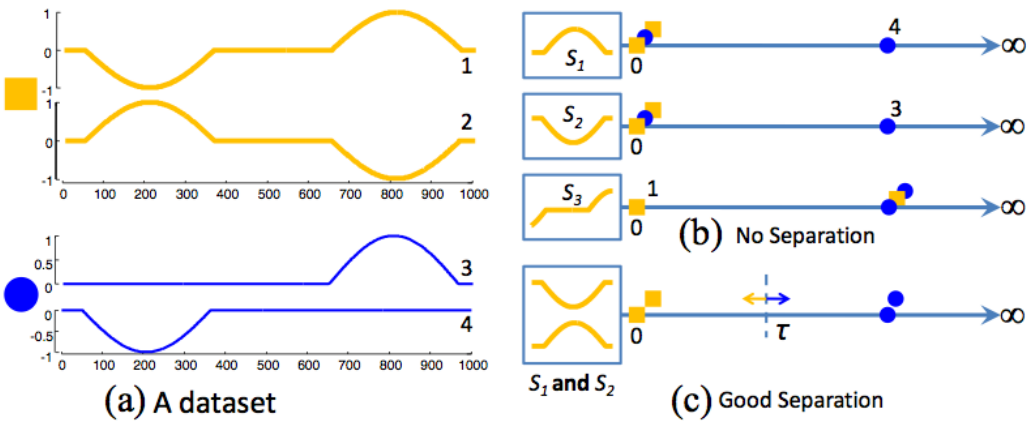


FIGURE 2.12: Figure of logical shapelets providing a better discrimination amongst a dataset with similar subsequences, taken from Mueen et al. (2011)

2.7.5 Summary

In summary, shapelets are clearly preferable to motifs both in the context of this thesis and in real-world astronomical data. In Ye and Keogh (2009) and Mueen et al. (2011) both clear definitions of shapelets and fast algorithms for their extraction are laid out. Classification is proposed using decision trees, but nearest neighbour with a suitably defined distance measure such as subsequence distance (see **??**) would work as well. They are potentially useful both as a stand-alone classification method, and for incorporating as features into a complex feature-extraction based classifier.

2.8 Astronomical Time Series Classification

Besides many generic approaches to classification presented so far, there are approaches that are astronomy specific. The work presented in (Richards et al., 2011), demonstrates a simple set of features that may be useful in providing additional discrimination amongst astronomical time series. The features include the standard deviation of the measurements, the distribution of flux amongst linearly spaced buckets across the intensity of the measurements, the maximum and minimum changes in the light curve, and many other simple and fast to compute properties of the data. These non-periodic features provide a 4% improvement in the error rate for that task, the classification of variable stars. When much of the light curve data is shape based (rather than spectral), these simple features should play a much larger role in classification.

2.9 Summary and Possible Research Approaches

Literature from many domains involving time series analysis was reviewed, all giving partial solutions to the astronomical time series classification problem. There are a few promising approaches that emerge from this review. The first is to develop an effective distance measure or a Kernel that copes with distortions effectively, and then to apply the Nearest Neighbour algorithm or a feature based classifier such as a Support Vector Machine. No perfect distance measure exists in literature surveyed so far, but several come close, such as Longest Common Subsequence. A simple modification may sufficient to get a practical solution to the problem. Gaussian processes are appealing because they handle missing data and noise naturally but have a high time complexity and are will not be able to classify transients that do not have defined start and end points. Feature based classification using any supervised classifier and features taken from the wavelet transforms and statistical features from (Richards et al., 2011) is the most promising classification approach as it is extensible and can incorporate features that capture the diversity of structural properties that appear in astronomical transients.

CHAPTER 3

Experimental set-up and data simulation

3.1 Introduction

The question to be addressed by this research is how effective time series classification algorithms will be on the data collected by the ASKAP telescope array. ASKAP will begin operations in 2012 and in existing astronomical datasets, the systematic properties of the survey dominate over transient behaviour. The sampling methods are inconsistent, often sparse, and the data quality typically far worse than that which ASKAP will produce. Since there is no suitable existing dataset representing the range of phenomena that we expect to see with ASKAP, a set of simulated time series based on models of transient events has been created. Distortions will be applied to these simulated light curves to simulate the ASKAP data conditions and to assess potential classification approaches. These models do not perfectly represent real world transients, but are sufficiently similar so that they challenge the classification algorithm in the same way. By modifying the type and severity of the distortions applied the simulated data, knowledge about the practical classification of survey classification can be gained. Any classification algorithm for transient events can be inserted into this framework and evaluated. This understanding of the complications of on-line classification will inform algorithm choices in the VAST pipeline (Figure B.1).

3.2 Transient types

The Universe contains a vast array of transient phenomena, some of which are well understood such as Supernova events, and others that are yet to be explained. For this project I have included seven simulated transient types that are representative of the kind of phenomena expected to be encountered by the VAST pipeline. The models for these simulations were provided by Kitty Lo (VAST memo in prep).

- Extreme scattering events (ESE)

- FSdMe flare stars (FSdMe)
- FSSCVn flare stars (FSRSCVn)
- Supernovae (SNe)
- Novae
- X-Ray binaries (XRB)
- Intra-Day Variables (IDV)

These 7 simulated classes are combined with a non-transient class (BG) representing a constant source with Gaussian noise added. Figure 3.1 shows samples of the undistorted simulated transients in the dataset.

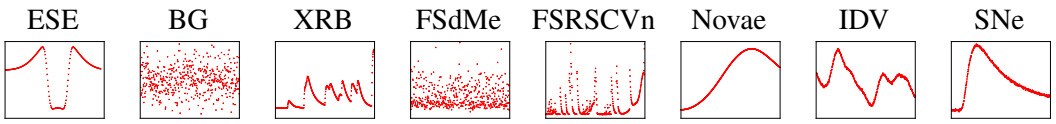


FIGURE 3.1: Samples of undistorted light curves taken from the simulated transients in the dataset

3.2.1 Description of the transients

Extreme scattering events (ESEs) are a lensing effect on the light from stellar objects produced by the passage of compact objects between the source and earth in the interstellar medium. They are characterised by a wobbling of the light curve as the object passes over the path. An example of an ESE event collected in the real world is shown in 3.2.

3.2.2 Extreme Scattering Events

Intraday variables (IDVs) are continuously and slowly varying sources that have light curves similar to those shown in 3.3. Flare stars like the FSdMe and FSRSCVn classes are constant sources with periodic or unexpected jumps in intensity called flares. Supernovae are enormous explosions produced by the collapse of large stars at the end of their lifecycle and are characterised by a sudden exponential increase in flux followed by a slow decay, usually emerging out of background noise. Novae are similar in structure but represent less dramatic stellar events. They have a much more gradual rise and decay. X-ray binaries are binary star systems where one partner is a very dense star or a black hole. They emit large amounts of energy as X-rays, but also in other spectra.

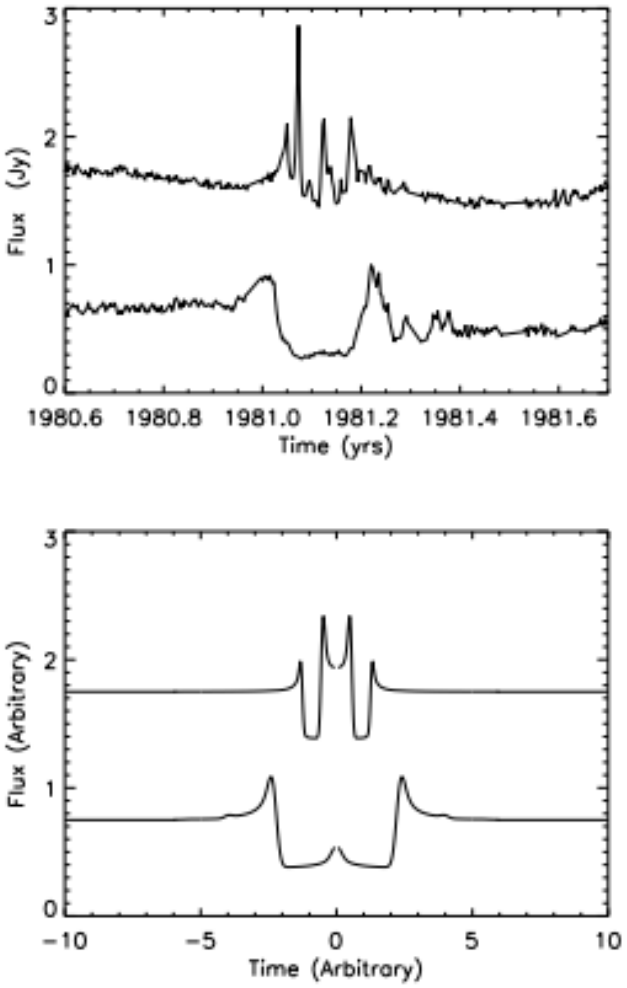


FIGURE 3.2: Figure of real world ESE data taken from Walker and Wardle (1998)

These classes do not represent all the potential transients that could arise in reality, nor are they completely accurate representations. Their structures however are sufficiently similar to the real thing to afford a preliminary investigation into the difficulties of machine-learned classification.

3.3 Data quality variables

In order to examine the performance of classifiers in terms of the different data quality issues present in the real world, classification was with a variety of distortions applied, both one at a time and simultaneously. Classification performance will be assessed while applying each of these distortions individually

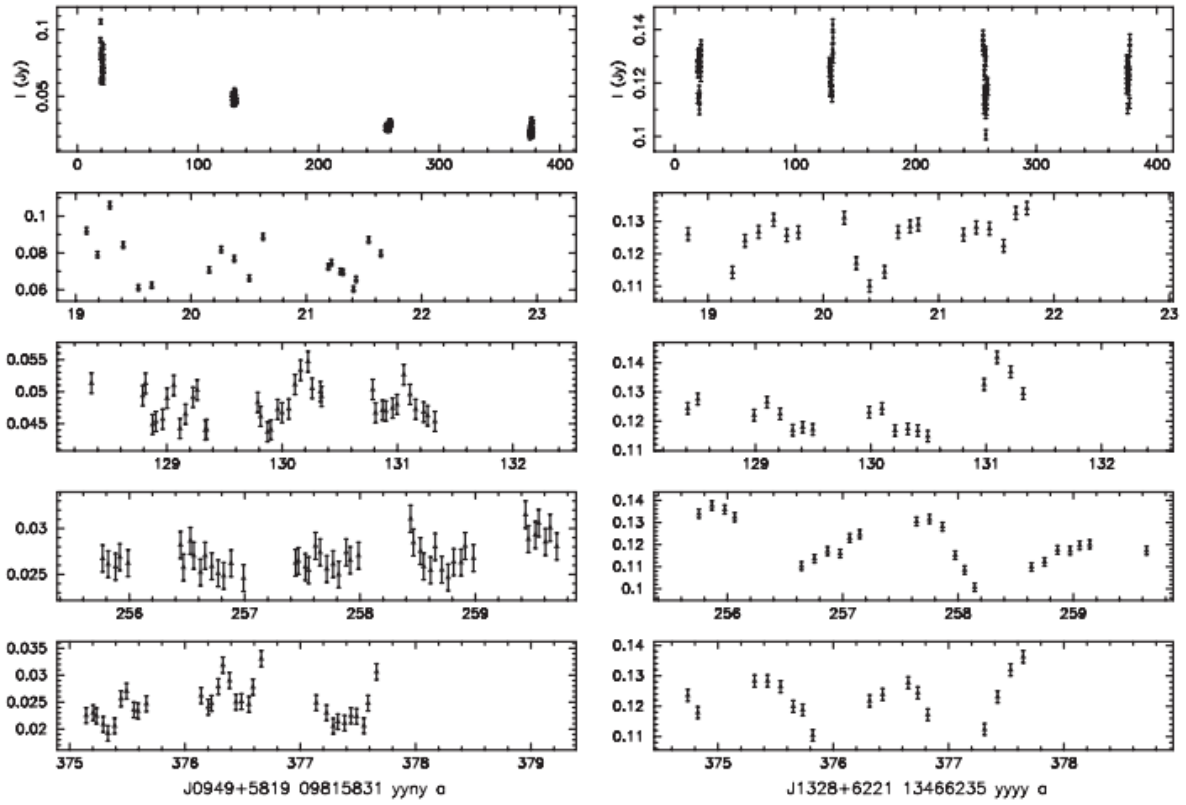


FIGURE 3.3: Figure of real world IDV data taken from Lovell et al. (2008)

and altogether. It is likely that the combination of multiple distortions will compound the individual losses in classification performance.

3.4 Implementating lightcurve distortions

The order in which multiple distortions are applied is important, and the exact way distortions are applied within each class must be carefully done to ensure to ensure there is no unusual influence on the results. Distortions are applied in the order and with the options outlined in the table below. The output of step n is fed into step $n + 1$.

3.4.1 Simulating a power law distribution

Centered is performed by subtracting the mean from the light curve and dividing by its standard deviation. The original signal should have no outlier points or noise so this should remove amplitude scaling

Incompleteness Data arrives as a stream and early classification is important for deploying telescopes around the world for more detailed study of an interesting event. Varying the percentage of the total light curve that is available is important for assessing the viability of on-line classification.

Noise Noise is a consistent factor in telescope observation. Signal noise results from atmospheric distortions, intrinsic equipment inaccuracy, and objects in the interstellar medium that interrupt light from distant objects.

Missing data Due to poor weather conditions or competing demands on telescope time, some data will simply not be available. Missing data has been modelled by removing small chunks (5-10%) of the full dataset, randomly distributed.

Amplitude scaling Due to the distribution of stellar events, as well as intrinsic differences in the brightness of these events, the actual intensity of points in the signal is not meaningful, *only the intensity of a point relevant to the other points in the signal*. The average intensity of observed signals corresponds roughly to a -2.3 power law distribution. This will be compared to light curves which are centered (mean subtracted and divided by their standard deviation) to examine the effect that the power law distribution has on the classification effectiveness.

Signal variation All the light curves used in classification will have variation in the way in that the light curves unfold. For light curves that are better defined by their shapes such as ESEs, differences in the strengths of slopes, time between maxima and minima will change. Periodic signals such as IDVs will have differences (but also similarities) in their characteristic frequency spectra. It is not possible to easily change or quantify the amount of variability within the dataset, but both differences in underlying frequency and structure will be present.

TABLE 3.1: Data quality conditions

Step	Distortion type	Amounts used
0	Raw signal from model	*none*
1	Distribution type	Either *centered* or *-2.3 power law*
2	Noise	0, 0.5, 1, 1.5 or 3 times the signal standard deviation as gaussian distributed noise
3	Signal available	10, 25, 50, 75, 90% of light curve data
4	Gapify signal	10, 25, 50, 75, 90% of signal randomly as 1, 2, 5% chunks
5	Stratification and classification	*none*

TABLE 3.2: Details of data distortion introduction

as a factor in classification.

The power law is implemented by drawing random numbers uniformly between $a_l^{-2.3}$ and $a_h^{-2.3}$, where

these values indicate the lower and upper bounds of the amplitudes desired in the power law distribution. The random value so drawn taken to the power -2.3 giving an amplitude in the desired range with values probability distribution corresponding to the power law.

3.4.2 Changing signal to noise ratio

Noise will be introduced into the signal by computing the signal variance and adding gaussian noise to 0.5, 1, 1.5 or 3 times that amount on top of the signal.

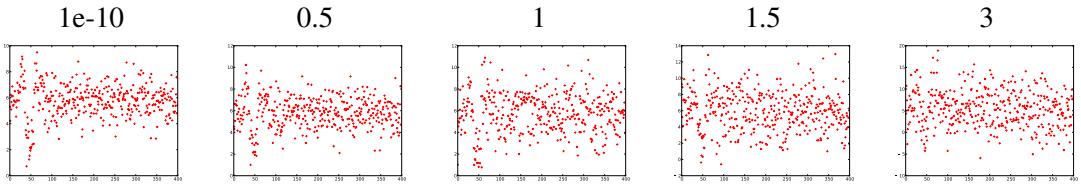


FIGURE 3.4: ESE light curves under the *noise* distortion

3.4.3 Removing part of the signal

This step is very straightforward. The latter $k\%$ of the signal is discarded and only the first part kept.

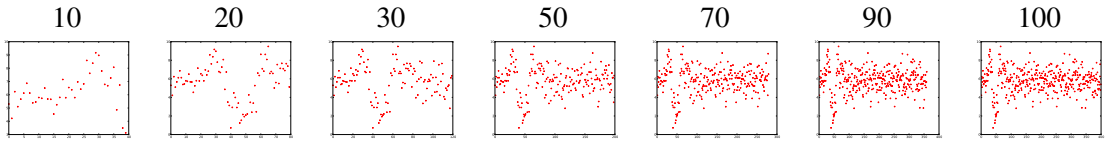


FIGURE 3.5: ESE light curves under the *available* distortion

3.4.4 Introducing gaps into the signal

This part must be done after a contiguous chunk of the signal is removed as in the previous step. This involves discarding randomly sized chunks at 1, 2, or 5% of total signal length (before step 3) at random locations, until the desired amount of data has been taken out. The procedure does not guarantee that the chunks so removed will not overlap (so larger contiguous sections may be removed than the individual chunks.

For more examples of the light curves with distortions applied refer to Appendix A

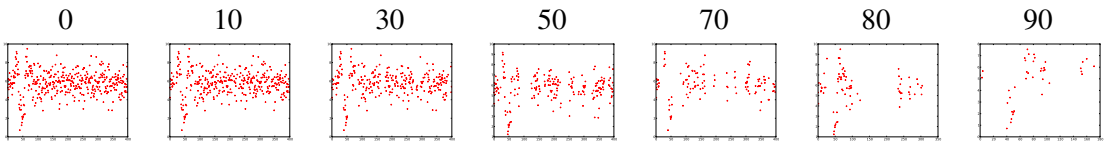


FIGURE 3.6: ESE light curves under the *missing* distortion

3.5 The dataset and classification

A dataset of 200 of each of the transient classes for a total of 1600 light curves is used to evaluate a classifier with 10-fold cross validation. Each light curve has 500 data points with the transient event possibly being shorter but situated at the start of the time series. This classification approach assumes that an earlier step in the VAST pipeline has detected a transient and an equal sized sliding window of data is provided for classification. Classification was performed on each of the single distortion steps 2, 3 and 4 outlined in 3.2 as well as a combined distortion set of 50% missing data, 0.75 noise and a power law distribution. The results of classification were returned as F-score and standard deviation of F-Score for each of the 10 cross folds as well as confusion matrices.

3.6 Summary

In this section I outlined a classification framework I implemented for evaluating the impact of both singular and combined distortions present in astronomical data on the performance of a classifier. I used simulated light curves provided by Lo (VAST memo in prep.) and implemented software to apply the distortions. All the experiments in this thesis used this framework.

CHAPTER 4

Supervised classification of astronomical Transients

4.1 Overview

Supervised classifiers are an extensible way to incorporate a variety of numerical features of a data object into a single classification scheme. In the context of classifying the transients in our dataset a number of features might be useful, such as properties of their flux distributions or their frequency domain representations. The first question addressed by this chapter is how well wavelet and statistical features can classify the dataset. Secondly I investigated how much classification performance varies under the introduction of distortions. The experiments also assessed the extent to which misclassification is caused by information loss in the light curve versus the inadequacy of our features for coping with distortions. Finally the experiments explored how the shift in feature values when using undistorted training and distorted test sets affects performance. These questions are critical to evaluate the usefulness of both the supervised classification approach and the features used for classifying transients in the VAST pipeline. The Random Forest supervised classifier is used because because it demonstrated a superior classification performance in preliminary experiments. Classification performance is measured by the F-Score, microaveraged across each class, and by confusion matrices for selected feature sets and distortion amounts.

4.2 Method

This experiment follows the experimental framework outlined in Chapter 3. Each experiment consists of 10-fold cross validation with the test set distorted according to some parameter and with the training set both with equal amounts of distortions applied, and undistorted. The exception to this is in the experiment exploring the *available* distortion, where both the training and test data have their signals cropped to the same extent. This is justified because when using a sliding window approach to classification the

length of the transient will be known. For other distortions this is not possible since the amounts of noise relative to the signal strength varies for each transient type. Additionally, the amount of missing data we in a test case will vary constantly. For details on how the distortions are applied, refer to Chapter 3.

The supervised classification method used is the `Weka` (Hall et al., 2009) implementation of the Random Forest (Breiman, 2001) classifier. The Random Forest is a variant of the decision tree classifier. Decision trees produce a class output by applying a set of logical rules on the values of the features. In the Random Forest, a collection of decision trees (called a *Forest*) working on a few randomly chosen features each provides a vote for the class of a test case. The most popular voted class is chosen as the label for the test case. Voting improves classification when some of the features disagree on the correct class. One concern with the use of Random Forests is that they are known to have a tendency to overfit training data. Given that a likely cause of misclassification is the shift of feature values under distortions, I ran a preliminary experiment to compare classification performance under the *missing* distortion of the Random Forest with some other classifiers that are known to be more robust to overfitting. The classifiers compared were the Support Vector Machine (SVM) classifier with both the RBF and linear kernel. The Random Forest outperformed the other classifiers for even large amounts of missing data.

As per the evaluation section in Chapter 3, experimental results are presented as confusion matrices and plots of F-score versus the level of the distortions applied within each experiment. Each plot of F-Score includes a subtractive analysis of the features by the removal of certain logical subsets. These results allowed me to determine which features were most important for classification and by how much; the relationship between the features and the various transient classes; and finally how effective the classification approach is overall.

Determining which factors contributed the most to misclassification is not always possible. It is hard to assess for example if the introduction of noise into a light curve leads to misclassification simply because the original signal is completely lost, or because the features are incapable of identifying the underlying structures hidden by noise. The analysis in some places makes subjective judgements about how well a human expert could classify a distorted light curve with reference to the figures in Appendix A.

4.3 Features

The features used in the experiment are outlined in Table 4.1 and are organised into logical subsets. The features included statistical properties of histograms of the light curve flux; the same statistical properties applied to a gradient histogram produced from a linear segmentation; the coefficients of a Haar wavelet transform; and the frequencies corresponding to the strongest peaks of a Lomb-Scargle periodogram.
An important component of feature extraction for the transient classification problem is that all light curves were z-normalised before feature extraction took place. Z-normalisation consists of setting the mean of the light curve to zero and its standard deviation to one:

$$f_i = \frac{f_i - \text{mean}(\text{L})}{\text{std}(\text{L})} \quad i = 1 \ldots N \tag{4.1}$$

where f_i is the ith datapoint out of N in the light curve L. This was done in the hopes of eliminating amplitude scaling and restoring the flux values of the original light curve. Z-normalisation has been applied in other work such as Loh et al. (2010) as a way of giving amplitude invariance to a time series classification approach. When data is missing or noise has been applied then the scaling will only give an approximation of the original flux values. When the distortions are severe this approximation will become worse.

4.3.1 Flux statistical features

The simplest features that could be useful in discriminating between astronomical light curves are statistical properties of the distribution of the magnitude or flux of its elements. Statistical features were used effectively to classify periodic astronomical light curve data in Richards et al. (2011). Information about the variability and dynamics of a transient is encoded in the shape of a histogram produced from the flux values. As with all feature extraction approaches in this thesis the light curves were z-normalised beforehand to eliminate the effects of amplitude scaling in the signal. The statistical properties computed are the *median*, *kurtosis*, *skew*, *minimum* and *maximum*, and the fractions of positive and negative elements of the flux set falling within predefined fractions of the standard deviation. The shape of the histogram should be affected less under a noise distortion than features working on a point-by-point basis, such as distance measures.

Skew and *kurtosis* are two statistical measures of the shape of a data distribution. Skew measures the degree of evenness in the distribution either side of the mean and for a set of flux values f with mean $\bar{f}$

Feature set	Feature	Description
statistical	Flux median	Median deviation of flux distribution
	Flux skew	Skew of flux distribution
	Flux kurtosis	Kurtosis of flux distribution
	Flux minimum	Minimum of flux distrubtion
	Flux maximum	Maximum of flux distribution
	Positive flux percentiles	Fraction of *positive* flux values within inc of the mean, for inc 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.75, 1, 1.5, 2, 3
	Negative flux percentiles	Fraction of *negative* flux values within inc of the mean, for inc 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.75, 1, 1.5, 2, 3
segmentation	Gradient median	Median of gradient distribution
	Gradient skew	Skew of gradient distribution
	Gradient kurtosis	Kurtosis of flux distribution
	Gradient minimum	Minimum of gradient distribution
	Gradient maximum	Maximum of gradient distribution
	Gradient positive percentiles	Fraction of *positive* gradient values within inc of the mean, for for inc in 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.75, 1, 1.5, 2, 3
	Gradient negative percentiles	Fraction of *negative* gradient values within inc of the mean, for inc in 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.75, 1, 1.5, 2, 3
haar	Coefficients of Haar wavelet transform	First 16 Haar wavelet transform coefficients
spectral	Lomb-Scargle periodogram frequencies	Frequencies corresponding to strongest 5 Lomb-Scargle periodogram peaks

TABLE 4.1: Core feature set and subsets used in the classificaton experiments in this chapter

the skew s is computed as:

$$s(f) = \frac{\frac{1}{n}\sum_{i=1}^{n}(f_i - \bar{f})^3}{(\frac{1}{n}\sum_{i=1}^{n}(f_i - \bar{f})^2)^{\frac{3}{2}}} \tag{4.2}$$

Kurtosis measures the 'peakiness' of the flux distribution, or, how strongly flux points are centered around the mean. For a flux sample f with mean $\bar{f}$ the kurtosis k is computed as:

$$k(f) = \frac{\frac{1}{n}\sum_{i=1}^{n}(f_i - \bar{f})^4}{(\frac{1}{n}\sum_{i=1}^{n}(f_i - \bar{f})^2)^2} \tag{4.3}$$

Figure 4.1 shows two sample light curves, their flux histograms and the corresponding skew and kurtosis. It illustrates the discriminatory power of these simple features.

Skew is highest (in magnitude) for sources with sudden peaks and troughs emerging from background

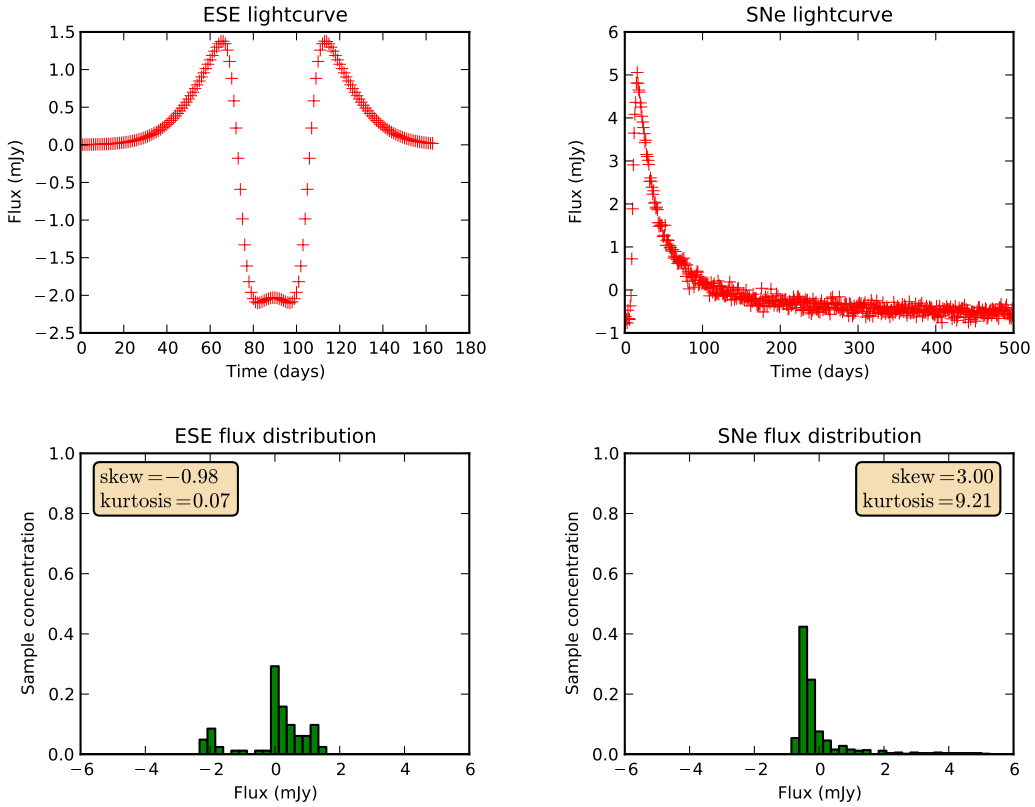


FIGURE 4.1: Figure showing flux histograms the ESE and SNe transient classes with the skew and kurtosis shown. There is a substantial difference in the skew and kurtosis betwen the classes demonstrating how these features capture the variability of the source.

noise in one direction. As a result, the Supernovae light curve has a very positive skew. The ESE has a slightly negative skew because of the relative intensity of the central dip to the peaks either side. For sources that are slowly varying or consistent, the skew will be closest to zero.

Kurtosis is highest for signals that are extremely consistent or have very sudden variations. Signals that oscillate a great deal with have much lower kurtosis. Again, the supernovae, characterised by sudden sharp increases in flux, has a very high kurtosis. The ESE also has sudden variations, but not as

strong as the Supernova. For most light curves the flux distribution will not be very similar to the normal distribution. Skew and kurtosis not always be inadequate to accurately describe the shape of the histogram and the way flux is spread across it.

A feature that does encode the entirely of the histogram of a light curve are the fractions of both the positive and negative components of the (z-normalised) flux lying within predefined increments of the standard deviation from the mean. This feature is very similar to the histogram based classification of time series proposed in Chen and Özsu (2005), as well as a similar approach involving bands around the median of the flux in Richards et al. (2011). Each increment's flux percentile forms a single feature and the group of increments form a full feature set for the positive and negative components of the histogram respectively. The feature is computed as the fraction of the flux lying within both positive and negative 0.05, 0.10, 0.20, 0.30, 0.40, 0.50, 0.60, 0.75, 1, 2, and 3 times the standard deviation from the mean.

4.3.2 Linear segmentation features

In noisy, gappy data, extracting information about the time-domain behaviour of an event is difficult. A potential method for refining poor quality data into a useful set of simple shapes outlining its behaviour is the piecewise linear segmentation proposed in Keogh et al. (2001). A bottom up segmentation of a time series into linear functions is produced, starting from every pair of adjacent points and progressively merging those points which give the lowest error in a line of best fit until the desired number of segments is reached. A visualisation of the algorithm on a light curve from our dataset is presented in Figure 4.2. A histogram of gradients is built from the linear segmentation. Each segment of a particular gradient contributes a unit of that gradient value to the histogram for each time index falling under it. Note that this means longer events with the same structure should have the same histogram, making this feature time-scale invariant provided the start and end points of the event are known. All of the features extracted for the feature histogram were also extracted for this gradient histogram, as outlined in Table 4.1.

4.3.3 Haar coefficients

The Haar wavelet transform produces a representation of the variance in a signal that is robust to noise. Square shaped Haar wavelets of decreasing granularity are fitted to data, the coefficients of each wavelet providing a means to reconstruct an approximation of the original signal. Provided that the width of the signal being transformed is the same, coefficients can be compared to evaluate the similarity of a signal

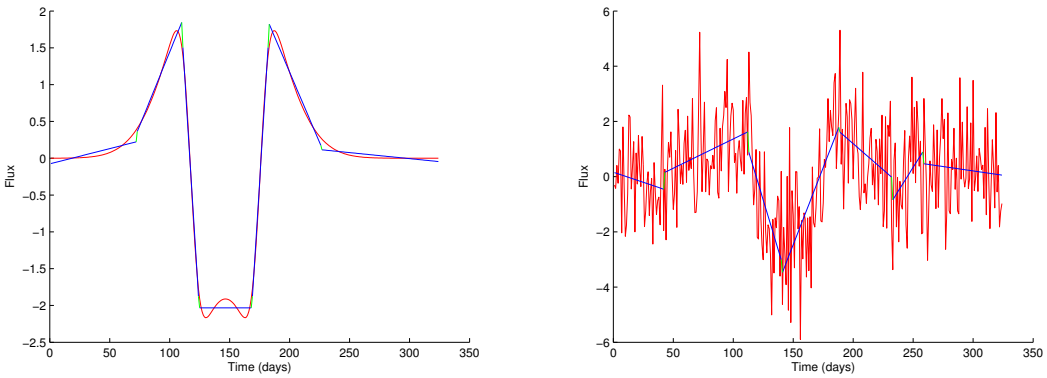


FIGURE 4.2: Left: Linear segmentation of a light curve from our dataset. Right: Linear segmentation for the same light curve with 1.5 times its variance added as Gaussian noise. The underlying structure is clearly revealed by the transformation.

and its variance at different levels. The first 4 levels of granularity of Haar wavelets were used, giving

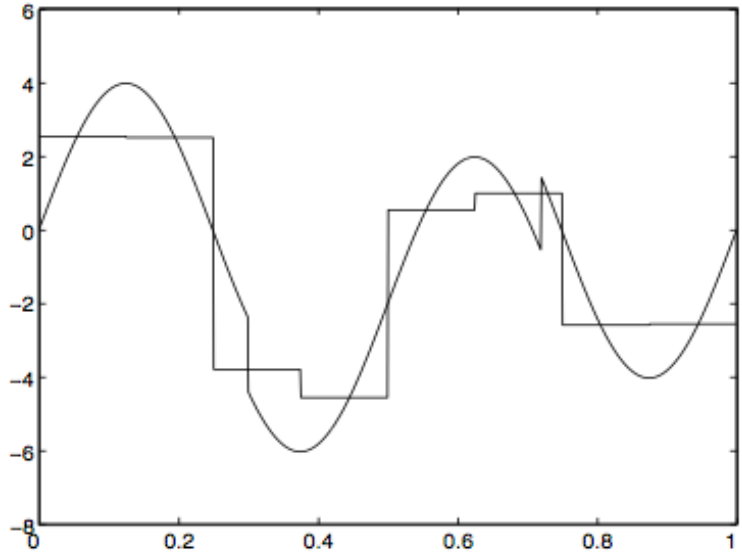


FIGURE 4.3: Reconstruction (indicated by the squarish waveform) from Haar wavelet coefficients of a signal (the sinusoidal waveform).

15 features in total (1 for the 1st level, 2 for the 2nd, 4 for the 3rd, and 8 for the 4th) corresponding the coefficients of each wavelet. The transform requires that all datapoints are equally spaced in the time domain, the width of the signal is a power of 2, and there are no gaps. Both missing data and the short ends of the signal with be filled with 0s.

4.3.4 Lomb-Scargle periodogram

An implementation of the Lomb-Scargle periodogram based on the work in Press and Rybicki (1989) was taken from astropython[1] This implementation allows us to extract as features the phase of significant peaks in the light curve and the intensity on these peaks. Computations were performed on z-normalised light curves and the strongest 5 frequencies in the signal were used as features.

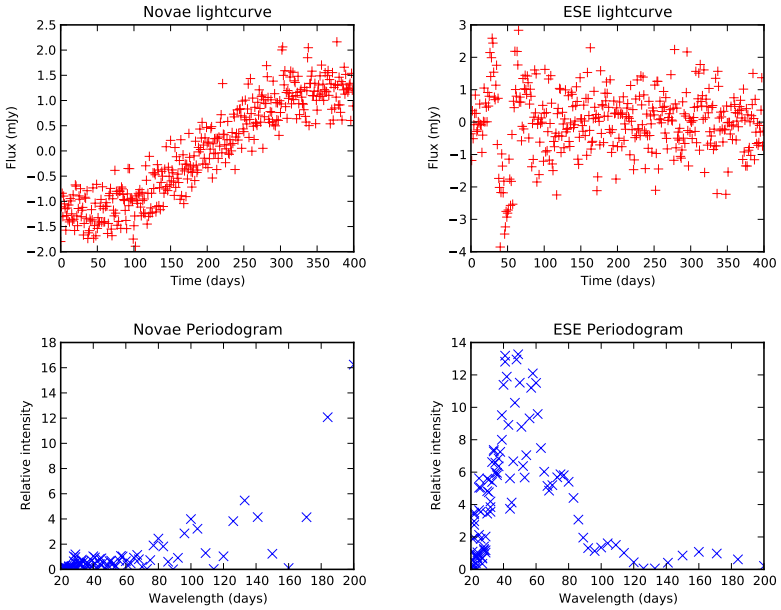


FIGURE 4.4: Spectrum produced by Lomb-Scargle periodogram for sample light curves. The Novae classes' strongest periodicities are at 200 days whereas the ESE has strongest periodicities of 40-60 days. These differences demonstrate the effectiveness of the periodogram in discriminating amongst light curves.

[1] http://www.astropython.org/blog/2010/9/Question-period-finding-packages-in-python

4.4 Experiment 1 — Undistorted data

This experiment assessed the usefulness of the features in separating the light curves without any distortions applied to the test set. Classification was performed with subtractive analysis of each of the feature sets and the results are presented as an F-Score and confusion matrix for each subtracted feature set. The results of these experiments give an upper bound on classification performance before distortions are introduced. The subtractive analysis shows the criticality of any particular feature set for achieving accurate classification.

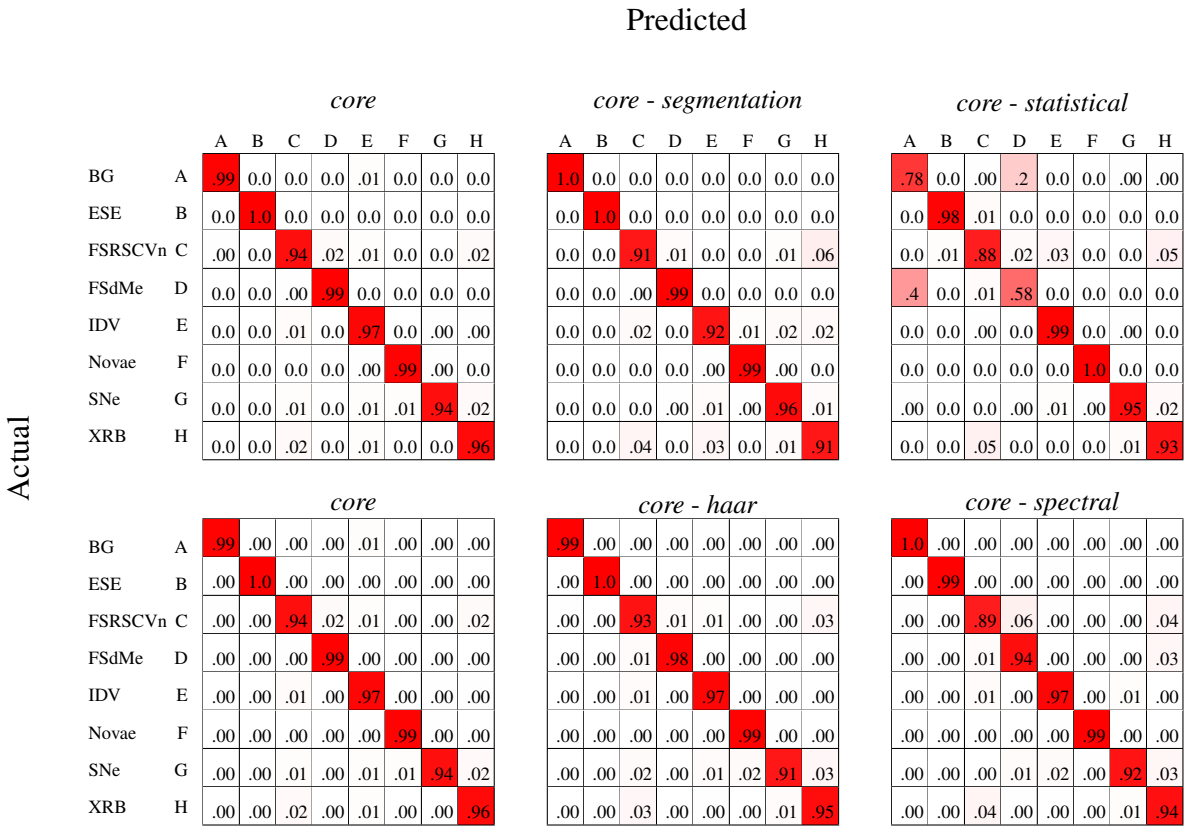


FIGURE 4.5: Figure of confusion matrices for the core classification features on undistorted light curves. The confusion matrices show that classification is near perfect unless the *statistical* feature set is excluded, which leads to misclassifications of the BG and FSdMe classes as one another.

The classification results for the combined set of features is given in the identical confusion matrices in the left column of Figure 4.5, and in Table 4.2. The results show that every class has more than 94% of test cases classified correctly, with a mean of 97%. The F-Score for the combined feature set is also 0.97, indicating both very low false positive and false negative rates. This means that in terms of the

Feature set	F-Score	std(F-Score)
core	0.97	0.010
core - {*segmentation*}	0.96	0.02
core - {*statistical*}	0.89	0.02
core - {*haar*}	0.97	0.01
core - {*spectral*}	0.96	0.02

TABLE 4.2: F-Score, and F-Score standard deviations on the 10 crossfolds

structure of the classes alone the *core* feature set is capable of accurately classifying them. An upper bound of 0.97 F-Score can be placed on all subsequent experiments.

To discuss the significance of each of the feature sets refer to the middle and right columns of Figure 4.5 and also to Table 4.2. An F-Score change for a feature set in the subtractive analysis is significant if it differs from the *core* F-Score result by more than one standard deviation. Applying this definition to the F-Scores and standard deviations in the table then, only the subtraction of the *statistical* feature set is significant. Referring to the confusion matrix for the *core* - {*statistical*} experiment the misclassifications causing this drop are between the FSdMe and BG classes. FSdMe is misclassified as BG 40% of the time, and BG as FSdMe 20% of the time. This misclassification likely occurs because the FSdMe and BG classes are superficially similar (Figure 4.6).

The FSdMe is distinct from background noise only by unpredictable, sudden, and bright flares. The Haar wavelet coefficients and Lomb-Scargle peak frequencies of the *haar* and *spectral* feature sets cannot characterise structures that are not consistent in the time domain. The *gradient* feature cannot necessarily descriminate the tall thin peaks in the FSdMe lightcurve from smaller scale peaks with the same shape occurring randomly in the BG light curve. However, the histogram of flux values can be used to tell them apart easily, as demonstrated in the stark difference in their forms in Figure 4.6.

These misclassifications demonstrate a vulnerability of the classifier to light curve structures that are locally variable - the *haar* and *spectral* feature sets become uninformative. Relying on solely the *statistical* and *segmentation* features to classify a class is not desirable since many transient classes in the real world could have similar flux and gradient profiles with otherwise very different structures. Features are needed which are locale-independent - that is, do not vary if the structure that characterises the class appears in unpredictable locations. These features need to explicitly encode these characteristic structures unlike the time independent information in a flux or gradient profile. These requirements motivated the introduction of the *shapelet* algorithm into the classification scheme in Section 5.

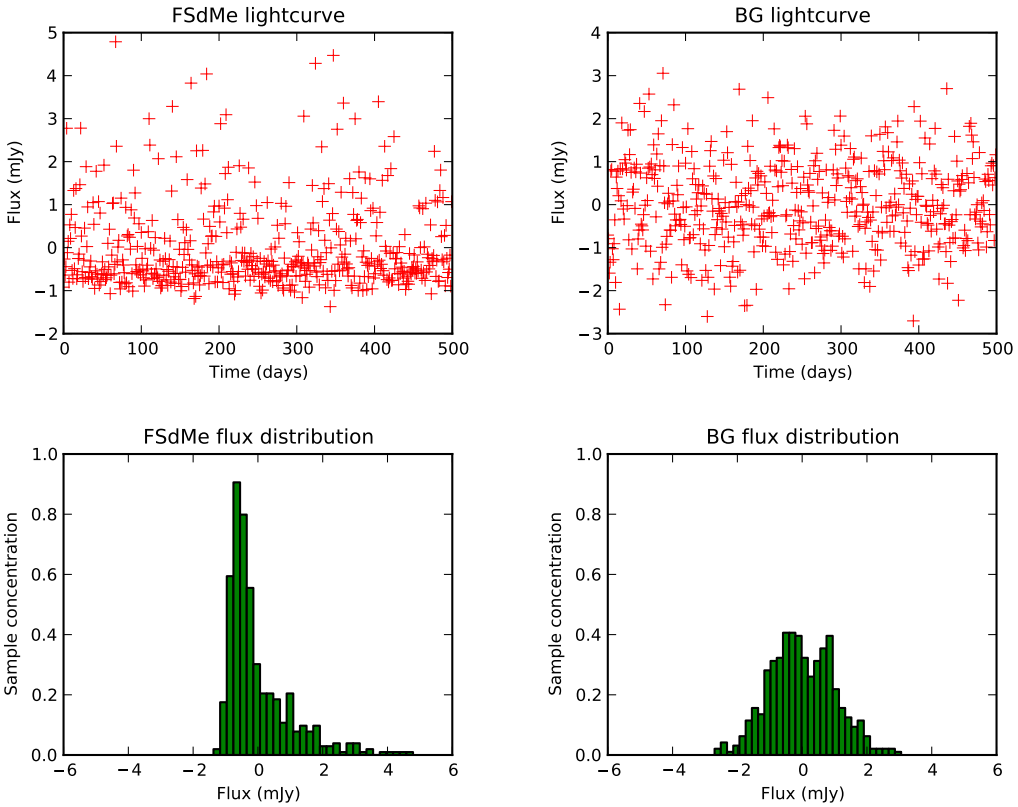


FIGURE 4.6: Light curve plots and flux histograms of centered FSdMe and BG lightcurves demonstrating the difference in the flux distributions. The BG flux distribution is much more even that that of the FSdMe

4.5 Experiment 2 — Introducing gaps into the light curve

This experiment assessed the impact of introducing small, randomly distributed gaps into a light curve on classification performance. 10, 25, 50, 75 and 90 percent of the signal was taken out in separate experiments and classified using undistorted training data. Subtractive analysis of each of the feature sets is provided as plots and confusion matrices. For this experiment only the confusion matrices for the subtracted *spectral* and *statistical* as well as the combined *core* feature sets are of interest. The other confusion matrices are omitted. The analysis concerns the extent to which classification performance is affected by the use of differently distorted training and test sets as well as the loss of the information in the light curve through the distortions.

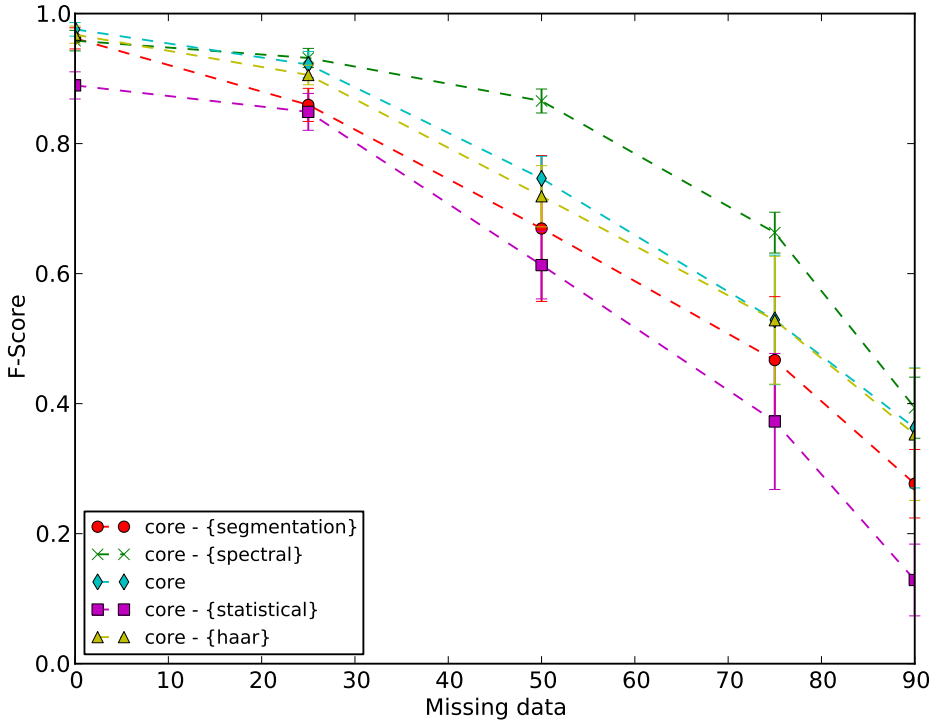


FIGURE 4.7: Plot of F-Score versus missing data with **undistorted training data and distorted training data**. All feature sets lose F-Score quickly except for subtracted *spectral*, only gradually falling until 50% missing data.

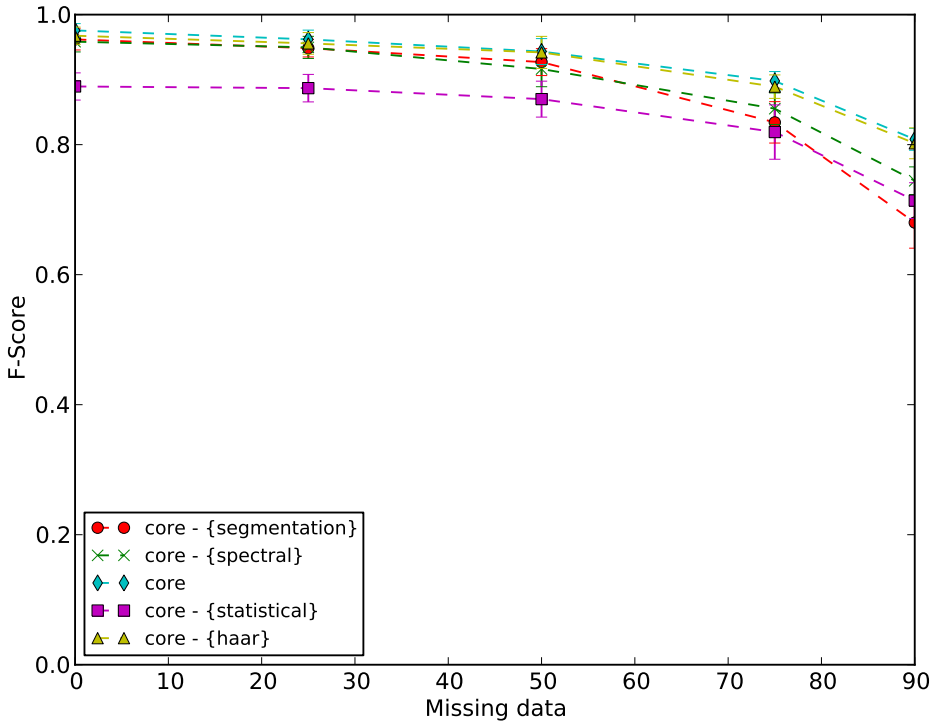


FIGURE 4.8: F-Score versus missing data with **equally distorted training and test datasets**. F-Score is consistent on all feature sets up to 90% missing data.

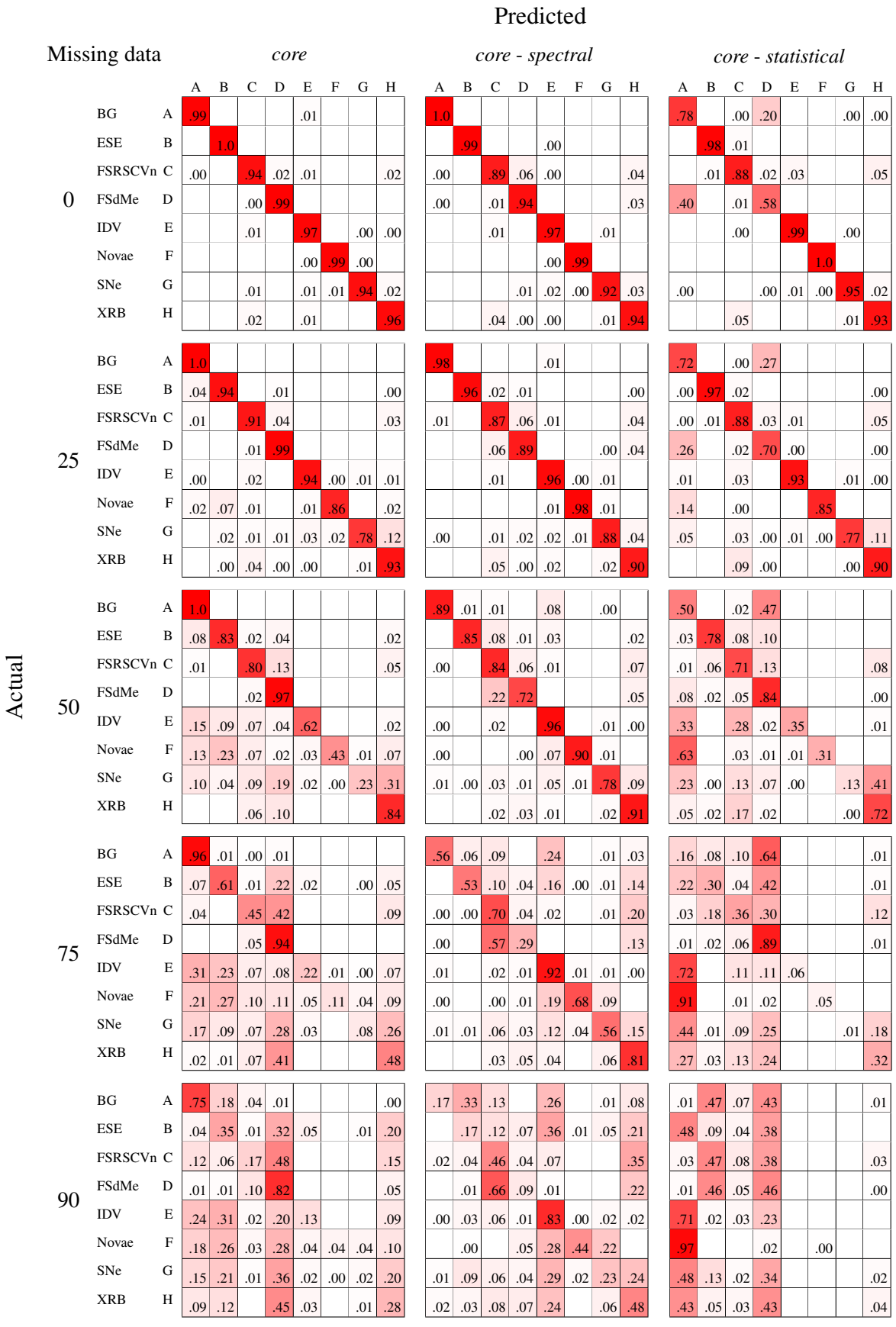


FIGURE 4.9: Selected confusion matrices for the missing data experiment with **undistorted training and distorted test data**. The middle column of confusion matrices shows that the exclusion of the *spectral* feature set improves classification performance on the IDV, Novae and SNe classes for 50% and 75% missing data. The right column shows the importance of the *statistical* feature set for correctly classifying all classes.

Figure 4.7 shows classification results when using undistorted training and distorted test set sets. Comparing this plot to Figure 3 tells us how much the disparity in feature values using unequally distorted training and test sets (I will refer to this as *asymmetric* and otherwise *symmetric*) affects classification performance. At 50% missing data the best feature set for *symmetric* training/test sets has dropped by less than 0.01. For the same amount of missing data the asymmetric test sets have an F-Score of 0.9 for the best feature set, a drop of 0.07. At 75% missing data the differences are larger still with F-Scores of 0.9 and 0.67 for the symmetric and asymmetric classification scenarios respectively. These are substantial differences in performance and indicate that the majority of misclassifications arise as the result of the disparity in the feature values across the asymmetric training and test sets.

The F-score drop which is not the result of the training/test set decision can also be found in Figure 3 as the change in F-Score as missing data is increased. As stated earlier F-Score does not change substantially for even up to 50% missing data, the F-Score of 0.96 lying within a standard deviation of the F-Scores at 0% missing data. Only 0.07 and 0.15 F-Score are lost for 75% and 90% missing data. Examining the remaining structure of the light curves at 90% missing data in Figure A.3 suggests that this F-Score drop is the result of information loss and unavoidable. What this means then is that the key to minimising the impact of missing data on classification performance is to solve this problem of training and test set asymmetry.

The cause of the misclassifications is that the values of the features are changing under the introduction of distortions. If the features were **invariant** to missing data, that is, their values are the same so long as no information has been lost from the signal, then we should expect classification results similar to those in Figure 3. Invariance to a distortion then is a desirable property of a feature. Examining the confusion matrices in Figure 4.9 for our asymmetric train/test set experiment will give clues as to which feature sets have the least and most invariance.

By far the most substantial increases in F-Score through subtractive analysis is for *core* - *spectral*, with an F-Score increase of 0.15 at 50% missing data and 0.2 at 75% missing data. The confusion matrices for that subtractive analysis compared with the combined *core* feature set shows misclassification occurring the most on the Novae, SNe and IDV and XRB classes for 75% and 90% missing data. Mass is shifted to essentially all other classes in the row indicating that the value of the feature is essentially unpredictable as more missing data is introduced. The FSdMe, FSRSCVn, BG and ESE classes are probably less severely affected because the classifier did not relate their highly locally variable structures to periodic features with the undistorted training data. The *spectral* features show invariance to noise in the next

experiment, so omitting them entirely is not the best course of action. A regression step to try to make the test data look more like the training data could improve classification performance.

As in the previous experiment the *statistical* feature set is the most critical to good classification accuracy for the asymmetric train/test missing data experiment. The confusion matrices in Figure 4.9 show that the misclassification rates for every class increase with the removal of the *statistical* feature set. A histogram remains a good approximation of itself when even a large fraction of its values are removed, so it is not surprising then that the *statistical* feature set displays some invariance to missing data. Designing features that are not just rough approximations under missing data but are very close approximations or completely invariant will greatly improve classification performance.

In summary, the features used are still able to accurately classify the dataset with symmetric test data, with > 0.9 F-Score for 90% missing data. The primary cause of misclassification then is the disparity of feature values from undistorted training to distorted test sets. There are a few possible avenues to improving this situation for missing data (but also may apply to later experiments):

(1) Pre-processing the test data to make it look more like the training data. For coping with missing data options include a number of regression approaches. Two examples are simple linear interpolation and Gaussian Process regression. Explorations of the scalability and effectiveness of such approaches are a topic for future work.
(2) Use features that are more invariant to distortions. The *shapelet* approach to time series classification might be a good candidate for invariance to missing data because it relies upon direct matches to structures in the signal — not distributions across it.
(3) Assume the kind of distortions present in the test data and apply similar distortions to the training data to reduce the strain on classifier rules. While it is perplexing to reduce the quality of the training data in order to improve classification, Comparing Figures **??** and demonstrate that this an effective approach. It is likely that this approach will be viable for ASKAP because the amounts of noise and missing data will be difficult to know ahead of classification time.

4.6 Experiment 3 — Limiting the amount of signal observed

This section assessed how early we can hope to classify the transient classes in our dataset. In this experiment both the training and test data are cropped to a particular percentage from the start of the light curve. Modifying the training data is acceptable for this particular kind of distortion since unlike noise or missing data, in a sliding window classification context we know how much of an event has transpired and what hence what training set to apply. The results here show that early classification is

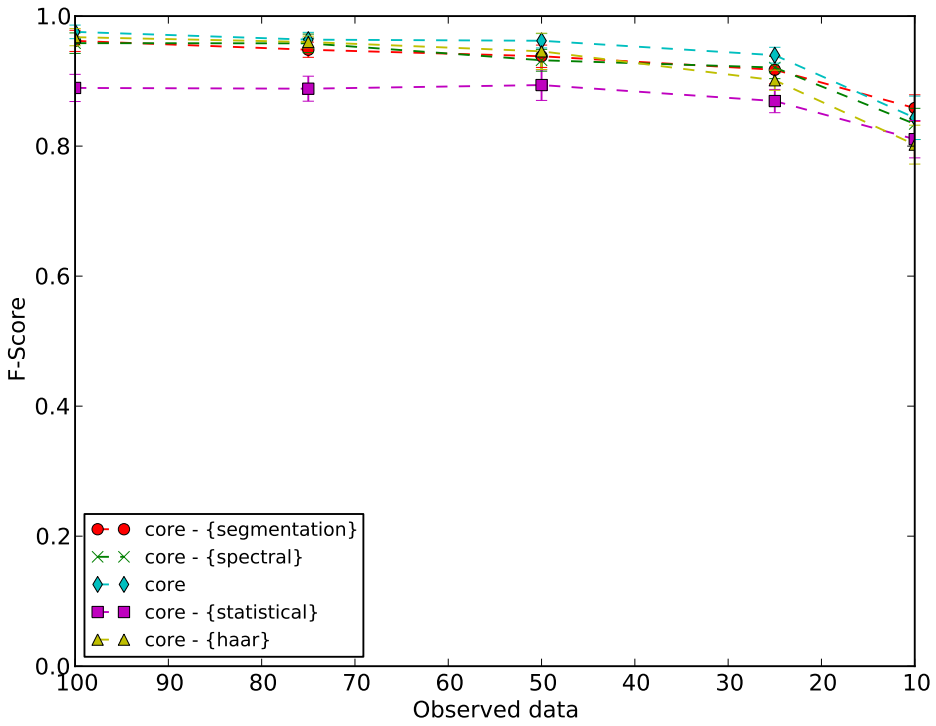


FIGURE 4.10: Plot of F-Score versus observed data with **equally cropped training and test data**. F-Score stays consistent up to 10% observed data

possible on the light curve collection provided there are no distortions and cropped training data are also used. At least, there is an upper bound of at most 0.9 F-score to be placed on classification performance. Up to 10% available data the F-Score does not change much from the results for the full light curve.

It is possible that the correct early classification of several of these classes is dependent on a subtle structure that would be lost in realistic astronomical data conditions. A better test of each classes' early classification ability is in experiment 5, evaluating combined distortions.

4.7 Experiment 4 — Modifying the signal to noise ratio

This experiment involves the addition of noise into our dataset by computing the signal standard deviation and adding a fraction of that value as Gaussian distributed noise to the light curve. The fraction of variance added becomes the light curve's noise-signal variance ratio, very similar to the signal-noise ratio used by Astronomers to describe how clear a signal is in noisy data. The magnitude of the noise-signal variance ratio determines how likely the characteristic subsequences of a class are still present. What this means generally for classifying our transient classes is hard to define, but an idea of just how much of the signal remains can be found in Figure A.2 in Appendix A. Light curves at 0.5 noise still have very clear structures and are still clearly identifiable by a human. The same is true for 1.0 noise but the with the two flare star classes FSRSCVn and FSdMe beginning to look similar. The 1.5 noise only the most distinctive structures remain. At noise 3.0 artefacts of the original signal are still visible but manual classification becomes very hard. The first question to explore in this experiment is how much the introduction of noise leads to misclassification with the same distortions applied to training and test data i.e. how much information is lost as a result of noise. The second is to what extent the use of undistorted training data and distorted test data causes misclassification, in light of the information loss.

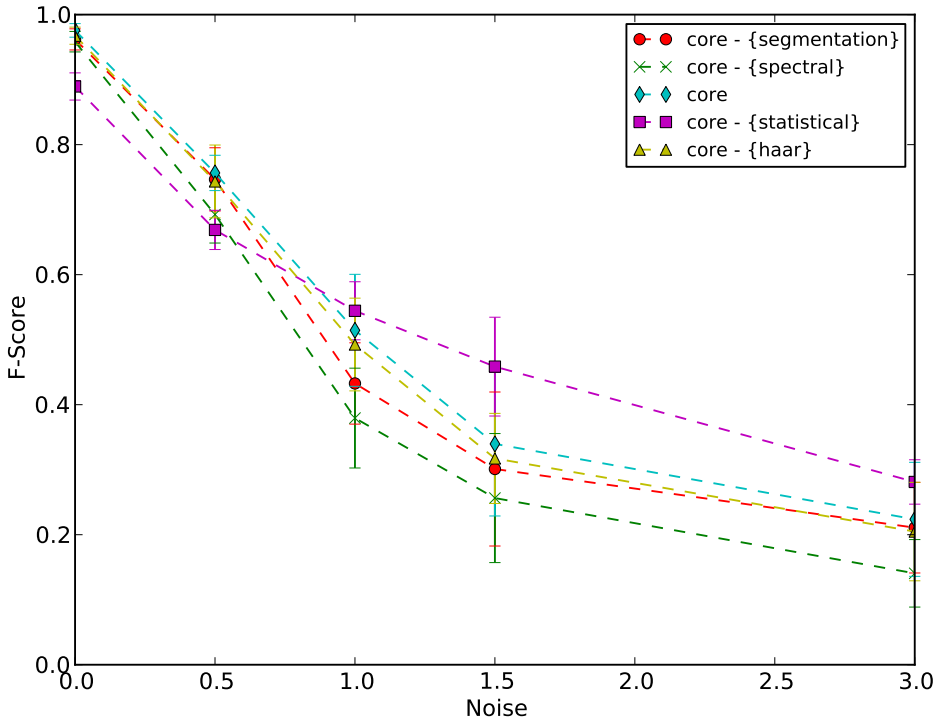


FIGURE 4.11: Plot of F-Score versus noise-signal variance ratio with **undistorted training data and distorted test data**. The classification F-Score decreases rapidly to 0.5 for the best feature set at 1.5 noise-signal variance.

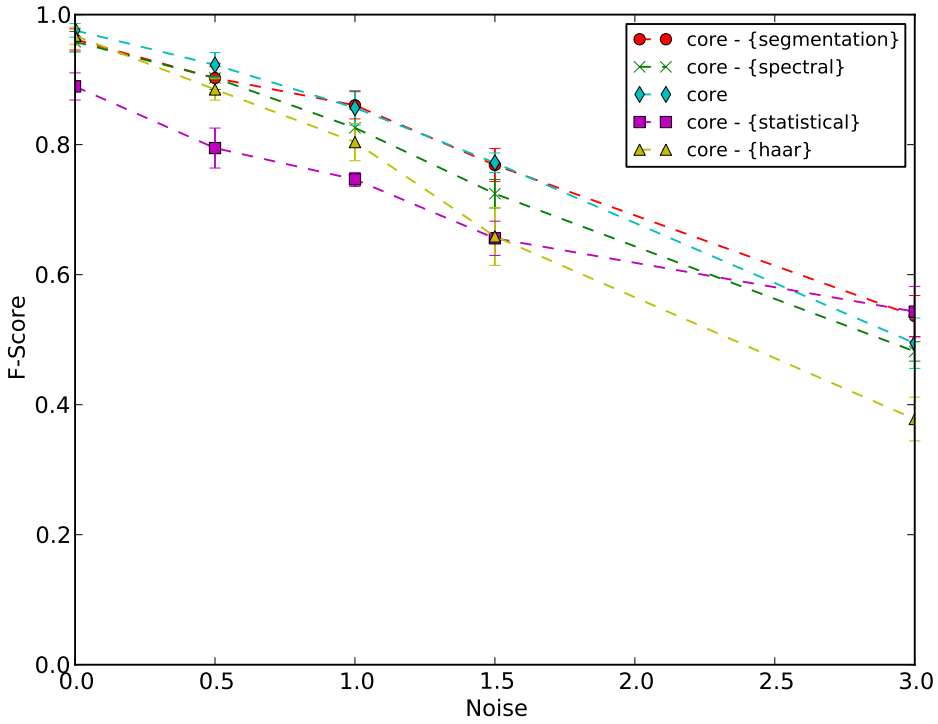


FIGURE 4.12: Plot of F-Score versus noise-signal variance ratio with **equally noisy training and test data**. The plot shows a linear trend of F-Score as the factor of noise is increased and is significantly higher than when using undistorted training data.

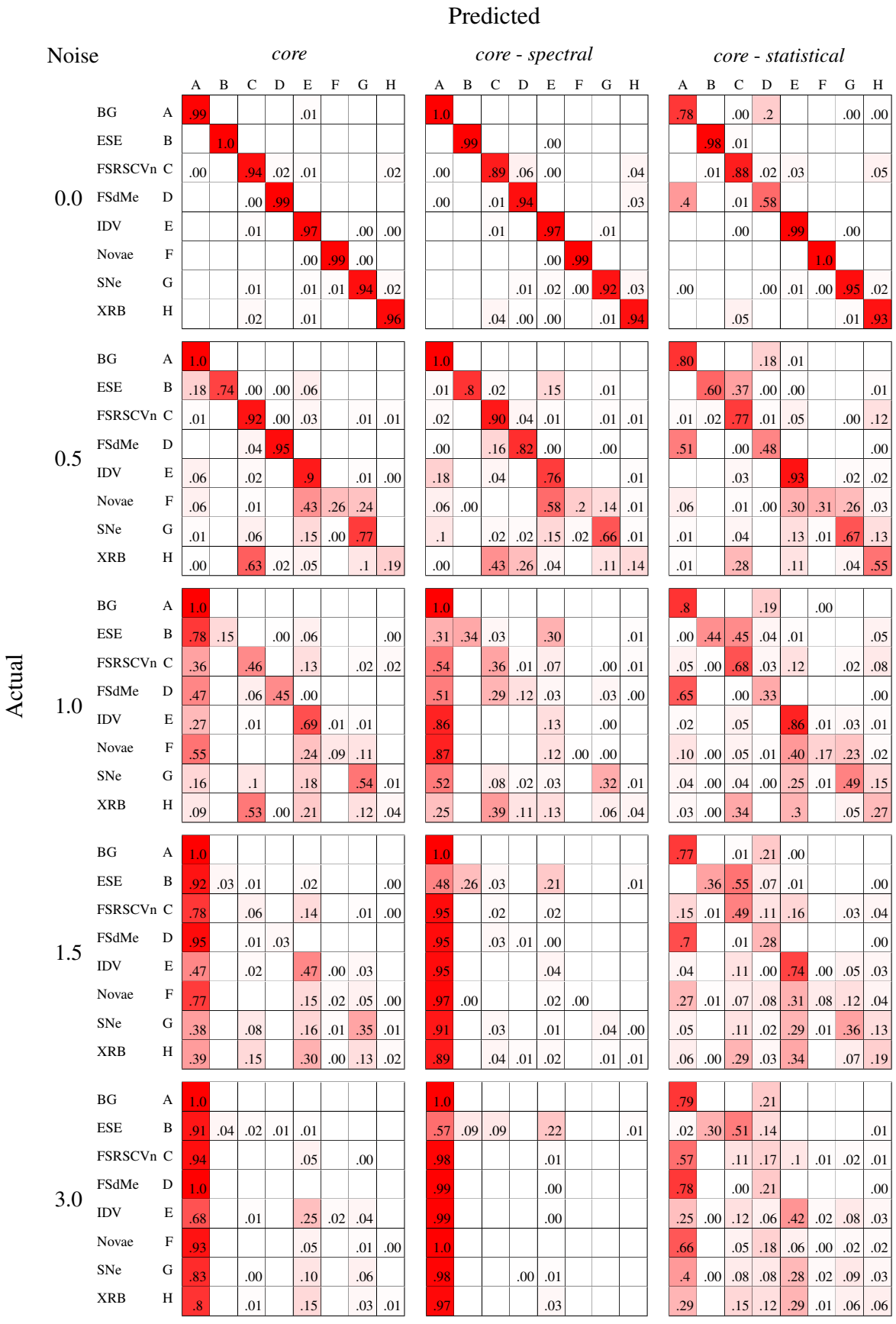


FIGURE 4.13: Selected confusion matrices with **undistorted training data and noisy test data**. The left and middle columns show the strong trend of the *statistical* feature sets' inclusion to cause misclassificatons to the BG class. The right column shows improved classification when *statistical* is excluded.

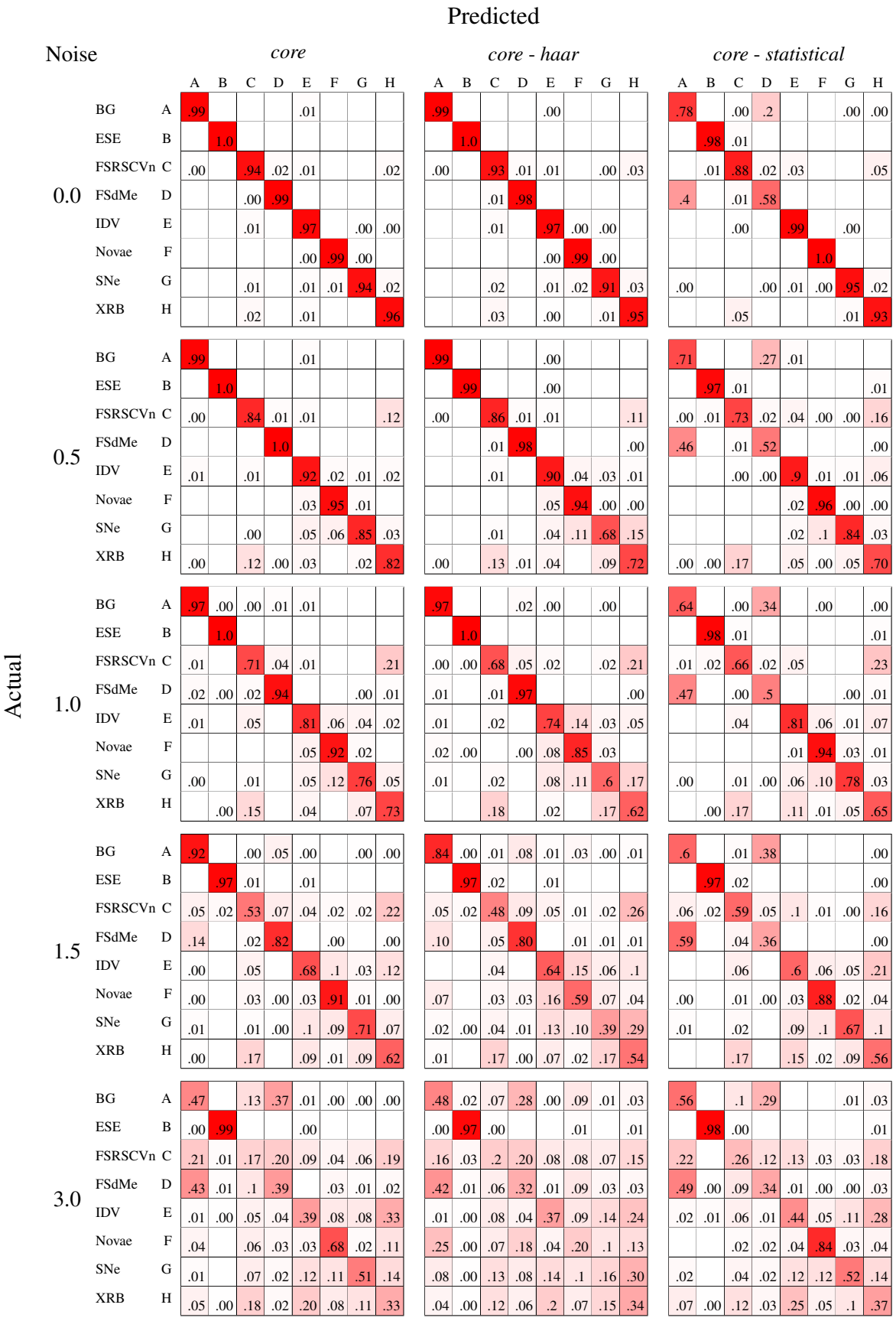


FIGURE 4.14: Selected confusion matrices with **noisy training and test data**. The middle column shows that classification performance drops and increases with the exclusion of the *haar* and *statistical* feature sets respectively.

Figure 4.11 shows the results of a classification experiment using noisy training and test data. There is an approximately linear decrease in F-Score as the noise-signal variance ratio (NSV) is increased. For the best performing feature set (*core* - {*segmentation*}) the F-Scores are significantly higher than those for using undistorted training data in Figure 4.11. This means that misclassification with regards to noise is not solely the result of a disparity in feature values across training and test data. Either the noise is causing information loss and making classification impossible, or our features are insufficient to recover the original structures now hidden in noise. As discussed in the introduction to the experiment Figure A.2 in Appendix A shows that for a NSV of 1.5 that the characteristic structures of most classes are not easily seen, and at 3.0 it is difficult for a human to discriminate amongst the classes that do not have large scale structures like the SNe, ESE and Novae classes. This observation along with Figure 4.14 partially supports the case for information loss as classes with locally variable and small scale structures like the FSdMe, FSRSCVn and XRB classes are either misclassified as one another or are classified as noise. Only the ESE, SNe and Novae classes - the classes with large scale structures - have more than 50% of their test cases classified correctly at 3.0 NSV.

Comparing the performance in the middle and left columns tells us the *haar* feature set is important for identifying the classes with large scale structures such as Novae and SNe and is an important feature for dealing with noise. Haar wavelets average large sections of the light curve to arrive at the coefficient values, making them like a primitive noise filter.

The results for noisy training and test data put an upper bound of 0.8 F-Score on any experiment involving 1.5 signal-noise variance for our feature set since classification accuracy will only be worse when using undistorted training data, removing data and applying a power law distribution. Figure 4.11 shows the performance of the classifier using undistorted training and noisy test data, as the noise-signal variance ratio is increased. The F-Scores of all feature sets drop rapidly for any amount of increased noise. From 0.5 noise-signal variance there is an F-Score drop of 0.2, falling consistently up to 1.5 with a gradual decline to 3.0 with an F-Score of about 0.3 for the *core* feature set.

The left and middle columns of confusion matrices in Figure 4.13 shows a very strong trend to misclassify every class as noise as the NSV is increased to 1.0 and higher. Comparing these two columns to the right column suggests that the cause of the misclassification is a smoothing of the flux histograms used by the *statistical* feature to appear like the noise of BG class at NSV 0. These results differ a lot from those seen in Figure 4.14 and illustrate that there is a great deal of shift in the feature values as noise is introduced.

Noise is a serious issue for the classifier both because because it blurs the distinctive structures between classes that were already similar (XRB and FSRSCVn, FSdMe and BG) and because the Random Forest cannot cope with the train/test disparity in the structure of the light curves under the distortion. The *haar* feature set demonstrated a significant performance increase in classifying noisy classes with large-scale distinctive structures like ESEs, Novae and SNe. The *statistical* feature set, despite having some robustness to missing data, is highly sensitive to the introduction of noise. No other feature set in subtractive analysis led to a significant shift in classification. As in all the other distortions experimented with in this thesis, advance knowledge of their severity would help classification greatly. Unfortunately, the noise-signal variance of every test case in the ASKAP scenario will be different. Exploring noise pre-processing techniques and how well they remedy the training-testing classifier disparity is an important direction for future work.

4.8 Experiment 5 — Power law applied to signal, 50% data missing, 0.75 Noise to signal variance ratio

This experiment assessed the impact of a combination of distortions on classification. A moderate amount (0.75 noise to signal variance) of noise was introduced into the signal and 50% of the data was removed as small randomly distributed chunks. This data in this experiment is most similar to read world astronomical data and the results are the best indicator of how these supervised classification system would perform in the VAST pipeline.

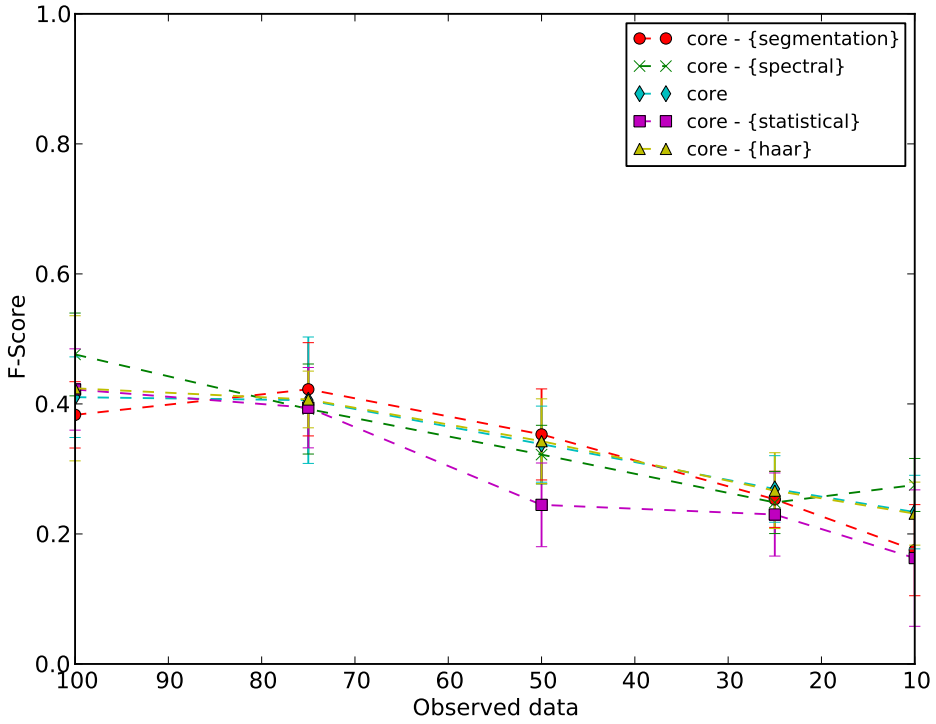


FIGURE 4.15: Plot of F-Score versus percentage of light curve observed with **undistorted and cropped training data and distorted and cropped test data**. The trend of F-Score for all feature sets is from 0.4 to 0.2 from 100% to 10% observed data.

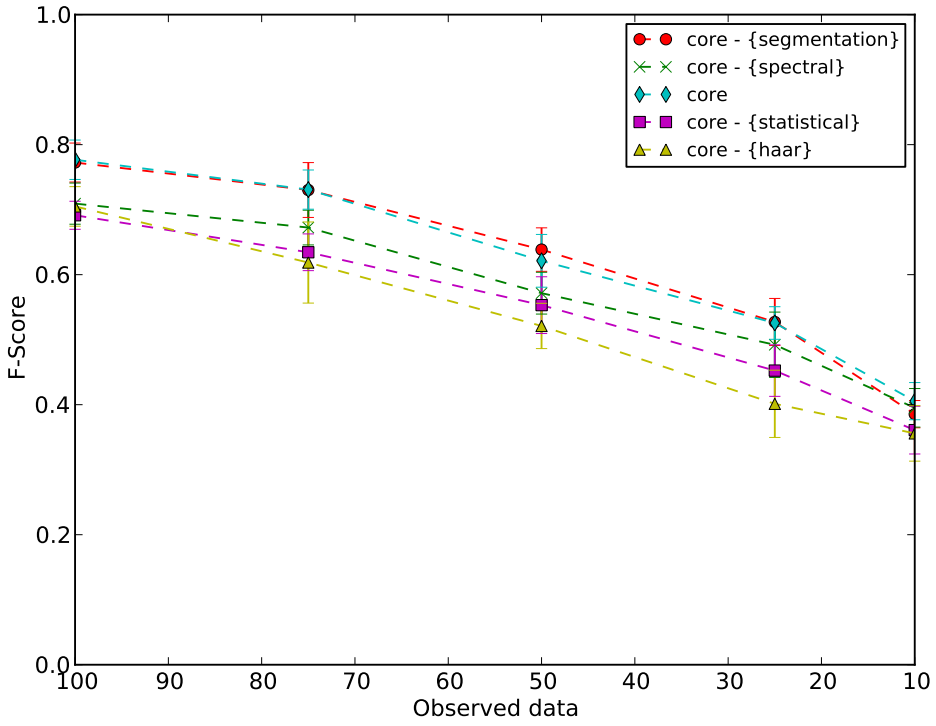


FIGURE 4.16: Plot of F-Score versus percentage of light curve observed with **equally distorted training and test data**. The trend of F-Score is from 0.8 to 0.4 from 100% to 10% observed data for the best feature set.

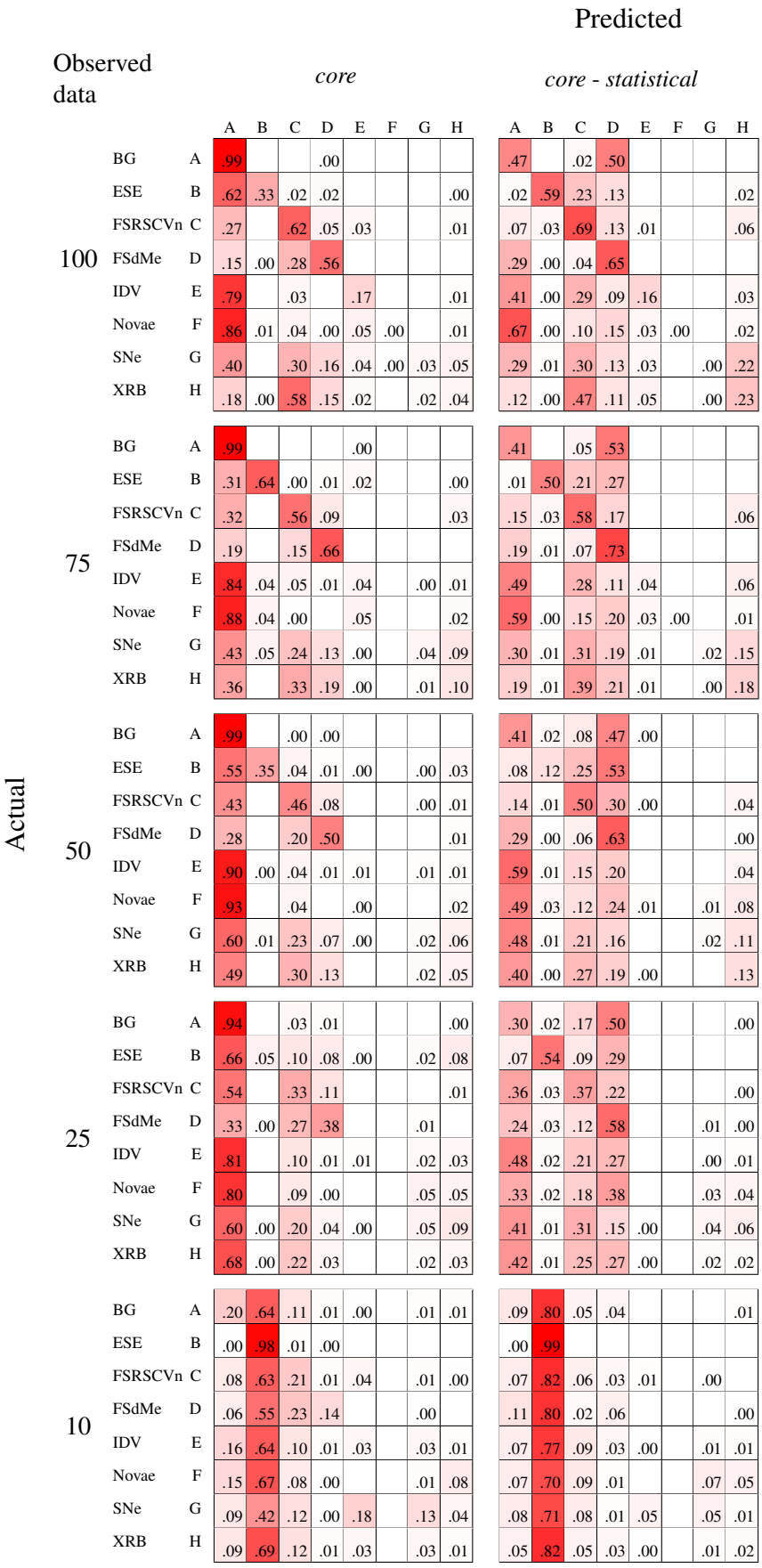


FIGURE 4.17: Selected confusion matrices for all distortions experiment with **undistorted training and distorted test data**. The two columns show how the exclusion of the *statistical* feature set improves classification performance.

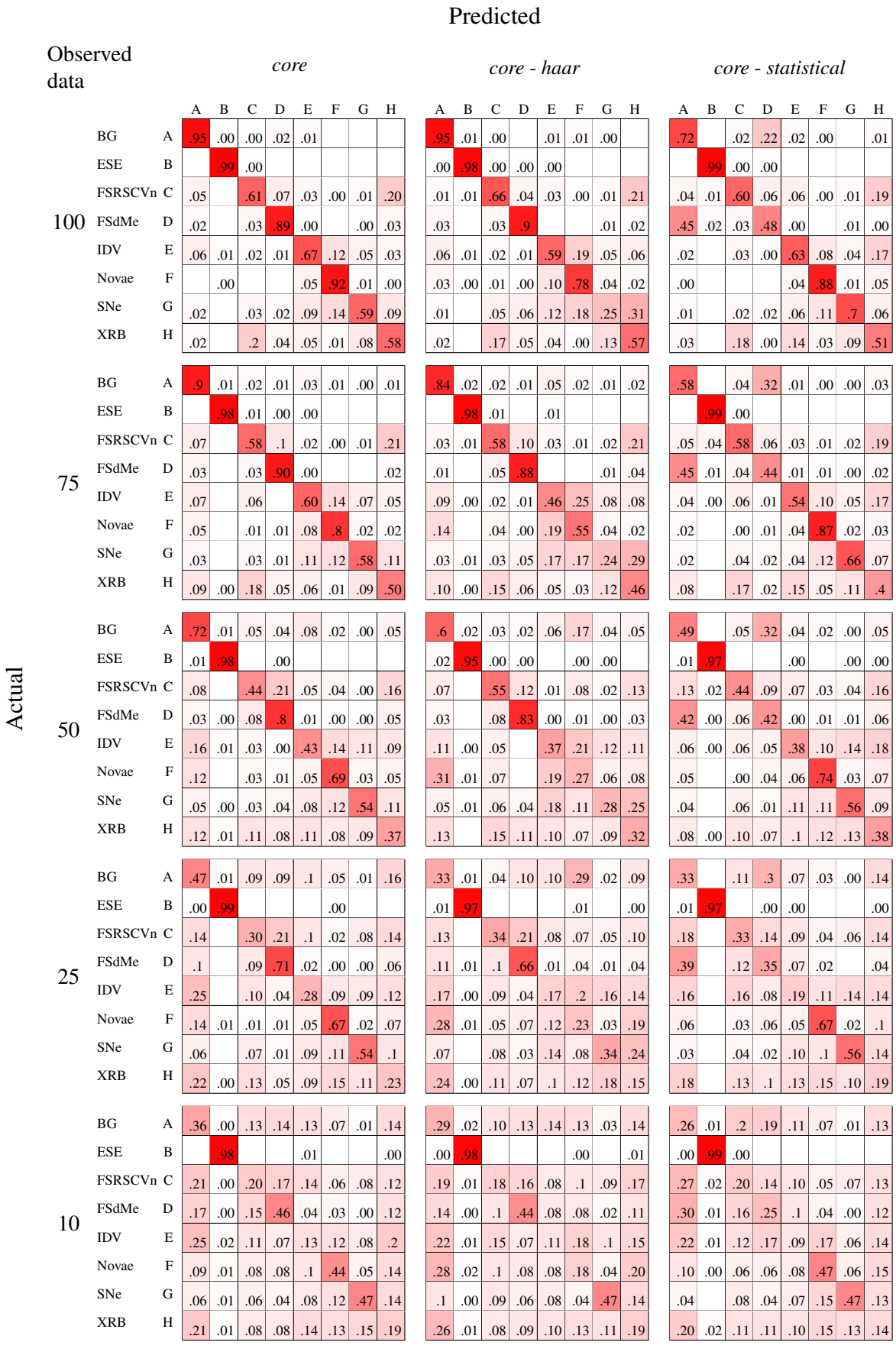


FIGURE 4.18: Selected confusion matrices for the all distortions experiment with **undistorted training and distorted test data**. The middle column shows that *haar* wavelets give slightly better classification performance on the Novae and SNe classes.

Figures 4.15 and 4.16 show the plots of F-Score versus percentage of test case observed with undistorted training and distorted test data (asymmetric), and equally distorted training and test data (symmetric) respectively. Comparing the two plots gives an indicator of the extent to which the shifting of feature values is responsible for misclassification. We should expect that since this experiment involves a combination of noise and missing data, two distortions that the asymmetric training/test classifier was very sensitive too, then the classifier should have the same difficulty in this experiment. This is verified by the differences in the F-Score for the *combined* feature set, 0.4 lower with asymmetric training and test sets. This difference is consistent and large for all amounts of observed data.

The *haar* and *statistical* feature sets are critical for classification with the symmetric training/test sets. Removing the *haar* feature set causes F-Score to fall 0.1 for all values of observed data except 10. The left and middle columns of confusion matrices in Figure 4.18 show in the 3rd 4th and 5th rows that Haar wavelets are critical amongst our features for identifying Novae and SNe - light curves with large scale structures in the time domain. The left and right columns affirm the observations in the no distortions experiment that *statistical* features are necessary for correctly classifying the FSdMe class.

The confusion matrices in 4.17 again resemble the confusion matrices of their combined distortion results for noise and missing data in Figures 4.9 and 4.13. Misclassifications to the BG class are most frequent with the *core* feature set, and to the first four columns with the *statistical* feature set subtracted. The classifier shows that correct identification at a 50% rate of FSdMe and ESE classes is still possible. There are far too many false positives and misclassifications for this classifier to be useful in the VAST pipeline however.

4.9 Conclusion

These explored the impact that distortions have on classification, both in terms of information loss and the inadequacy of our features to separate distorted light curves, and also how the shift of feature values under distortions causes misclassification when using undistorted training data.

The first experiment demonstrated that our classifier can effectively separate the light curves when no distortions are applied. The F-Score of 0.97 in that experiment means the structures of the light curves themselves do not lead to misclassification. The experiment did indicate a vulnerability of the classifier in that it relied on the *statistical* feature set to accurately classify the FSdMe class. Distortions impacting

the *statistical* feature, such as the introduction of noise will mean the FSdMe class can no longer be discriminated accurately.

The second experiment, introducing gaps into the light curve, showed that the feature set can effectively separate light curves with even large amounts of missing data, but only when the training set is equally distorted. F-Score did not fall below 0.9 for the best feature set until 90% missing data. The results show that the feature values shift a lot under distortions, in particular, the *spectral* feature set, reducing F-Score by 0.1 for 25-75% missing data. The *statistical* set showed the greatest invariance to missing data. Missing data is a serious issue for the classifier.

The third experiment involved limiting the amount of the light curve observed in both the training and test sets and attempting classification. The F-Score did not fall below 0.9 for all percentages of the light curve observed except for 10%. Since there may be subtle features that allow early classification this experiment does not say much for early classification in realistic data conditions. At least however there is an upper bound of at most 0.9 on classifying our light curves if we have observed at least 10% of the signal.

The introduction of noise into the light curves in the fourth experiment showed that our features cannot very accurately classify noisy light curves. For a noise to signal variance of 1.5 with equally distorted training and test sets the best feature set achieves an F-Score of 0.8. The *haar* feature set performed best when large amounts of noise were present, improving classification primarily for the Novae and SNe classes and are marginally more important than the *statistical* feature set. The kind of noise seen at 1.5 noise to signal variance is typical of astronomical data (see Figure A.2 for sample light curves). Classification performance is worse still when using clean training data. The noise in the light curves leads most classes to be classified as background noise.

Finally, the combined distortions in experiment 5 show that both the information loss seen with the introduction of noise and with equally distorted train and test data is compounded with 50% missing data and a power law distribution. The F-Score is 0.2 lower than the results of the noise experiment for all amounts of observed data but 10%. This means that an F-Score greater than 0.8 is not possible using undistorted training data even with the full light curve available. For that experiment, classification performance at 100% observed data is 0.4 F-Score. This decreases to 0.2 F-Score at 10% observed data. Similar patterns of misclassification for the noise and missing data experiments are seen.

F-Scores of 0.4 are unsuitable for use in the VAST classification pipeline. Potential improvements to the classifier involve preprocessing techniques to recover the structure of the original signal from a distortion, and also to make the test data look more like the training data. This will reduce both the shift of feature values which is the main cause of misclassification, and may improve the classification baselines with distorted training and test data. Examples of directions to pursue are noise filters and smoothing, and regression techniques (simple linear interpolation or more complex statistical models like Gaussian processes) for filling in missing data. Finally, making assumptions about the nature of the distortions that will be present in the test data and using already distorted training data, although paradoxical, could improve classification performance.

CHAPTER 5

Shapelet representations of time series

5.1 Overview

A *shapelet* is a distinctive subsequence of a class of time series objects within a dataset. They are interesting to transient classification because they explicitly encode discriminative substructures of the transient events. This in contrast to the features used in the experiments of Chapter 4 which either summarised large sections of the light curve (*haar*), converted the light curve to an entirely different representation (*spectral*) or summarised distributions across the light curve (*statistical* and *segmentation*). Shapelets are used to produce features by giving to a classifier the *subsequence distance* of the shapelet to a test case. Distances will be close to 0 if the shapelet appears in the test case, and substantially greater than 0 otherwise.

Since no other feature explicitly contains the substructures of the transient classes it was hoped that shapelets would improve classification performance with the *missing* and *noise* distortions. This was expected for the experiments using equally distorted test data because the shapelets might recognise a transient structure better or in a complementary fashion to the other features. Additionally, it was expected that performance would improve when using undistorted training data and distorted data. This is because the distance of a shapelet to a matching section of a test case would change much less than the values features generally involving the entire light curve. Finally the values of the shapelet features were expected to be much more consistent for light curves have substructures repeating at unpredictable intervals (XRBs, FSRSCVns, FSdMEs) than the features of Chapter 4

The shapelet extraction algorithm was implemented as outlined in Section 5.2 and the best single shapelet for each class was extracted forming the *shapelet* feature set. An clustering algorithm to discover a number of useful shapelets, not just the absolute best, was implemented (referred to as

20shapelets). This feature gave marginal performance increases of between 0 and 0.1 F-Score over the *shapelet* feature set for all distortions. Unfortunately due to inherent complications with the shapelet algorithm the classification accuracy did not improve above the *core* feature set in most cases. These complications included the inadequacy of binary entropy to separate the XRB and SNe classes, false positives when matching shapelets to gappy data and the use of undistorted data for shapelet extraction resulting in shapelets which are then not robust to the introduction of noise into the light curve. There were some exceptions to this, with a marginal (0.05) increase in F-Score at 10% observed data with undistorted light curves, and a paradoxical result of a greater than 0.1 F-Score improvement in classification accuracy when combining the *core* feature set from Chapter 4 at 50% and 75% missing data. A final note on the shapelet algorithm not evaluated by experiment but important in the context the application of this thesis to the VAST pipeline is that the shapelets can still classify light curves with unknown start and end points. All the features in Chapter 4 will likely not work when they are trained on full light curves and used to classify partial ones. Shapelet classification performance should not be affected so long as the Shapelet still appears in the test case. Provided a number of shapelets are used this is fairly likely.

5.2 Shapelet extraction and experiments

The shapelet extaction algorithm used in these experiments is the same as that given by Ye in Ye and Keogh (2009). The algorithm is a brute force assessment of the discriminative power of every subsequence of every time series in the dataset. Discriminative power for each subsequence is assessed by computing a distance measure called the subsequence distance (Equation 5.1 to every light curve in the collection:

$$S(x,y) = \frac{\sqrt{\min_{i=1\ldots L_y-L_x} \sum_{p=1\ldots L_x} (x_i - y_{j+i})^2}}{|x|} \tag{5.1}$$

$S(x,y)$ is the subsequence distance of time series x and y, L_x and L_y their respective lengths, and x is the shortest (or equal shortest) of the two sequences. This equation represents the minimum Euclidean distance computed over all alignments of the two sequences. When applied to missing data in this experiment the subsequence distance will be forced to match at least 5 datapoints. If this constraint were not introduced then the subsequence distance would match the shapelet onto gaps in the light curve, frequently returning 0 distance. The light curves are then ordered on a *separation line* (Figure 5.1) according to their closeness to 0 distance from the subsequence. A shapelet is said to be discriminative

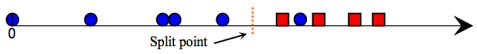


FIGURE 5.1: Separation line of classes in a single shapelet evaluation step of the brute force shapelet extraction algorithm. The better the separation of classes for some splitting point on the separation line, the more discriminative the shapelet. The separation here is nearly perfect with 1 test class out of place.

if there is a way to partition the separation line such that there is a clear separation of the classes in the dataset across the partition. This goodness of separation is quantified by the binary entropy in Equation 5.2:

$$E(D) = -\frac{n_a}{N}log(\frac{n_a}{N}) - \frac{n_b}{N}log(\frac{n_b}{N}) \tag{5.2}$$

D is a side of a partition of the separation line. n_a and n_b are the number of time series labelled as class a and class b respectively and N is the total number of time series in the split. An entropy of 0 indicates perfect separation. To determine if there exists a good partition of the separation, the value $E(D_l) + E(D_r)$ is computed for all possible partitions of the separation line into left and right sides D_l and D_r. The minimum of these values becomes the final indicator of discriminative quality for a shapelet

The most discriminative shapelet for any member of a particular class output from the procedure given above becomes a shapelet feature for that class. The distances of the shapelet under subsequence distance to a light curve are used as features in both training and testing in a supervised classifier.

It is likely that there are many distinct subsequences of our light curves that are useful in discriminating them from the other classes. A simple extension to the shapelet algorithm that can identify these different but still useful subsequences is to cluster the shapelets according to their subsequence distance from one another. The shapelets do not necessarily have to extracted and evaluated on the same dataset. An additional extension to the shapelet algorithm, unfortunately not included in these experiments due to time constraints, is to extract shapelets from undistorted time series and then to evaluate their discriminative power (as above) on the same time series but with noise added or with data missing. This would make the extraction algorithm select shapelets that still have as much information as possible about the original discriminative structure but are also robust to distortions.

A limitation of the binary entropy approach to shapelet extraction is that shapelets will be chosen if they separate only the class they belong to from the other classes. If two classes share a distinctive structure then that will never be chosen as a shapelet. Another extension of the shapelet algorithm to allow

the extraction of shared discriminative subsequences is with multi—class entropy, proposed by Mueen in Mueen et al. (2011).

$$E(D) = -\sum_{i=1}^{C} \frac{n_i}{N} log(\frac{n_i}{N}) \tag{5.3}$$

Where n_i is the number of time series with class label i out of C labels appearing in the dataset, and N is the total number of time series. Again, an exploration of multi-class entropy based shapelet extraction was out of the time scope of this thesis and is left for future work.

The brute force extraction algorithm does include performance some performance improvements involving early abandon of distance computations but still has a scalability of $O(N^2m^2)$ where m is the average length of a shapelet and N is the number of time series in the dataset. Fortunately shapelet extraction needs to be performed only once and then these shapelets can be applied any number of times. Computing the minimum distance to a test time series takes only $O(m^2)$ and in practice m is small. Shapelets are suitable in terms of scalability for the VAST classification framework.

5.3 Preliminary - Shapelet extraction results

The set of shapelets giving the best information gain for their source class is shown in Figure 5.2. Each shapelet is shown in the context of the time series it was extracted from.

The extracted shapelets are puzzling because the most obviously distinctive structures of the light curves such as the sharp spikes of the XRB or the decay of the SNe are not chosen. For several classes very short shapelets are selected and in the case of the SNe a sequence that looks like background noise is chosen.

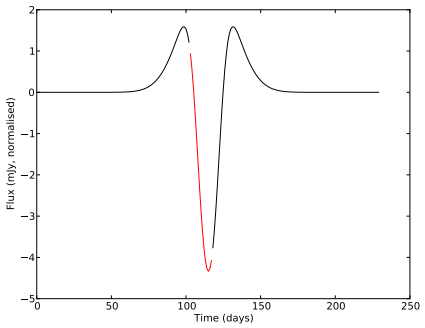
To investigate the odd results I produced a plot of the separation lines (Figure 5.1) of one crossfold of shapelets to the dataset they were extracted from. The results are in Figures 5.4 and 5.5.

The mass for the dataset elements whose class matches the shapelet's class are shown in green, for other classes, red. The algorithm's purpose is to choose precisely the subsequences that produce the best possible separation of the green mass from the red masses. Note also that a good separation here leads directly to good classification performance since these plots represent the values of the features and a supervised classifier will give good classification performance when the feature values are distinct for

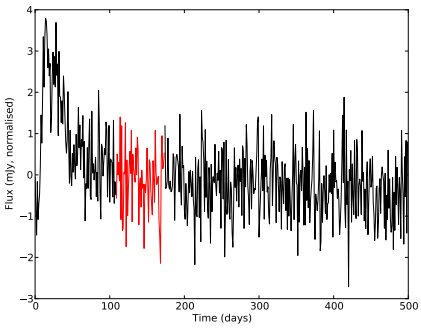
each class.

The figures demonstrate that the shapelet algorithm has found discriminative shapelets for all but the SNe and XRB classes. The ESE, FSdMe and FSRSCVn classes all have very clear separations. The BG and IDV classes are close but still clear. The Novae, SNe and XRB classes all have some degree of overlap. This is surprising since the SNe and XRB classes in particular have very distinctive structures. It is critical to my investigation that this poor separation is explained.

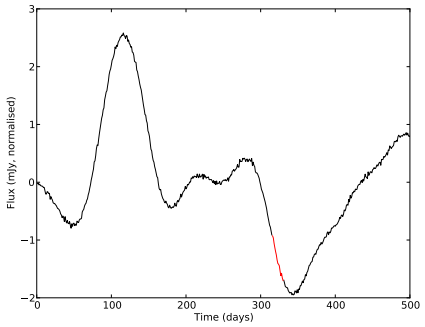
I selected regions of the SNe that intuition suggests should have been better utilised by the shapelet extraction algorithm. From these regions I drew shapelets that had the highest information gain and plotted their separation lines on the shapelet evaluation set. The results as well as figures of the structures I chose are in Figure 5.3 and show that there is too much similarity between the most intuitive shapelet choices for the SNe and XRB classes for the binary entropy based extraction algorithm to select them. This demonstrates a limitation of binary entropy — when two classes have very similar discriminative shapelets they will never be chosen. The shapelets could not classify either class, but could allow a classifier to separate them effectively from the rest of the dataset. This could improve classification performance if added to a featureset that could clearly separate the SNe and XRB classes alone, but not from the entire dataset. These results also explain the difficulty in extracting shapelets for the XRB class.



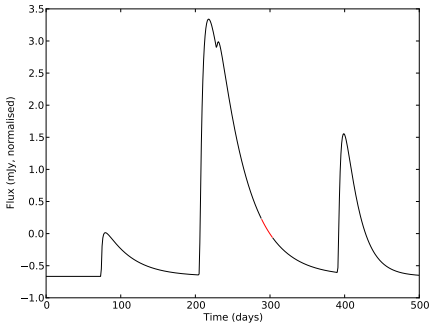
(a) ESE shapelet



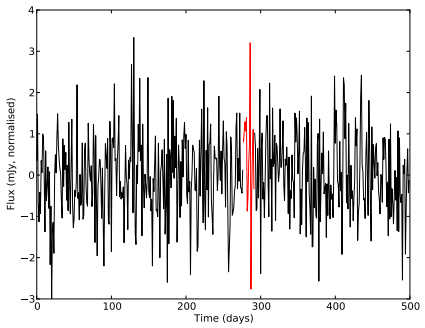
(b) SNe shapelet



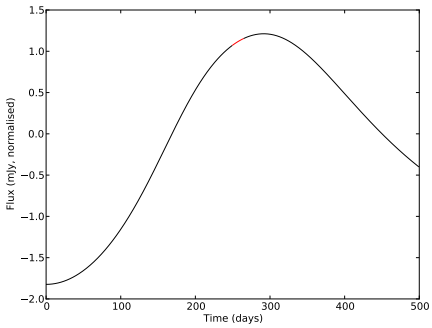
(c) IDV shapelet



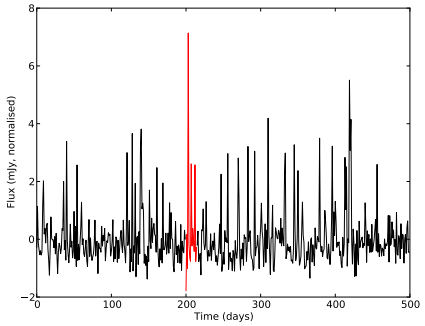
(d) XRB shapelet



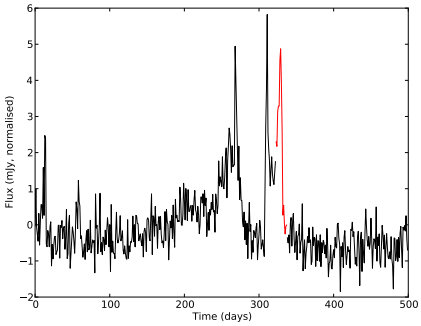
(e) BG shapelet



(f) Novae shapelet

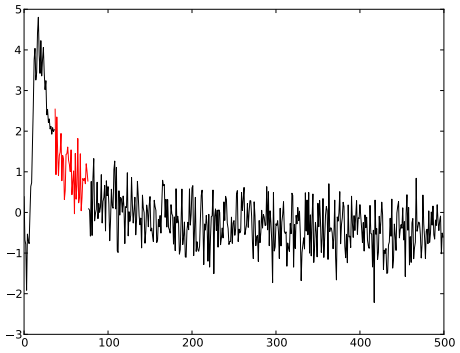


(g) FSdMe shapelet

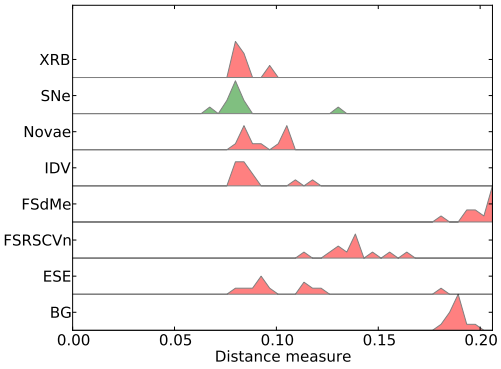


(h) FSRSCVn shapelet

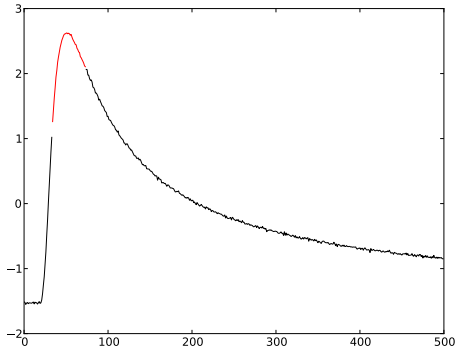
FIGURE 5.2: Single best shapelets per class extracted by the shapelet algorithm in their extraction context. The shapelet is highlighted in red.



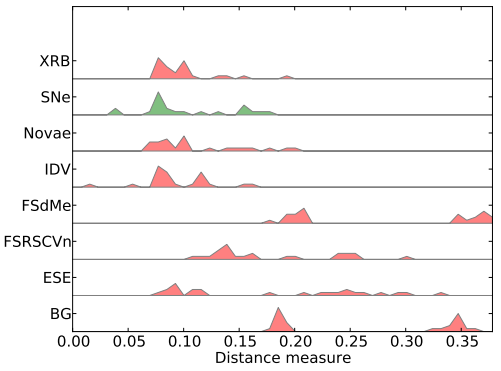
(a) SNe decay shapelet



(b) Decay shapelet splitline



(c) SNe peak shapelet



(d) Peak shapelet splitline

FIGURE 5.3: Figure illustrating why the shapelet algorithm fails to choose more variable structures for both the SNe and XRB classes. The figures on the left show the best shapelets in terms of separation for the peak and decay regions of the SNe class. Even the best shapelets extracted from these regions have large collisions with the XRB, IDV and Novae classes as shown in the right column. The green masses overlap in both cases with the XRB, Novae and IDV classes.

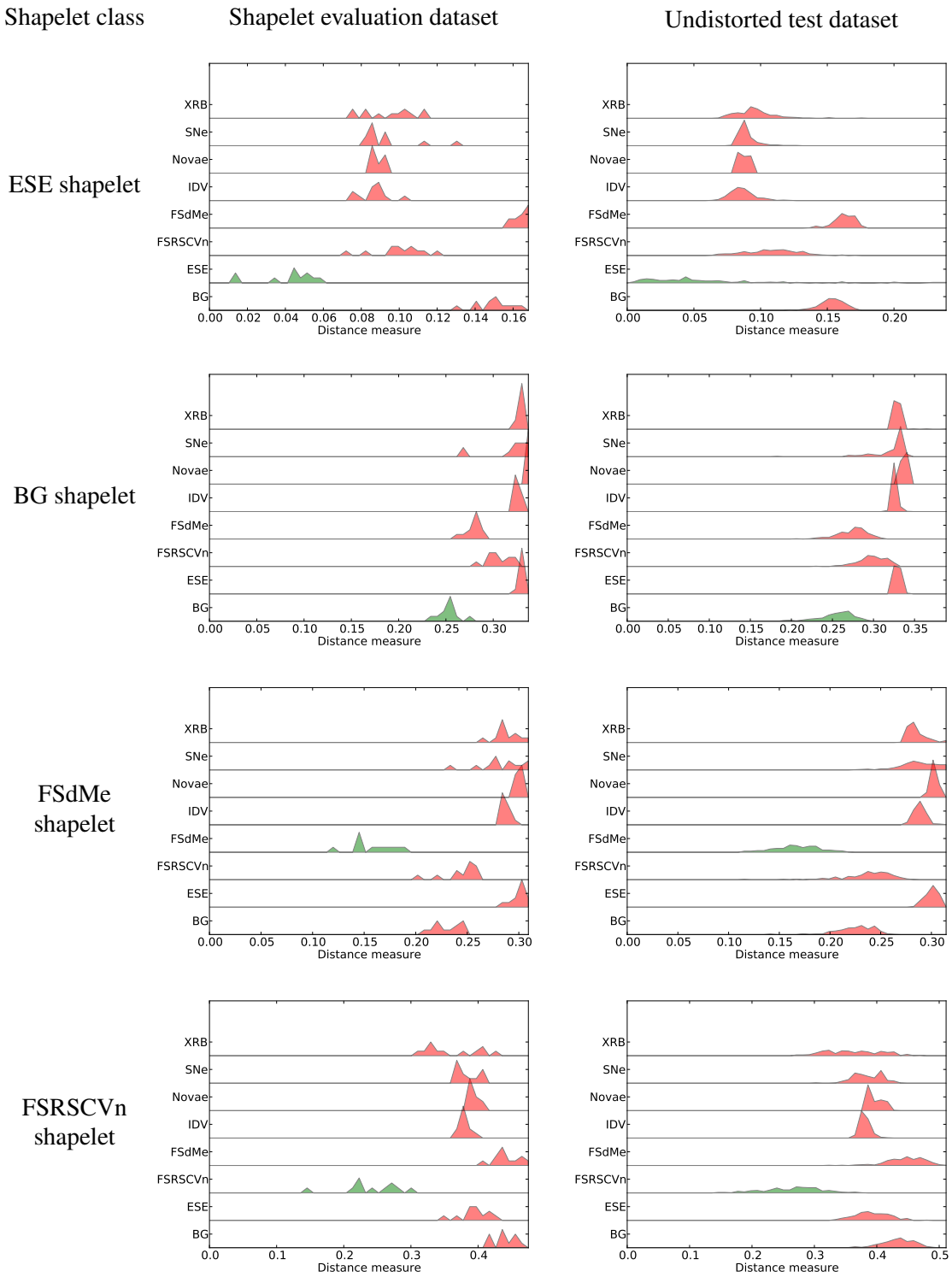


FIGURE 5.4: First set of separation lines for the sample and evaluation shapelet sets. The labels on the y-axis of each plot indicate the distribution of subsequence distances for the shapelet extracted from the class indicated in the left column. The left column of figures shows the separation lines on the dataset they were extracted from. The right column shows the separation lines on the full dataset, a superset of the extraction set. The shapelet algorithm has worked effectively if the mass of distances for the light curves for the class matching a shapelet is distinct from the other masses in a figure along the x-axis. The masses are much distinct in the left column than in the right column indicating that the shapelet algorithm is working but is not generalising that well.

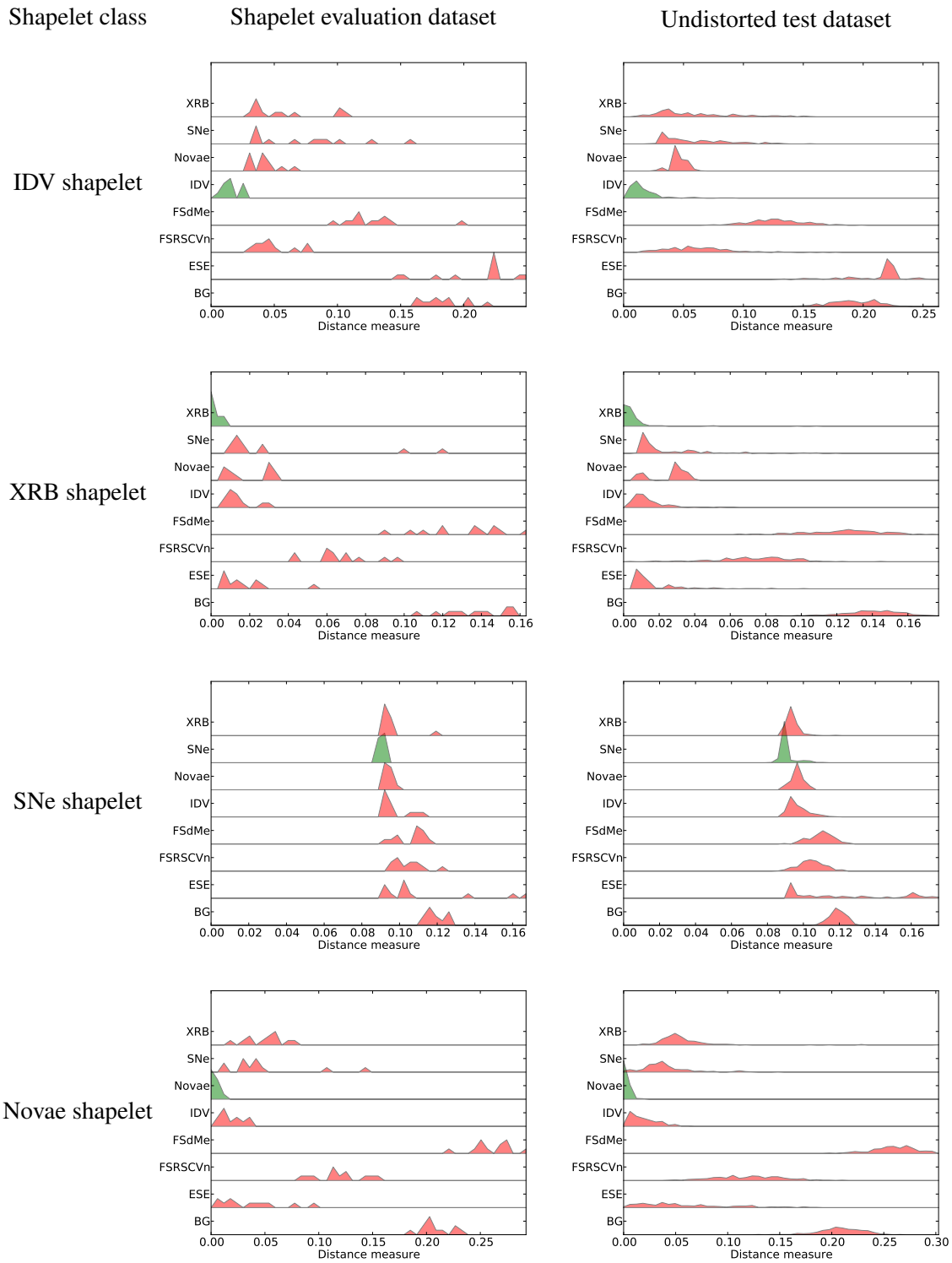


FIGURE 5.5: Second set of separation lines for the sample and evaluation shapelet sets. The labels on the y-axis of each plot indicate the distribution of subsequence distances for the shapelet extracted from the class indicated in the left column. The left column of figures shows the separation lines on the dataset they were extracted from. The right column shows the separation lines on the full dataset, a superset of the extraction set. The shapelet algorithm has worked effectively if the mass of distances for the light curves for the class matching a shapelet is distinct from the other masses in a figure along the x-axis. The masses are much distinct in the left column than in the right column indicating that the shapelet algorithm is working but is not generalising that well.

5.4 Experiment 1 - Undistorted data

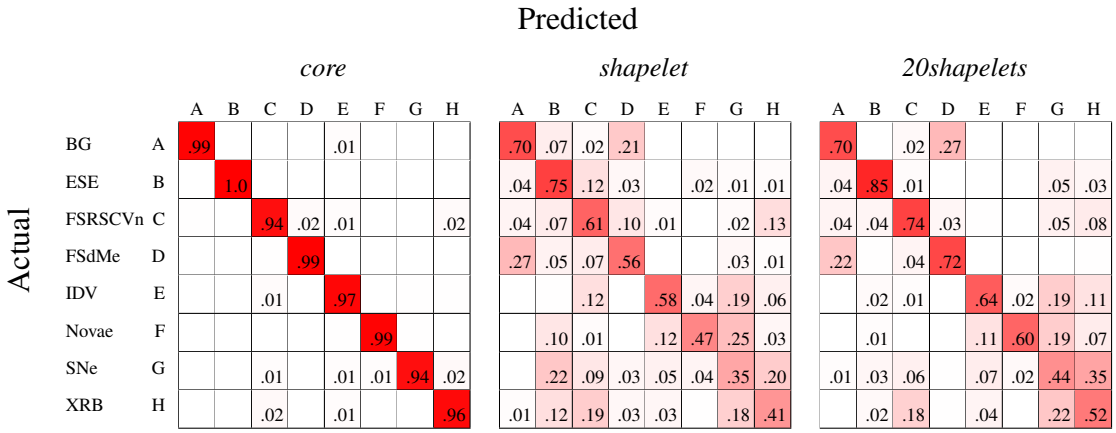


FIGURE 5.6: Confusion matrices showing classification of classes with undistorted training and test data. The matrices show significant misclassification rates (greater than 50%) for the XRB, SNe and Novae classes with the *shapelet* feature set. The *20shapelets* feature set gives marginal improvements in correct classification for all classes except BG which remains the same.

Feature set	F-Score	std(F-Score)
core	0.97	0.010
shapelet	0.56	0.17
20shapelets	0.63	0.093

TABLE 5.1: F-Score and F-Score standard deviation for classifying with shapelets and undistorted training and test data.

The performance of both sets of shapelets on the undistorted set of lightcurves is shown in the confusion matrix set in Figure 5.6 and Table 5.1 with an F-Score of for the *shapelet* feature set and 0.63 for the *20shapelets* feature set. We saw in the previous section that there are issues with the shapelet extraction algorithm that are contributing to classification errors. The poor separation for the XRB and SNe classes should not amount to an F-Score of 0.58 however, and further investigation is needed. The most likely explanation for the poor performance is that the shapelet extraction algorithm, trained on a subset of the training data, fails to choose general enough shapelets to accommodate slight variations in the testing data - a kind of overfitting. The extent to which this explains the classification performance can be demonstrated by comparing the separation lines for the shapelet evaluation and the undistorted test sets. If the separation becomes worse as we move from evaluation to testing, then the above statement is verified as a source of misclassification.

Figure 5.4 shows the separation lines for a single crossfold of the shapelet evaluation and undistorted test sets to the *shapelet* feature set. Again, the mass of test cases with the same class as the shapelet in question is coloured green. If the green mass is clearly separated from the red masses belonging to the other classes, then this will lead directly to good classification performance for that class. If some green mass is not separated from the other classes, then classification will be poor.

The separation lines show for most classes an increased overlap from the mass distributions seen on the evaluation set. The clear separations for the two flare star and ESE classes are no longer present, replaced by a smooth overlap with the next nearest class from the evaluation separation line. To demonstrate how clearly these overlaps correspond to diminished classification performance, not how conflicting classes are linked directly to off—diagonal entries on the confusion matrices in Figure 5.6. Key examples are XRB and SNe co-confusion, BG and FSdMe co—confusion, and SNe and XRB misclassification to many classes.

The conclusion to draw from the increase in overlap is that the original shapelets chosen from the limited evaluation set do not generalise well. To improve generalisation the size of the evaluation set could be increased at a large but one-off computational cost. Additionally it might be worth exploring other ways to decide on the 'best' shapelet besides the absolute highest information gain. It is possible that the split line approach selects a shapelet because its grouping has a very minor improvement in entropy over another shapelet whose overall separation from the other classes is much more distinct.

As with all the experiments in this chapter the *20shapelets* feature set showed a marginal performance improvement over single shapelets. The confusion matrices in Figure **??** showed an increase in classification rates for every class of between 0% and 10%. This indicates that the clustering algorithm is effective at finding additional useful shapelets.

5.5 Experiment 2 - Introducing gaps into the light curve

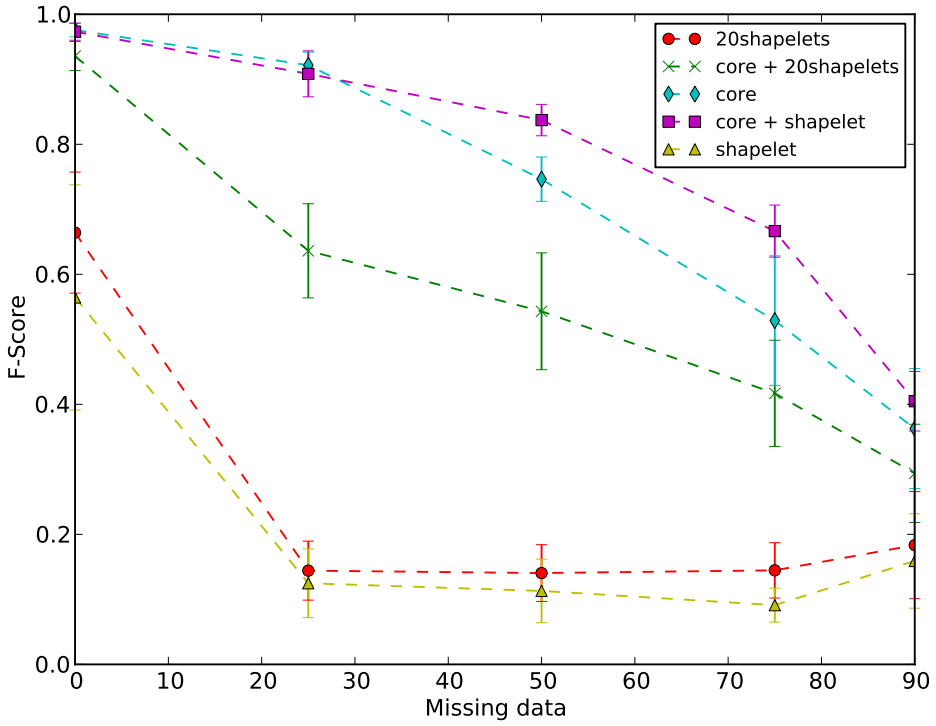


FIGURE 5.7: Plot of F-Score versus amount of missing data in signal. The *core + shapelet* feature set gives the best performance. The *20shapelet* feature set reduces classification performance. Both shapelet feature sets have F-Scores lower than 0.2 for 25% missing data and above.

There are some interesting observations to make on the results in this experiment. Figure 5.5 shows that the F-Score for classification with both the *shapelet* and *20 shapelet* drops to about 0.15 as soon as any amount of data is removed from the signal stays below 0.2 for any amount of missing data. An F-Score of around 0.125 essentially means random choice by the classifier in an 8-class classification problem. These results clearly indicate that the application of the shapelet features has failed in some way.

Paradoxically, the introduction of the *shapelet* feature set into the classifier boosts F-Score by a the *shapelet* feature set, boosting F-score by close to 0.1 at 50% missing, and close to 0.15 at 75% missing. This performance is in contrast to the *20 shapelet* set which reduces classifier performance as we would expect.

The first result, that both shapelet sets perform very poorly on missing data, has two likely contributing factors:

(1) For sufficient amounts of missing data the distinct features the shapelets use for classification are not still present
(2) The missing data allows the distance measure to ignore critical parts of shapelets in determining a match.

The first explanation certainly must become true at a point. If we had only 5 data points sampled from a different part of the light curve than our shapelets were extracted from then classification using that shapelet is not possible. However, for smaller amounts of missing data the most critical structures are still evident as can be seen in sectionřefexpframework, experimental framework chapter, and in appendix A.3, showing samples from the dataset for each parameter in this experiment. The results for the 25% missing data experiment are more likely to be explained by the second complication.

The distance measure used so far for producing features from shapelets, subsequence distance, is intended for use on fully sampled time series and did not immediately extend to the transient classification problem. As outlined in the introduction to this chapter, it was modified so a shapelet must fit at least 5 data points to prevent the algorithm matching a shapelet onto an empty region of data. These results suggest that this modification is not sufficient to make shapelets functional for any amount of missing data

Although a shapelet must contain a structure distinctive to the class from which it was extracted, it may have subsequences that are not. An illustrative example is the noise situated in the shapelet extracted for the FSdMe class (Figure 5.2), consisting of three peaks and then flat noise. If the peaks are omitted in the distance measure computation then the underlying flat noise could fit any other light curve in our dataset with a near 0 distance. This could well explain the solid mass in the FSdMe column in the confusion matrices in Figure 5.8

To verify that these false positives are the cause of the poor performance I did a manual investigation of the minimum distances found for a few shapelets to the light curves for one crossfold of the 25% missing data test set. The results are shown in Figure 5.10. As a rough way of quantifying exactly how much of the critical part of these signals goes unmatched, I include for each shapelet the fraction of the total deviation from the mean of the shapelet datapoints that go unmatched (visualized in Figure 5.10).

For contiguous time series this measure will always be 1. If no point was matched at all, the minimum distance would always be zero, and this measure would be 0. If half the deviation of the shapelet is matched, it would be 0.5. The greater this fraction then the more likely that the match indicates a false positive for the shapelet feature.

The matches in the figure show exactly why the classifier is so confused. Whenever there is a small gap in the time series the minimum distance measure chooses that region to perform a match and getting the same value for almost every piece of test data. There is no meaning as a feature for the distance of a shapelet to a test case.

To fix this problem I produced a modified distance measure that discards all matches not meeting a deviation fraction threshold. The problem with this approach is that there is no certain way to decide on a sensible value of the threshold. It should probably be closer to 1, say 0.9, since as we see in the figure above matchings can still be very poor even with 70% of the variance matched. If the value is 1 then the minimum distance would ignore matches with even a single unmatched point. What value would work best in practice would have to be determined by experiment and might vary depending on the kinds of data being classified. Figure 5.11 shows the minimum distance matches for the same time series with the threshold modification.

The actual choice of match for the IDV time series is only shifted slightly, still utilising that gap to omit the small fraction of the variance it is still allowed to (only 10% of total variance). However, the spread between the distances is at least now distinct (0.08 for the IDV and 0.04 for the FSdMe). This distinctness will at least allow the possibility of correct classification. Missing data in time series is a serious issue then for the shapelet classification approach. A modified distance has demonstrated potential to improve the situation but no time was available to run further experiments. These then are left for future work.

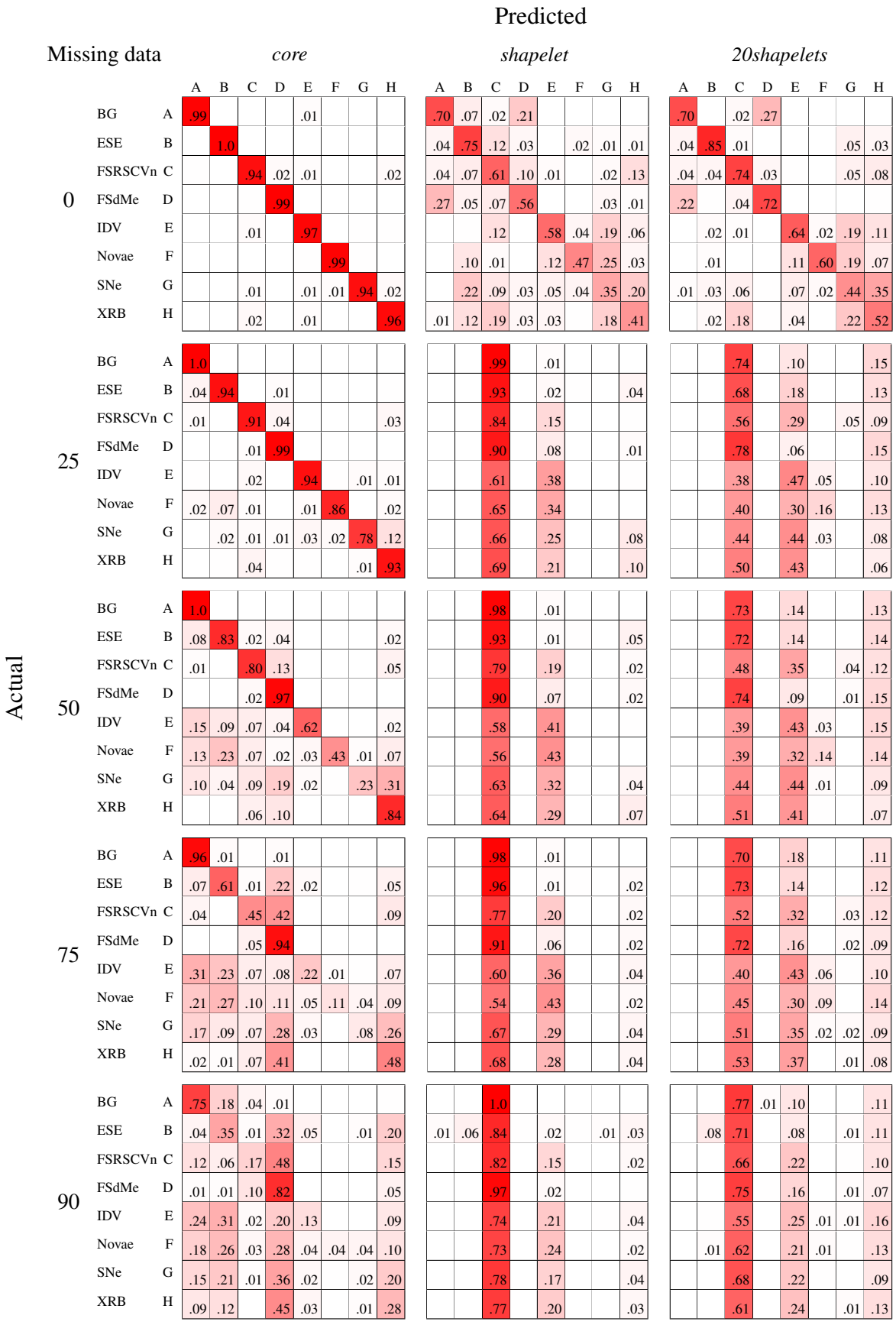


FIGURE 5.8: Confusion matrices for the missing data experiment. The confusion matrices show a very strong tend for all classes to be misclassified as FSRSCVn and IDV for all amounts of missing data

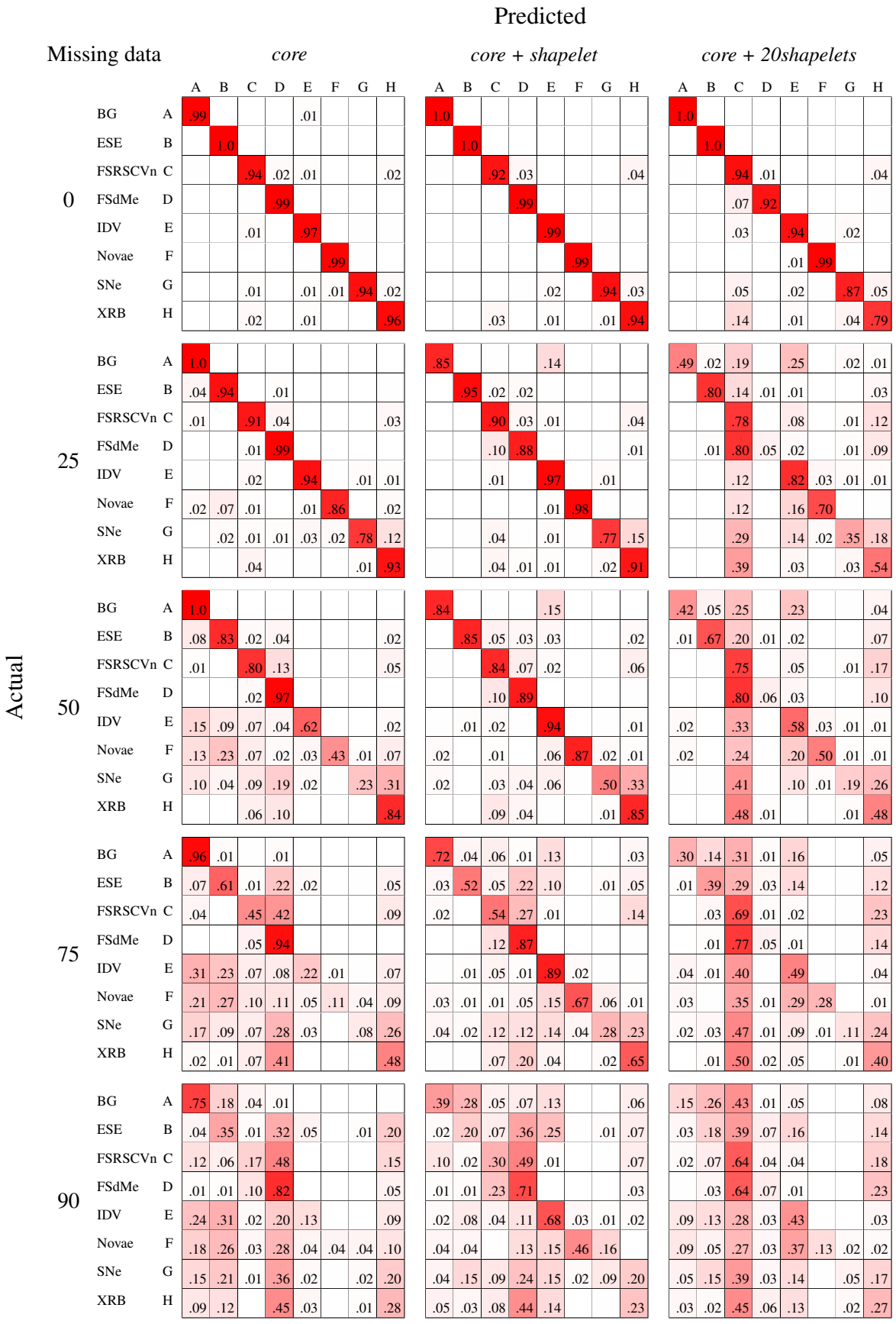
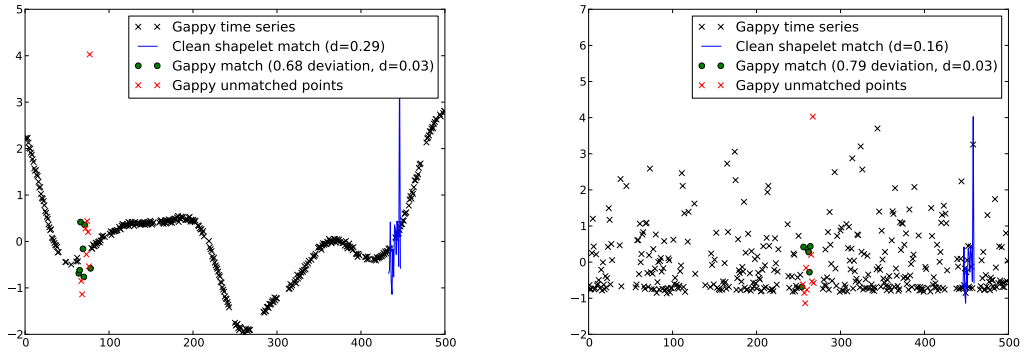
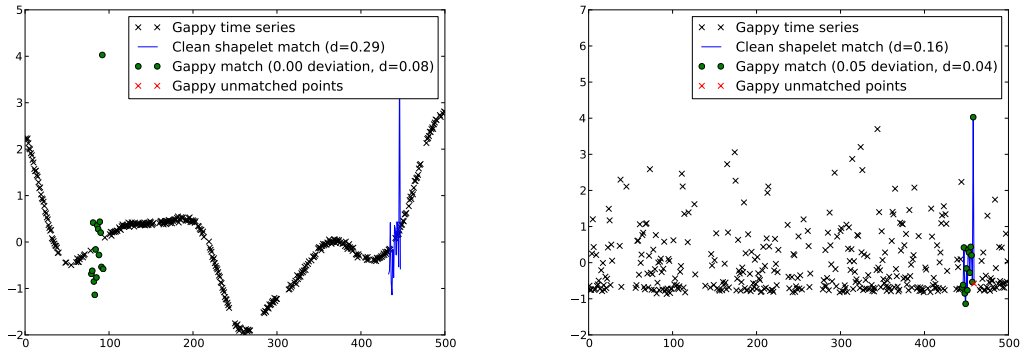


FIGURE 5.9: Confusion matrices for the missing data experiment. The confusion matrices show paradoxically that classification performance on missing data is improved with the addition of the *shapelet* set. In particular for the XRB, SNe, Novae and IDV classes



(a) Match of FSdMe shapelet to gappy IDV time series (b) Match of FSdMe shapelet to gappy FSdMe time series

FIGURE 5.10: False positives for the 25% missing data experiment for the IDV and FSdMe classes to an FSdMe shapelet. d means the value of the minimum distance, deviation is the total fraction of deviation matched



(a) Match of FSdMe shapelet to gappy IDV time series (b) Match of FSdMe shapelet to gappy FSdMe time series

FIGURE 5.11: Results of modifying the distance measure to use a deviation matched threshold. The distances are equal with no threshold, and are somewhat distinct otherwise, although not as clearly separated as on undistorted data.

5.6 Experiment 3 - Limiting the amount of the light curve observed

This experiment involves limiting the percentage of light curve observed and observing how classification performance in terms of F-Score changes as the visible part of the light curve is reduced. The features used are the single shapelet feature sets, both alone and combined with the *core* feature set.

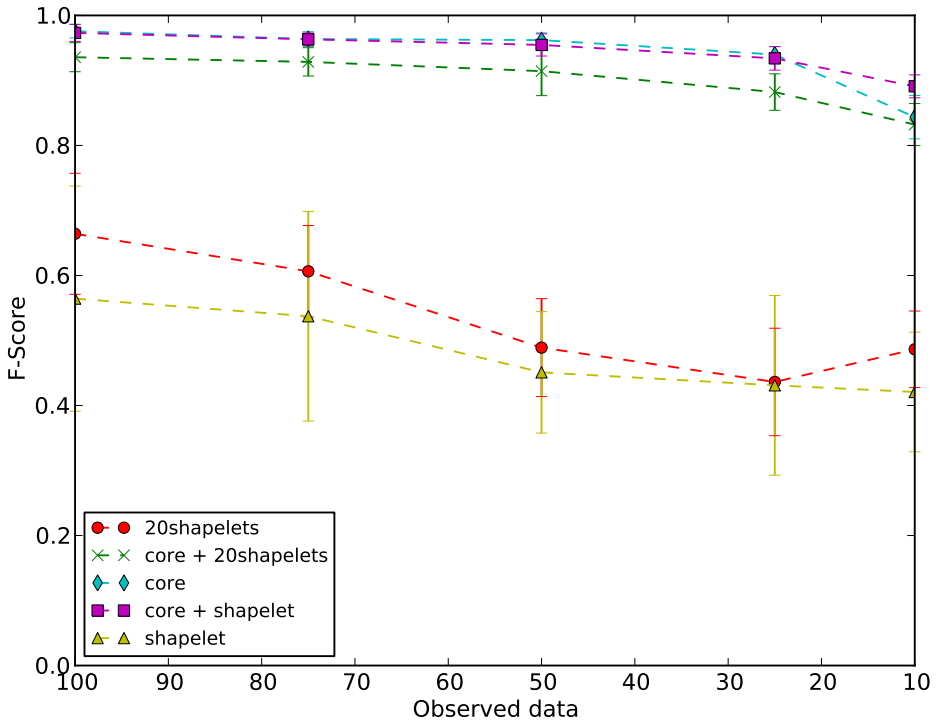


FIGURE 5.12: Plot of F-Score versus percentage of light curve observed. There is a marginal increase in classification perofrmance at 10% observed data for the *core* + *shapelet* feature set.

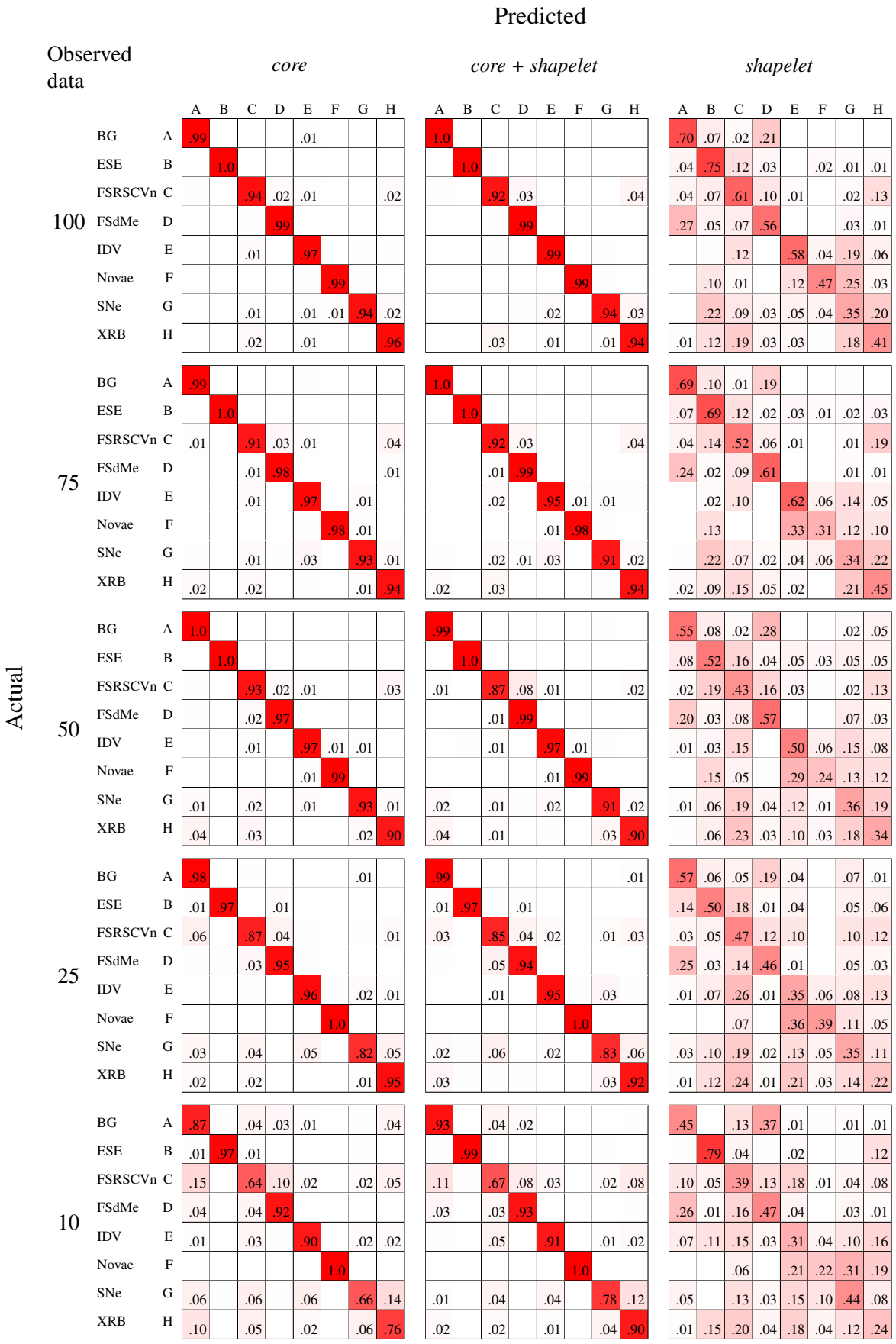


FIGURE 5.13: Confusion matrices for the observed data experiment. The confusion matrices show that at 10% observed data that classification is improved by superior discrimination of the SNe and XRB classes from the BG class.

The F-Scores in Figure 5.12 show that classification performance for the two shapelet feature sets decreases gradually as the percentage of the light curve observed is decreased. At 10% observed data there is a marginal improvement in F-Score when using the combined *core* + *shapelet* feature set of 0.07 F-Score. Referring to the confusion matrix in Figure 5.13 we see that this is the result of the shapelets allowing a more accurate identification of the background noise class. The difference of F-Score is too marginal to suggest that the shapelet algorithm is useful for early classification.

5.7 Experiment 4 - Introducing noise into the light curve

The aim of this experiment was to assess the impact of noise on the shapelet classification algorithm.

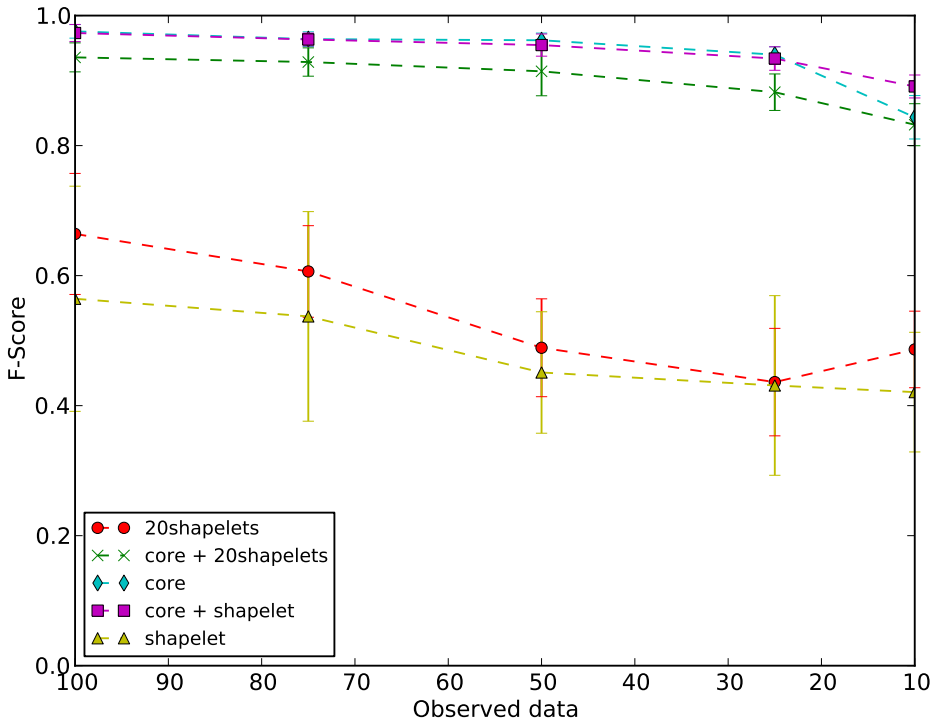


FIGURE 5.14: Plot of F-Score versus amount of noise introduced into the signal. The shapelet sets perform poorly for any amount of noise and also decrease classification performance when combined with the *core* feature set.

Classification becomes poor very quickly on all shapelet sets. At a 1.5 noise to signal variance ratio the *shapelet* feature set produces near random classification. With very strong signals (at 0.5 and 1.0 noise to signal), classification is at 0.3 and 0.2 respectively. Classification accuracy for the Nova and IDV drops to 0 for a 0.5 signal to noise ratio and below 0.2 for both the XRB and SNe. The most likely explanation for this is that the short shapelets chosen by the shapelet extraction algorithm can fit essentially anywhere on noisy data. A suggestion for improving the shapelet classification algorithm on noisy data is to use a separate evaluation and sample set for the shapelets. If slightly noisy data were used to evaluate the clean, undistorted data, then short shapelets would never be chosen unless they had very clear structures that stood out from noise.

So introducing noise is yet another difficulty for the shapelet algorithm and the transient classification problem, so much so that the number of shapelets used corresponds with a large decrease in classification performance when combined with the *core* feature set. If it is actually possible to use shapelets to classify transients when they have some or a lot of noise in their signal will be demonstrated by modifying the evaluation set and is left for future work.

5.8 Conclusion

This chapter explored the shapelet feature representation algorithm for time series and its ability to improve classification performance on distorted transients. The preliminary results when classifying undistorted data showed that the shapelets can classify the light curves with an F-Score of 0.58, and when using clustering an F-Score of 0.63. The clustering algorithm provided marginal performance improvements in all situations. These results meant that the shapelet features might be useful for improving classification performance when combined with the *core* feature set used in 4. However, as soon as distortions were introduced the shapelet algorithm gave very poor F-Scores, lower than 0.3 for any amount of noise or missing data. The cause of the problems with missing data was identified as false positives due to the way that subsequence distance matched missing data and a modification that forces subsequence distance was proposed. Noise was an issue for shapelets because they were originally extracted and evaluated on undistorted data. As soon as noise was introduced the short shapelets chosen from the clean data no longer had discriminative properties. I proposed using noisy data to evaluate the discriminative power of the shapelets during extraction to prevent the algorithm from choosing short shapelets.

CHAPTER 6

Conclusion

The next generation of radio telescopes such as the ASKAP telescope array will produce large volumes of data in the form of time series that represent the behaviour of stellar objects. Identifying which of these time series contain transient behaviour — a rapid change over time — is of great interest to scientists because these typically indicate the unfolding of extreme processes out in space. The challenge is identifying these transient phenomena and classifying them as early as possible to allow astronomers around the world to investigate them with other instruments.

In this thesis I contribute to solving the transient classification problem by proposing, implementing and evaluating a feature based supervised classifier for astronomical transients. The evaluation involved characterising how the various distortions present in astronomical data impact on classification accuracy and proposing improvements to the approach. There are many components involved in this contribution that are in themselves significant:

(1) I give a thorough review of existing time series literature, identifying feature representation as a good candidate for effective classification of transients.
(2) I propose an experimental framework to evaluate classifiers by stating transient classification as a multi-class classification problem with simulated transients.
(3) I propose and implementing consistent ways of simulating the distortions present in astronomical data that can be re-used by collaborators doing similar research.
(4) Using the experimental framework I evaluate Random Forest supervised classifier, implementing wavelet transforms and statistical properties of the light curves as features.
(5) I use the results of the feature based classification evaluation to characterise the effect of distortions on classification.

(6) As an extension to the core features I give a preliminary investigation into the Shapelet time series feature representation, identifying the limitations of the approach and proposing improvements.

I tie the analysis of these experiments back to the VAST classification pipeline and conclude the feature based classified is a viable option. I suggest directions for future work as introducing the preprocessing of distorted test data before feature extraction and improving the application of the Shapelet feature representation to distorted light curves.

In the introduction I introduced astronomical data as time series which are sequences of time indexed data points. I defined a transient as a temporary, possibly repeating phenomena appearing in a time series that can be identified by its characteristic shape or spectral properties. I also give definitions of the distortions present in astronomical data that can make an underlying transient signal difficult to classify. The distortions included noise in the signal due to interference of the light with the interstellar medium, missing datapoints as the result of a particular sampling routine or shared telescope responsibilities, and scaling of the intensity of the transient light curve according to a power law distribution to simulate the distribution of the sources of transient signals in space. I also identify some of the complications of transient classification in the context of the VAST project, including the requirement that any classification algorithm be highly scalable and identify transients as early as possible. Each of the distortions along with early classification is directly assessed by the classification experiment in Chapter 4.

My literature review gives a thorough exploration and evaluation of existing time series classification literature across multiple application domains. It identifies supervised classification and feature extraction including Haar wavelets, Lomb-Scargle periodograms, and statistical properties of the flux and gradient distributions as a good choice for classifying transients in time series. The literature review also explored and compared a variety of other classification approaches including distance measures like Dynamic Time Warping, Temporal Grammars, Gaussian Processes and Shapelets as classification approaches. The review shows that distance measures are probably not effective for coping with distortions but having a background understanding of them is useful for understanding other classification approaches such as the subsequence distance measure used in the Shapelet feature representation algorithm. I give an introduction to the application of motif-finding algorithms and the Shapelet extraction algorithm for time series and illustrates that it is a promising method to make a feature based classifier robust to unknown start and end points of a transient event. Additionally I discuss Temporal Grammars, an interesting classification approach because they are naturally invariant to amplitude scaling, but I

conclude that z-normalisation to approximately remove amplitude scaling then the application of other approaches such as statistical models is likely to be more accurate. Finally I review Gaussian Processes, a classification approach with a natural robustness to both noise and missing data, and conclude that they are unsuitable for the problem due to their high time complexity and lack of flexibility in classifying time series without defined start and end points.

In Chapter 3, I implement an experimental evaluation scheme of astronomical transient classification to assess the impact of distortions on classification. It involves limiting the scope of the problem to an multi-class classification problem with 8 simulated transient classes: 7 kinds of astronomical phenomena with periodic, repeated and large-scale characteristic structures, and one class representing background noise. The simulated transient models were provided by Kitty Lo (VAST Memo in prep) representing real world transients including Extreme Scattering Events (ESEs), Intra-Day variables (IDVs), Two kinds of flare stars (FSdMe and FSRSCVn), X-ray binaries (XRBs) and Supernovae and Novae (SNe, Novae). The background noise (BG) class was simulated using Gaussian noise. The process of introducing the distortions for early classification (cropping), noise, missing data and a power law distribution was outlined and I implemented software to apply the distortions to the raw light curves. The framework made the assumption that the start point of a transient event was known and uses a sliding window of 500 datapoints for each light curve. The framework used 200 of each transient class giving a total dataset of 1600 light curves for reliable classification results. For evaluation it proposed 10-fold cross validation on the dataset with F-Score as a measure of classification performance and the standard deviation of F-Scores used to evaluate the reliability and significance of a classification result. This framework represents a significant contribution to exploring the astronomical transient classification problem since any classifier can be inserted into the framework and evaluated.

In Chapter 4 I apply a Random Forest supervised classifier to the experimental framework and implement the features identified in the literature review as likely to be effective in classification. These features included the frequencies corresponding to the strongest peaks of a Lomb-Scargle periodogram, The coefficients of a Haar wavelet transform, statistical properties of the flux distribution such as its kurtosis, skew, and distributions of flux relative to the mean and standard deviation, and those same statistical properties extracted from a gradient distribution produced by a linear segmentation. The key results obtained from the experiments are as follows:

- The classifier performs well on undistorted data with an F-Score of 0.97 indicating very high precision and recall. Separating the transient structures is not a problem for the classifier.

- Neither is the early classification of undistorted light curves, with an F-Score above 0.9 for up to and including 20% of the signal being observed, and 0.8 at 10%.
- When using training and test sets with equal amounts of data missing the classifier also performs well, staying above 0.9 F-Score up to 75% missing data. At 90% missing data the F-Score falls to 0.8. This means that our features can cope with classifying transient light curves with large amounts of missing data
- When using undistorted training data and distorted test data the differences in feature values seriously affects classification performance. The *spectral* feature set is the most sensitive and removing it in the subtractive analysis increased F-Score by 0.2 at 50% and 75% missing data. The *statistical* feature set was the most robust, and its removal caused F-Score to drop by 0.1 F-Score for 25% to 75% missing data
- In dealing with noise F-Score falls linearly as classification performance is increased. For small amounts of noise the *statistical* feature set is most important, improving F-Score by 0.1 up to 1.5 noise to signal variance. From and after that amount of noise however, the *haar* wavelet feature set is most important, also improving F-Score by 0.1 on the combined feature set. The experiment demonstrated that the Haar wavelet features are best at identifying the SNe and Novae classes
- When using undistorted training data classification performance falls to 0.57 F-Score at 1.0 for the best performing feature set, and 0.5 at 1.5. Those results clearly show that the *statistical* features, most robust to missing data, are the most sensitive to noise in terms of the shift in feature values. The confusion matrices demonstrate that these results arise largely from misclassification of all classes as the background noise class.
- The combined distortion results when using equally distorted training and test data show that the *haar* and *statistical* features are the best for dealing with combined distortions. Their exclusion in subtractive analysis causes a drop of at least 0.1 F-Score for all amounts of observed data. The F-Scores show a linear decrease from 0.8 at 100% observed data to 0.4 at 10% observed data. This means that are features are not adequate to classify the distorted light curves for the accuracy required in the VAST project.
- When using undistorted training and distorted test data with combined distortions the classifier does not perform better than 0.4 F-Score for any amount of observed data. Since in reality the amount of noise and missing data in the light curve will not be known in advance then this result is what we would expect if the classifier were placed in the VAST pipeline.

- Preprocessing to address the shift in feature values when using unequally distorted training and test sets would make this classifier more viable.
- Exploring new features to improve classification performance with equally distorted training and test sets is important as well. The upper bounds placed in the combined distortions experiment with equally distorted training and test data are still too low to be useful in the VAST pipeline.

In Chapter 5 I give a preliminary investigation into the shapelet time series feature representation. The key results of these experiments are as follows:

- I showed that for undistorted data the shapelet algorithm without modification achieves an F-Score of 0.58.
- I demonstrated that the basic algorithm as proposed by Ye in Ye and Keogh (2009) has some fundamental problems in coping with noise and missing data and propose improvements to the algorithm that will improve classification performance.
- I identified that the use of undistorted training data for extracting and evaluating shapelets leads to choices of subsequences that are not useful for dealing with noise, and proposed evaluating shapelet discrimination on distorted datasets to make the extraction algorithm choose more robust shapelets.
- I identify that the shapelet algorithm cannot discriminate between the XRB and SNe classes because they have similar distinctive substructures and suggest using multi-class entropy to extract shapelets that are useful for classifying multiple transient classes at once.
- I Implemented a shapelet clustering algorithm that showed a marginal (0 to 0.1 F-Score) performance improvement on all kinds of distortions.
- I Showed that the shapelet algorithm gives a marginal 0.1 F-Score improvement when combined with the *core* feature set for 10% observed data by helping the classifier identify discriminate the background noise class.

These experiments led me to conclude that the shapelet algorithm shows some promise for classifying transients with an F-Score of 0.58 on undistorted data. With future work to improve the way they are applied they should improve classification performance of distorted light curves.

Finally in the discussion I tie the results back to the VAST project and make recommendations for improvements and future work. I conclude that the results are not yet acceptable for use in the VAST

classification pipeline with F-Scores no higher than 0.4 for fully observed transients. However, I note that the robustness of the features demonstrated with F-Scores of 0.9 on up to 90% missing data and 0.8 at 1.0 noise-signal variance when using equally distorted training and test data means that this approach holds promise. I assert that the addition of preprocessing techniques involving regression and noise filtering or smoothing would improve classification performance for the missing data and noise distortions when combined with the *core* feature set.

The problem of transient classification is as yet unexplored in astronomical or time series literature. This thesis characterises the problems in terms of the choice of supervised classification and the types and severity of distortions appearing in test data. It demonstrates that feature extraction and supervised classification is a viable technique for transient classification. Future improvements will make this approach suitable implementation in a classification pipeline capable of dealing with the large volumes of data that will be produced by next generation telescopes.

Bibliography

C. Bahlmann, B. Haasdonk, and H. Burkhardt. 2002. Online handwriting recognition with support vector machines-a kernel approach. In *Frontiers in Handwriting Recognition, 2002. Proceedings. Eighth International Workshop on*, pages 49–54. IEEE.

G.E. Batista, X. Wang, and E.J. Keogh. 2011. A Complexity-Invariant Distance Measure for Time Series.

D. Berndt and J. Clifford. 1994. Using dynamic time warping to find patterns in time series. In *AAAI-94 workshop on knowledge discovery in databases*, pages 229–248.

L. Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

P. Capitani and P. Ciaccia. 2007. Warping the time on data streams. *Data & Knowledge Engineering*, 62(3):438–458.

L. Chen and M.T. Özsu. 2005. Using multi-scale histograms to answer pattern existence and shape match queries. In *In SSDBM*. Citeseer.

M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten. 2009. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

M.W. Kadous and C. Sammut. 2005. Classification of multivariate time series and structured data using constructive induction. *Machine learning*, 58(2):179–216.

E. Keogh, S. Chu, D. Hart, and M. Pazzani. 2001. An online algorithm for segmenting time series. In *icdm*, page 289. Published by the IEEE Computer Society.

E. Keogh and M. Pazzani. 1998. An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. In *Proceedings of the 4th International Conference of Knowledge Discovery and Data Mining*, pages 239–241. AAAI Press.

M Lázaro-Gredilla, J Quiñonero-Candela, CE Rasmussen, and AR Figueiras-Vidal. 2010. Sparse spectrum gaussian process regression. *Journal of Machine Learning Research*, 11:1865–1881.

J. Lin, E. Keogh, S. Lonardi, and P. Patel. 2002. Finding motifs in time series.

Woong-Kee Loh, Yang-Sae Moon, and Jaideep Srivastava. 2010. Distortion-free predictive streaming time-series matching. *Information Sciences*, 180(8):1458 – 1476.

NR Lomb. 1976. Least-squares frequency analysis of unequally spaced data. *Astrophysics and space science*, 39(2):447–462.

JEJ Lovell, BJ Rickett, J.P. Macquart, DL Jauncey, HE Bignall, L. Kedziora-Chudczer, R. Ojha, T. Pursimo, M. Dutka, C. Senkbeil, et al. 2008. The micro-arcsecond scintillation-induced variability (masiv) survey. ii. the first four epochs. *The Astrophysical Journal*, 689:108.

A. Mueen, E. Keogh, and N. Young. 2011. Logical-shapelets: An expressive primitive for time series classification.

R.T. Olszewski. 2001. *Generalized feature extraction for structural pattern recognition in time-series data*. Ph.D. thesis, Citeseer.

M. Osborne and S.J. Roberts. 2007. Gaussian processes for prediction. Technical report, Technical Report PARG-07-01. Available at www. robots. ox. ac. uk/parg/publications. html, University of Oxford.

I. Popivanov and R.J. Miller. 2002. Similarity search over time-series data using wavelets. In *icde*, page 0212. Published by the IEEE Computer Society.

W.H. Press and G.B. Rybicki. 1989. Fast algorithm for spectral analysis of unevenly sampled data. *The astrophysical journal*, 338:277–280.

A. Ranganathan, M.-H. Yang, and J. Ho. 2011. Online Sparse Gaussian Process Regression and Its Applications. *IEEE Transactions on Image Processing*, 20:391–404.

Carl E. Rasmussen and Christopher Williams. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

C.E. Rasmussen. 1996. *Evaluation of Gaussian processes and other methods for non-linear regression*. Ph.D. thesis, Citeseer.

J.W. Richards, D.L. Starr, N.R. Butler, J.S. Bloom, J.M. Brewer, A. Crellin-Quick, J. Higgins, R. Kennedy, and M. Rischard. 2011. On machine-learned classification of variable stars with sparse and noisy time-series data. *The Astrophysical Journal*, 733:10.

H. Sakoe and S. Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 26(1):43–49.

J.D. Scargle. 1982. Studies in astronomical time series analysis. II-Statistical aspects of spectral analysis of unevenly spaced data. *The Astrophysical Journal*, 263:835–853.

H. Shimodaira, K. Noma, M. Nakai, and S. Sagayama. 2002. Dynamic time-alignment kernel in support vector machine. *Advances in neural information processing systems*, 2:921–928.

E. Snelson and Z. Ghahramani. 2005. Sparse Gaussian Processes using Pseudo-inputs. *Neural Information Processing Systems 18*.

M. Vlachos, D. Gunopoulos, and G. Kollios. 2002. Discovering similar multidimensional trajectories. page 673. Published by the IEEE Computer Society.

G. Wachman, R. Khardon, P. Protopapas, and C. Alcock. 2009. Kernels for Periodic Time Series Arising in Astronomy. *Machine Learning and Knowledge Discovery in Databases*, pages 489–505.

C. Walder, K.I. Kim, and B. Sch
"olkopf. 2008. Sparse multiscale Gaussian process regression. In *Proceedings of the 25th international conference on Machine learning*, pages 1112–1119. ACM.

M. Walker and M. Wardle. 1998. Extreme scattering events and galactic dark matter. *The Astrophysical Journal Letters*, 498:L125.

Z. Xing, J. Pei, P.S. Yu, and K. Wang. 2011. Extracting Interpretable Features for Early Classification on Time Series. In *SIAM International Conference on Data Mining (SDM)*.

L. Ye and E. Keogh. 2009. Time series shapelets: a new primitive for data mining. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 947–956. ACM.

Appendices

APPENDIX A

Samples of distorted light curves from experiment test sets

A.1 Limiting the length of the observed lightcurve

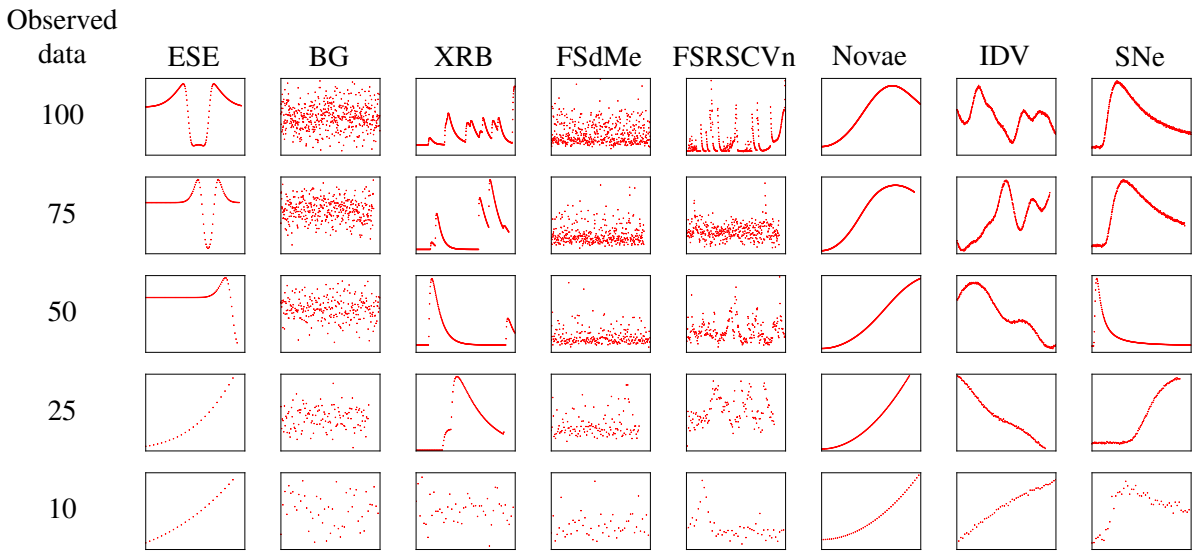


FIGURE A.1: Light curve samples with Gaussian noise introduced as the fraction on the y axis multiplied by its standard deviation.

A.2 Introducing noise into the light curve

A.4 Simultaneous distortions and limiting the observed light curve

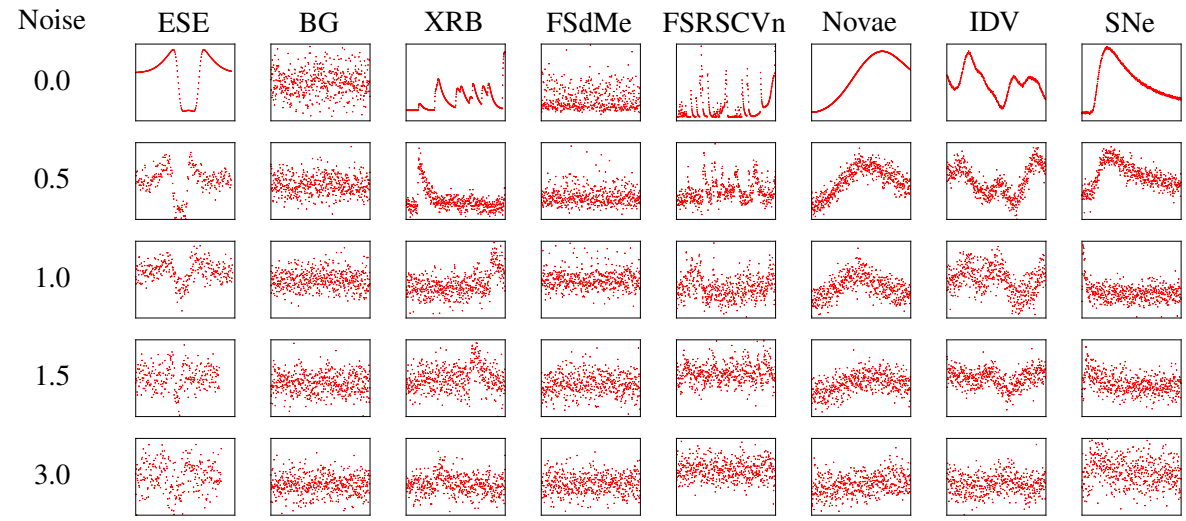


FIGURE A.2: Light curve samples with Gaussian noise introduced as the fraction on the y axis multiplied by its standard deviation.

A.3 Introducing gaps into the light curve

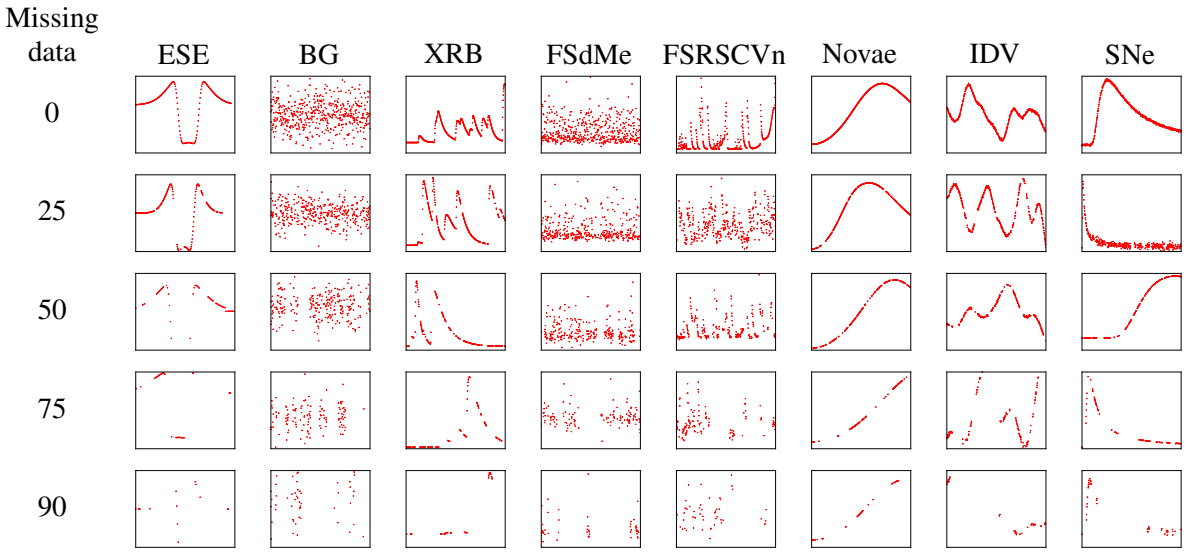


FIGURE A.3: Light curve samples from our dataset with missing data introduced as small random chunks until the percentage of the data points indicated in the left column remains

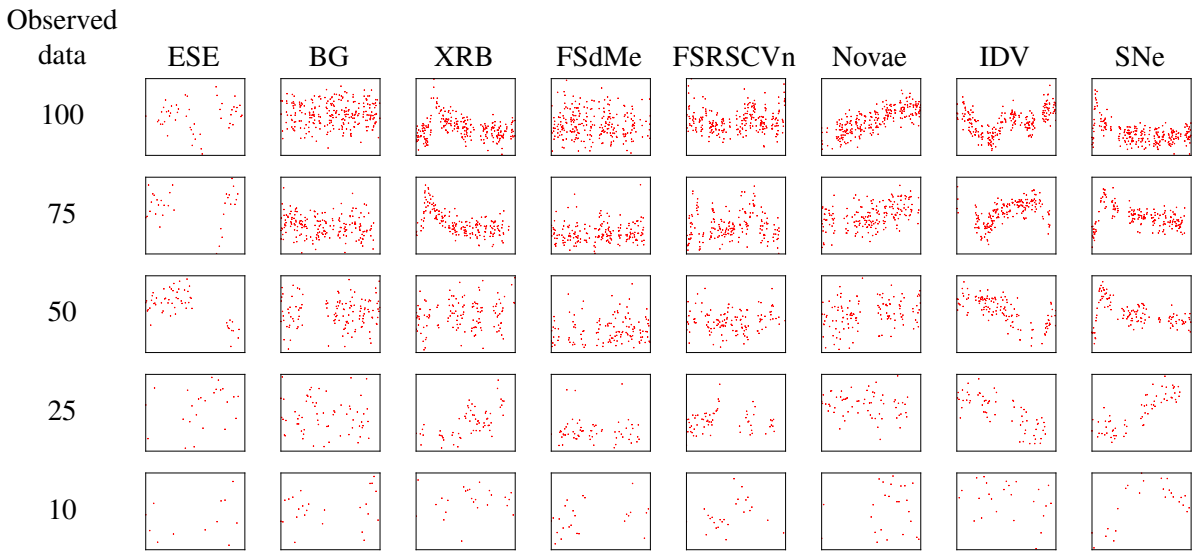


FIGURE A.4: Light curve samples from our dataset with missing data introduced as small random chunks until the percentage of the data points indicated in the left column remains

APPENDIX B

Schematic of VAST pipeline

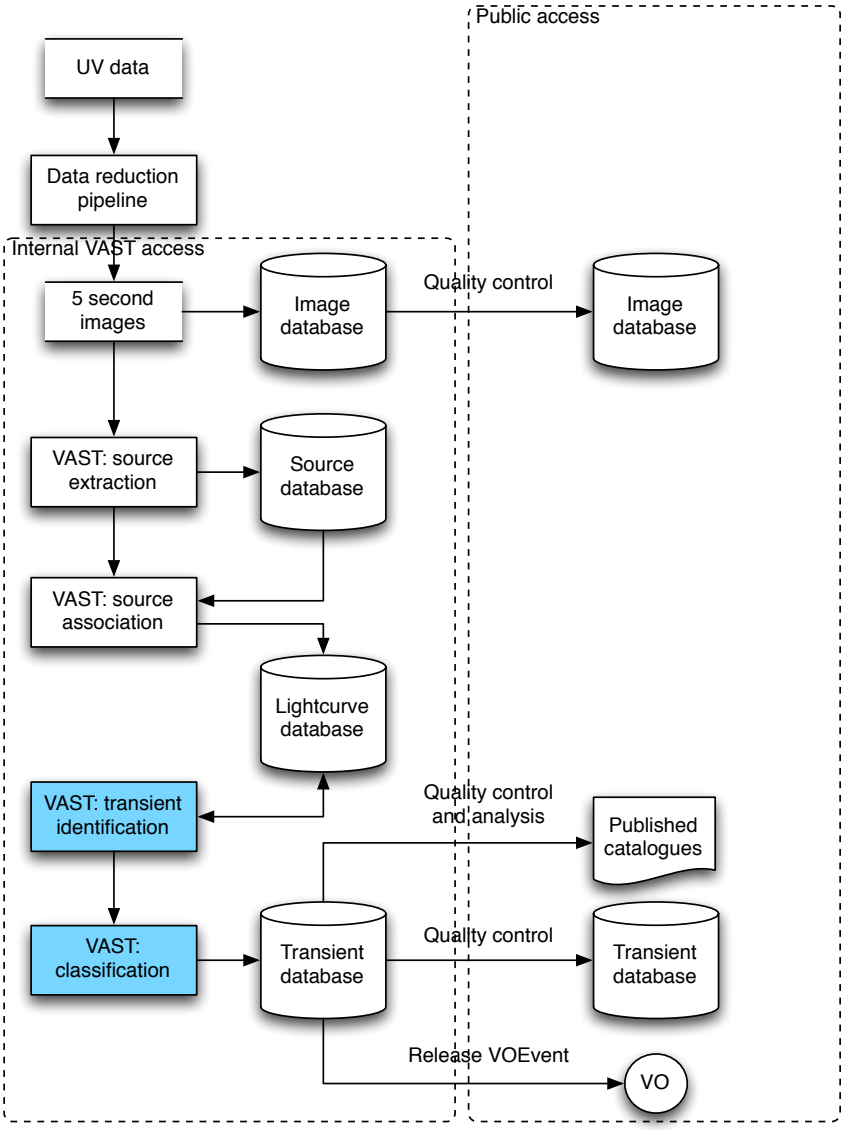


FIGURE B.1: The VAST transient detection and classification pipeline