MNRAS **000**, 1–18 (2021) Preprint 4 April 2021 Compiled using MNRAS LaTeX style file v3.0

Unsupervised machine learning approach to extraction of X-ray variability patterns from light curves of GRS 1915+105

Jakub K. Orwat-Kapola,[1★] Antony J. Bird,[1] Adam B. Hill,[1,2] Diego Altamirano[1] and Daniela Huppenkothen[3]

[1]*School of Physics and Astronomy, University of Southampton, Southampton, Hampshire SO17 1BJ, UK*
[2]*HAL24K Data Intelligence Labs, Herikerbergweg 292, 1101 CT, Amsterdam, the Netherlands*
[3]*SRON Netherlands Institute for Space Research, Sorbonnelaan 3, 3584 CA, Utrecht, the Netherlands*

Accepted XXX. Received YYY; in original form ZZZ

ABSTRACT

Time series data mining is an important field of research in the era of Big Data. Next generation astronomical surveys will generate data at unprecedented rates, creating the need for automated methods of data analysis. We propose a method of light curve pattern extraction using a neural network with Long-Short Term Memory Variational Autoencoder architecture. The network encodes time series into a condensed set of variables in an unsupervised manner. Clustering of the variables reveals an exhaustive set of time series patterns found within the data set. The set of patterns can be readily used as input of subsequent classification and outlier detection algorithms. We use the proposed method on a data set of Rossi X-ray Timing Explorer observations of a galactic black hole X-ray binary GRS 1915+105, which was chosen because of extraordinarily complex variability of its X-ray light curves. We find that the proposed method can generate a representation which characterises the observations and reflects the presence of distinct classes of GRS 1915+105 X-ray flux variability. We find that this representation can be used to perform efficient classification of light curves. We also present how the representation can be used to quantify the similarity of different light curves, highlighting the problem of the popular classification system of GRS 1915+105 observations, which does not account for intermediate class behaviour.

Key words: X-rays: binaries – methods: data analysis

1 INTRODUCTION

Automated approaches to data analysis are becoming increasingly relevant as we are entering the era of big data in and outside of astronomy. Industrial applications include, for example, analysis of smart utility meter data and city traffic data. The ability to identify appliance energy usage patterns can provide actionable insights to utility providers (Singh & Yassine 2018), as smart meters are being installed in millions of houses. Prediction of the rate of traffic flow and smart route planning can aid in the management of congestion in intelligent transportation systems (Zhu et al. 2019). Within the field of astronomy, future and ongoing surveys, like those conducted by Vera C. Rubin Observatory (previously known as the Large Synoptic Survey Telescope (LSST)) (Ivezic et al. 2019) and Zwicky Transient Facility (Bellm 2014), will produce terabytes of data at unprecedented rates, and manual analysis of this volume of data by human experts is impossible. In order to make sense of this data, analysts need methods which can extract descriptive variables from individual data observations (i.e. perform feature engineering). Resulting variables can then be used in machine learning pipelines to compare observations and perform tasks like classification, outlier detection or clustering.

Feature engineering often requires domain specific expertise from the analyst, who needs to identify descriptors containing relevant information about the observations in the data set (for example Richards et al. 2011). Automated feature extraction is an alternative to manual feature engineering, and requires less specific domain knowledge. Automated feature extraction often involves methods which encode data into a more abstract, low-dimensional, latent representation. Interaction of the resulting latent variables is thought to be responsible for all of the significant variance in the data set, and machine learning algorithms can leverage the information they contain. Several methods of automated feature extraction for light curve data (time series data describing intensity of an astronomical object) have been studied in the past, for example Kohonen self-organising maps (Armstrong et al. 2015) and pattern dictionary learning (Pieringer et al. 2019). Mackenzie et al. (2016) extracted light curve segments and clustered them to find common patterns of variability in variable star candidates, creating a representation compatible with machine learning classifiers. Similarly,

[★] E-mail: j.k.orwat-kapola@soton.ac.uk

Valenzuela & Pichara (2018) used a sliding window method to extract light curve segments and classified the light curves based on the presence of characteristic patterns. Charnock & Moss (2017) used a recurrent neural network to process and classify light curves of the Supernovae Photometric Classification Challenge data set and achieved impressive results. Mahabal et al. (2017) transformed the light curves of Catalina Real-Time Transient Survey to two-dimensional images and used a convolutional neural networks to classify them. Naul et al. (2018) extracted features of phase-folded light curves using an autoencoding recurrent neural network and used a random forest classifier on observations of variable sources from All Sky Automated Survey Catalog, Lincoln Near-Earth Asteroid Research survey, and Massive Compact Halo Object Project, achieving classification accuracy of well over 90%.

In this work, we contribute to the toolbox of automated feature extraction methods for light curve data, as we explore the use of neural networks for descriptive feature generation, and demonstrate that such methods can be useful in the analysis of the evolution of a particular source of interest. We analyse the complete data set of GRS 1915+105 observations captured by the Proportional Counter Array on-board of the Rossi X-ray Timing Explorer (RXTE/PCA) (Glasser et al. 1994) between 1996 and 2011, and derive a set of light curve patterns, which can be used in downstream classification, outlier detection and clustering tasks.

GRS 1915+105 is a galactic black hole X-ray binary system discovered in 1992 (Castro-Tirado et al. 1994), which shows an extraordinary complexity of X-ray flux variability. It was the only known source to exhibit such intricate patterns of behaviour, until the discovery of black hole candidate IGR J17091-3624 (Kuulkers et al. 2003; Capitanio et al. 2006), which shares some of the same characteristics (Altamirano et al. 2011).

In an attempt to demonstrate that the complex behaviour of GRS 1915+105 is controlled by just a few simple variables, Belloni et al. (2000) constructed a classification system, which assigned observations of the source to one of 12 classes. Classification was based on the presence of qualitative patterns in light curves and colour-colour diagrams of source observations. Work that followed had expanded the classification system to the total of 14 classes (Klein-Wolt et al. 2002; Hannikainen et al. 2003, 2005). This classification system is hereafter referred to as the "Belloni et al. system". Figure 1 shows an example of GRS 1915+105 light curve for each one of the 14 classes. Some classes show steady flux without any structured variability, other show periodic flares, dips or different types of periodic and aperiodic variability. There are both inter-class and intra-class variations in the amplitude of flux variability.

Highly complex patterns of light curve variability of GRS 1915+105 are commonly interpreted as being caused by a partial or complete disappearance of an observable innermost region of the accretion disc. Disappearance of the disc, in turn, is caused by thermal-viscous instability of the inner region of the disc (Belloni et al. 1997a,b). X-ray variability patterns corresponding to the 14 classes of source behaviour can repeat almost identically months and years apart. This suggests that the instability of the disc triggers a very specific and reproducible response (Belloni 2001).

GRS 1915+105 was the first discovered stellar-mass source producing highly relativistic jets (Mirabel & Rodriguezt 1994). In this regard, it sparked great research interest, as it offered the possibility of studying coupled inflow-outflow processes of an accreting black hole, which unlike more massive active galactic nuclei, evolves in observable time scales (Fender & Belloni 2004). Plasma ejections of GRS 1915+105 in the form of jets have also been associated with the instability of the accretion disc (Belloni et al. 2000; Nayakshin et al. 2000; Fender & Belloni 2004), which supports the notion of disc-jet coupling.

Furthermore, Naik et al. (2002) found that periods when the innermost region of the accretion disc is not observable (which are associated with variability class χ), are preferentially followed by classes showing periodic bursts, i.e. classes ρ and α. Therefore, following the changes of variability classes can improve our understanding of the evolution of the source over longer time scales, and it is an important method of probing the accretion dynamics, as pointed out by Huppenkothen et al. (2017).

Huppenkothen et al. (2017) conducted the first study of the whole set of GRS 1915+105 observations from RXTE/PCA using machine learning. They characterised the observations of the source and classified them according to the Belloni et al. system, using machine learning classification methods. They used features derived from power spectra, light curve features extracted with an autoregressive model, and hardness ratio features derived from energy spectra.

In this work, we perform classification of GRS 1915+105 observations, using exclusively time series features derived from light curve data in an unsupervised manner, using a neural network. We also show that these features can be used to quantify the similarity of observations and perform cluster analysis. We choose not to use energy and power spectral features, found in the works of Belloni et al. (2000) and Huppenkothen et al. (2017), hence making our method more generalisable to other data sets, sources and energy bands. In principle, any type of univariate time series data can be analysed using this method.

This paper contains the following parts: Section 2 describes the data preparation process, which starts with RXTE/PCA observations and produces a data set of light curve segments, a suitable input for a neural network. Section 3 provides details about the proposed neural network and describes the process of dimensionality reduction of the data set, which results in the encoded data representation. Section 4 describes the process of cluster analysis of the encoded representation. Section 4.1 describes how Gaussian mixture models are used to identify the set of light curve patterns from the encoded data representation. Section 4.2 shows how the set of light curve patterns is used to construct a representation referred to as observation "fingerprints". Section 4.3 shows how "fingerprints" can be used to assign light curves to classes of the Belloni et al. system, and Section 4.4 demonstrates how the "fingerprints" can be leveraged to refine the classification system in a data-driven way. Section 5 summarises the main results, discusses limitations of presented approach and lists some ideas for further research.

2 DATA PREPARATION

We retrieve all available RXTE/PCA observations of GRS 1915+105 in Standard-1 format (0.125 second resolution light curves which combine all energy channels over the range of 2 - 60 keV) [1]. Extraction is limited to the most reliable counting array number 2 (PCU2). This yields 1776 light curves, which are subsequently re-binned. We generate two data sets: one where binning is performed at 1 second resolution, and another where binning is performed at 4 second resolution. Two data sets are generated because input size of the neural network is limited (to 128 data points,

[1] https://heasarc.gsfc.nasa.gov/db-perl/W3Browse/w3browse.pl

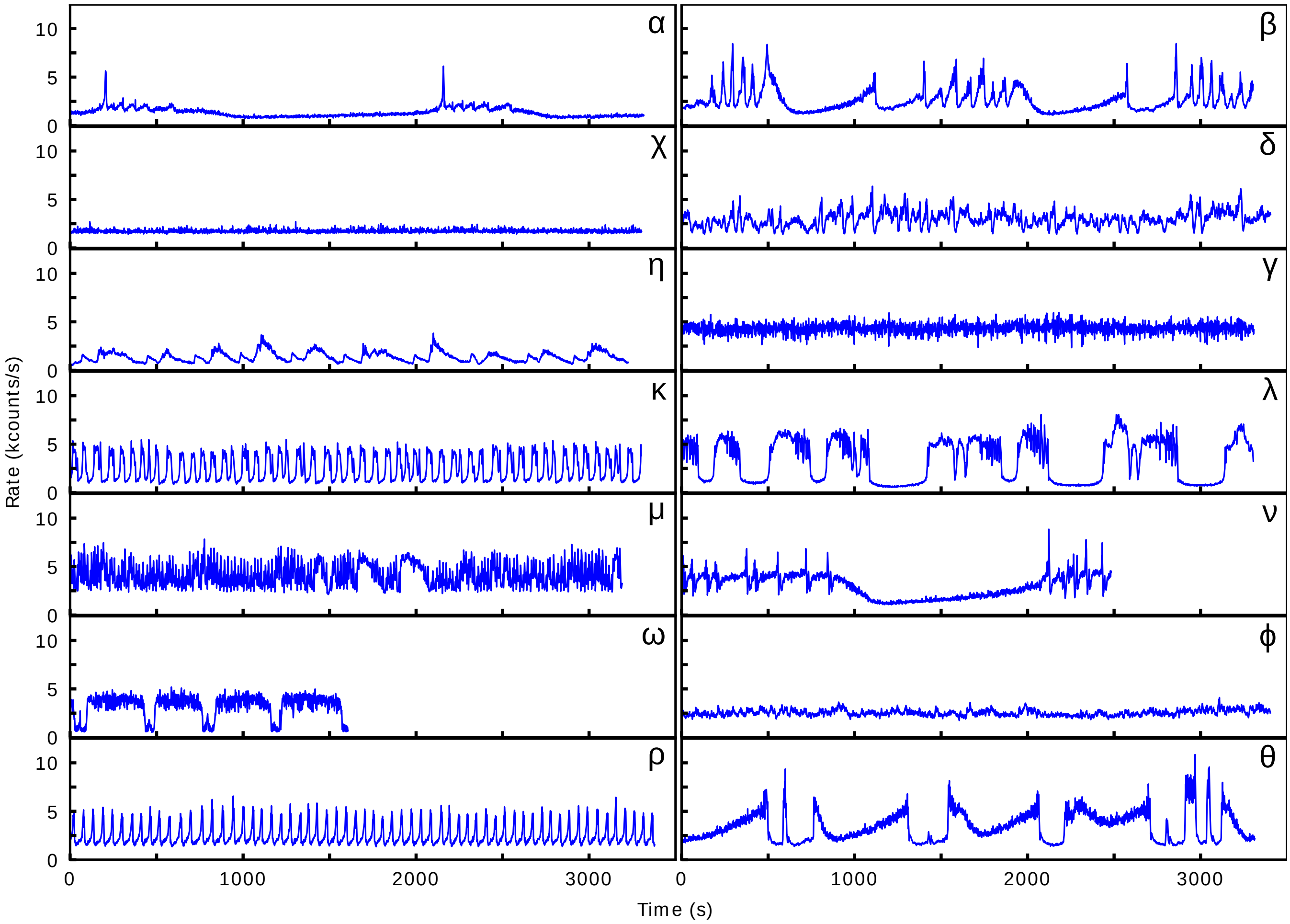


Figure 1. Examples of GRS 1915+105 X-ray light curves. Shown light curves have been classified according to the Belloni et al. system. Classifications are shown in the upper-right corner of each sub-figure. We use the curated set of classifications published by Huppenkothen et al. (2017). Light curves have 1 second cadence.

Table 1. Parameters of the two data sets of light curve segment.

Parameter	4s data set	1s data set
Cadence (s)	4	1
Segment length (s)	512	128
Stride length (s)	8	10
# segments	468,202	474,471

which is explained in more detail in the following paragraphs), and therefore the time resolution of the light curves is the main parameter which influences the amount and type of information that is provided to the network. On the one hand, short time resolution data can resolve fast variability structures of the X-ray source, but it cannot capture longer variability structures in a light curve segment of fixed size. On the other hand, longer time resolution data can capture more context within a light curve segment, but any fast variability structures are smoothed or lost. Hence, we choose to train two separate models on the two data sets and compare them in order to explore the effect of changing data resolution.

Similarly to Huppenkothen et al. (2017), we perform light curve segmentation in order to produce a data set of segments of standard length. Only the good time intervals from each observation are segmented, and the interruption periods of missing data are skipped. We choose the segment size of 128 data points, resulting in segment length of 512 seconds for data with 4 second resolution. A moving window segmentation is performed with a stride of 2 data points (8 seconds), yielding a set of 468,202 segments derived from 1738 observations which satisfy the segmentation criteria. This data set of light curve segments is hereafter referred to as “4s data set”. The same set of 1738 observations, binned to 1 second resolution, is segmented with a stride length of 10 data points, yielding 474,471 segments. This data set of light curve segments is hereafter referred to as “1s data set”. Stride size is adjusted in order to make the number of segments as close as possible to the 4s data set. Table 1 lists parameters of the two data sets.

The time duration captured by the segments is not sufficient to contain the longest cycles of flux variability produced by the source. For example, some observations of class α show intervals of quiescence which last ~1000 seconds, and are interlaced by periods of flaring which last ~500 seconds. However, the main goal of our study is not to classify individual light curve segments, but rather to construct a new, data-driven system of segment templates, which create classifiable observation signatures when grouped together with other segments from the same light curve. See Section 4.3 for an example of a classification experiment which illustrates the use of observation signatures (“fingerprints”).

A small stride size is selected in order to maximise the number of light curve segments available for neural network training. It also ensures that the model is exposed to the full range of phase shift of light curve patterns (Huppenkothen et al. 2017).

Light curve segments are independently standardised; count rate values of each segment are mean centred and scaled to units of standard deviation. Segments are standardised in order to decouple their shape and intensity information. Many of the patterns observed in the variability of GRS 1915+105 repeat at various mean count rate levels. Therefore, standardisation of the segments is likely to cause segments showing similar shape patterns to align in the latent feature space extracted from the data. We also allow for the possibility that some of the shapes could be shared by several classes of variability, as described by Belloni et al. (2000), but at different intensities, and standardisation can make it easier to draw links between those cases.

The resulting data set of light curve segments, together with corresponding count rate uncertainty values (needed to calculate χ_r^2, see Equation 1), is used to train a neural network. The process is described in Section 3.

Original count rate levels of the light curve segments are also important in the data analysis, so intensity information is added to the final feature set in the form of four descriptive statistics; mean, standard deviation, skewness and kurtosis, which are calculated from the distribution of count rate of each segment. These statistics make up one of the two sets of features used in cluster analysis in Section 4, and this set of four intensity features is hereafter referred to as “intensity features of light curve segments” (IFoS).

3 FEATURE EXTRACTION WITH A NEURAL NETWORK

In order to represent the light curve data using a small number of variables, and hence allow us to analyse that data space in a relatively small number of dimensions, we perform dimensionality reduction using a neural network. This process aims to extract a small set of “hidden” variables, which encode information about the structured variability of the light curves. Various methods of modelling of time series data have been employed in numerous fields of research (Längkvist et al. 2014; Hyndman et al. 2015; Benkabou et al. 2018; Ismail Fawaz et al. 2019).

In order to address the problem of dimensionality reduction of light curve data, we propose a Variational Autoencoder (VAE) with Long-Short Term Memory (LSTM) cells within its encoder and decoder blocks. VAE is a type of probabilistic neural network model first proposed by Kingma & Welling (2014). As mentioned, it uses two blocks of neurons; an encoder, which maps input data onto a small set of distributions (commonly referred to as continuous latent variables), and a decoder, which samples from those distributions and maps them back to the input data space, hence reconstructing observations of input data. VAE is therefore trained to effectively perform compression and decompression of input data. The compression process requires the construction of a meaningful latent space of considerably smaller dimensionality than the input. Resulting latent variables are the compressed representation of data and can be leveraged in the data analysis process.

LSTM cells found in the encoder and decoder blocks of the proposed architecture are a type of recurrent neural network (RNN), first proposed by Hochreiter & Urgen Schmidhuber (1997). RNNs learn from sequential data, and their use has been researched extensively for the processing of text, handwriting, speech and sound (Yu et al. 2019, and references therein).

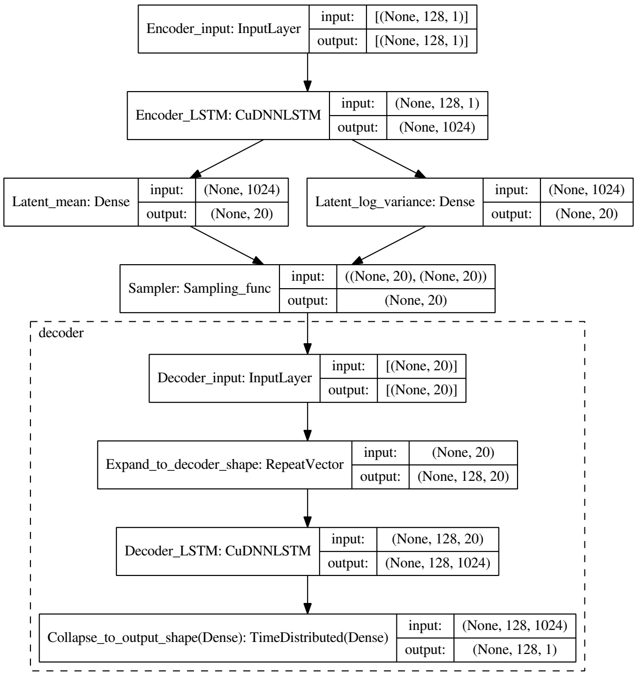


Figure 2. Architecture of the proposed LSTM-VAE model. Figure was generated using the Keras utility `plot_model`. Left-most cell of each block contains the label assigned to corresponding instance of a Keras object, followed by class of Keras model layer (except from `Sampling_func` which is a custom function; see Section 3.1). Right-most cells contain shapes of input and output tensors of the objects. Shapes are presented using the convention followed by Keras.

Cells of an RNN process sequential input one data point at a time. At every iteration, the state of the cell is fed as input of the next iteration, which allows the network to learn from consecutive points of the sequence and extract temporal patterns. RNNs are able to make predictions based on the immediate context of the processed data point, but they tend to quickly forget information that is not frequently reinforced, which means that they struggle to capture long term patterns. LSTM cells address this issue through the introduction of so called “gates” which consist of non-linear functions that control the state of the LSTM cell and protect the relevant information over long time scales.

3.1 Training of the neural network

Both data sets are subdivided into training, validation and testing subsets. In order to ensure that the subsets are independent, segments derived from the same observation are only included in one of the subsets. In order to ensure completeness of the subsets, observations which have Belloni et al. system classifications available (Huppenkothen et al. 2017), are assigned to the subsets in a random, stratified fashion. At least one observation of each class is assigned to each subset, and the remaining observations are assigned to training, validation and testing subsets according to the split ratio of 7/1/2. Since only two observations of η class are available at this stage, both observations are assigned to the training subset. Observations without Belloni et al. system classifications are randomly distributed between the three subsets, while accounting for the fact that observations contain variable number of count rate data points. The resulting training sets contain ~70% of total data points, validation sets contain ~10% of data points, and testing sets contain ~20% of data points.

Two LSTM-VAE neural networks are trained to compress light curve segments of 128 data points into 20 continuous latent variables, one network for each data set. Data from the training set is used to adjust the parameters of the networks, and the validation set is used to measure whether the adjustment improved the networks' ability to process previously unseen examples of data. Testing set is kept aside until all training and fine-tuning of models is finished. We perform dimensionality reduction on both data sets of standardised light curve segments, which are generated using the method described in Section 2).

The networks are built using Keras, an open-source neural network library (Chollet 2015) with Tensorflow backend, and Python 3 programming language[2]. Both networks use identical architecture. Figure 2 shows a visualisation of the proposed LSTM-VAE architecture. The purpose of each layer is as listed below.

- `Encoder_input` creates an instance of a tensor with dimensions of the model input, i.e. a sequence of 128 values.
- `Encoder_LSTM` is a layer of CuDNNLSTM cells. CuDNN stands for the CUDA Deep Neural Network library, which was developed by NVIDIA (Chetlur et al. 2014). The library accelerates training of neural networks using a graphical processing unit (GPU). This layer consists of 1024 such LSTM cells, which are not interconnected, but perform recurrent computation on the input sequence, one data point at a time. Output from every point of the sequence is stored within the state of the cell and used as input of the next computation of the sequence. Output of this layer consists of the final state of the cells, produced after the entire sequence has been processed. LSTM cells are trained to extract informative variables from the data through the process of backpropagation of errors. Increasing the number of LSTM cells tends to improve network's reconstruction loss (see below for the definition of reconstruction loss), and the number of 1024 is selected due to the GPU memory size constrain.
- `Latent_mean` and `Latent_log_variance` are two separate layers of fully interconnected neurons. Their purpose is to perform the dimensionality reduction of the 1024 variables extracted by the `Encoder_LSTM` layer. `Latent_mean` and `Latent_log_variance` each output 20 values (which is the dimensionality of the latent space). First set of 20 values is used as the mean of the continuous latent variables, whereas the other set encodes their spread. In other words, `Encoder_LSTM`, `Latent_mean` and `Latent_log_variance` make up the Encoder block of the VAE, which maps the network input to the mean and (log) variance vectors. Increasing the number of latent variables tends to improve network's reconstruction loss, and the number of 20 is selected as a compromise between the reconstruction performance and the complexity of the latent space.
- `Sampler` generates random numbers from normal distributions whose parameters are set to the values of latent variable mean and variance. It is required in order to allow for deterministic treatment of the inherently probabilistic network during training (the so-called "re-parameterisation trick" (Kingma & Welling 2014)).

Table 2. Summary of the LSTM-VAE training and fine tuning. Validation loss stopped improving after the quoted number of training epochs. Training was stopped after 50 consecutive epochs without validation loss improvement. Best validation loss is shown.

Optimiser (rate)	# epochs	Loss	Data set
Adam (Default)	176	14.44	1s
SGD ($3 \cdot 10^{-4}$)	43	14.20	
SGD ($1.5 \cdot 10^{-4}$)	112	14.17	
SGD ($7.5 \cdot 10^{-5}$)	164	14.16	
Adam (Default)	209	107.63	4s
SGD ($3 \cdot 10^{-4}$)	5	103.67	
SGD ($1.5 \cdot 10^{-4}$)	4	102.55	
SGD ($7.5 \cdot 10^{-5}$)	12	102.02	

- `Decoder_input` initialises the input tensor of the Decoder block of the model.
- `Expand_to_decoder_shape` replicates the values of latent variables to create sequences of the same length as the initial light curve sequences. In other words, each LSTM cell of the Decoder block receives values of the 20 latent variables at each iteration of the sequential computation.
- `Decoder_LSTM` layer is the Decoder counterpart of `Encoder_LSTM` layer of the Encoder. It also performs recurrent computation on the sequential input, but rather than processing a single sequence of variable values, it processes 20 sequences of constant values. This layer has 1024 cells, each producing a sequence of cell states from each iteration of recurrent computation.
- `Collapse_to_output_shape` is a fully connected layer which combines the 1024 sequences from the LSTM layer into a single sequence of 128 data points.

Output of the `Collapse_to_output_shape` layer is the reconstruction of an input light curve segment. Performance of the network is quantified using the loss function, which depends on the reconstruction error and a regularisation term. The loss function is a sum of the reduced chi-squared (Equation 1) and Kullback-Leibler divergence (Equation 2). Reduced chi-squared is defined as:

$$\chi_r^2 = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{x_i - \hat{x_i}}{\sigma_i}\right)^2 \qquad (1)$$

where N is the number of processed data points, x_i is the value of a data point i in the input light curve segment, $\hat{x_i}$ is the value of data point i in the reconstructed sequence, and σ_i is the uncertainty of the value of i in the input light curve sequence. Uncertainty values are scaled with the same value of standard deviation as the count rate values of light curve segments.

Kullback-Leibler divergence is a regularisation term, which prevents the distribution of the latent variables from substantially deviating from the standard normal distribution. This term is used to ensure that the latent space is continuous and meaningful. Kullback-Leibler divergence is defined as:

$$D_{\rm KL} = -\frac{1}{2Z}\sum_{j=1}^{Z}\left(\log(\hat{\sigma_j}^2) - \hat{\mu_j}^2 - \hat{\sigma_j}^2 + 1\right) \qquad (2)$$

where Z is the number of latent variables, $\hat{\sigma_j}^2$ is the variance of latent variable j, and $\hat{\mu_j}$ is the mean of latent variable j.

We train networks using the Keras implementation of Adam optimisation algorithm (Kingma & Ba 2015), and fine-tune them

[2] relevant code for data pre-processing and model training will be available at https://github.com/jorwatkapola/autoencoders-GRS-1915

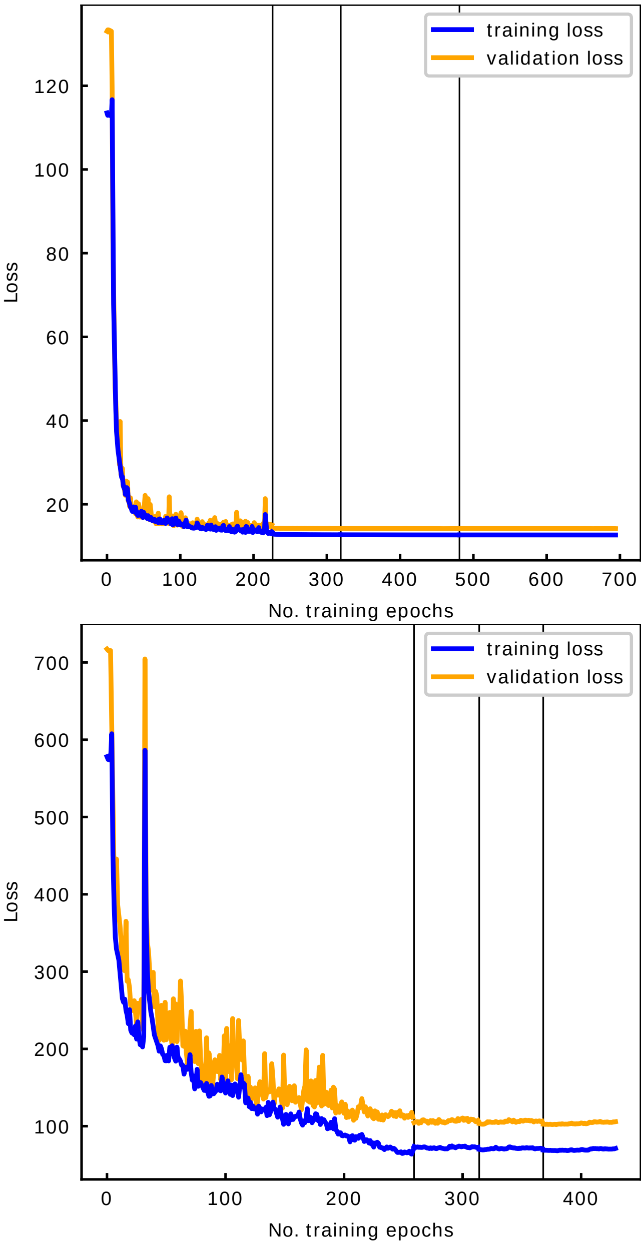


Figure 3. Loss curves resulting from the training of LSTM-VAE models on 1s data set (top) and on the 4s data set (bottom). Black vertical lines indicate that no improvement in validation loss in the last 50 consecutive epochs and change of the optimiser. See Table 2 for full training history.

using the stochastic gradient descend (SGD) algorithm. The `clipvalue` argument is set to 0.5 for both optimisers, which prevents numerical errors due to exploding gradients. Training is performed with the batch size of 2048 (number of light curve segments propagated through the network simultaneously) for various numbers of epochs (i.e. complete passes through the training set). Training is terminated when the validation loss value does not improve for 50 consecutive epochs. See Table 2 for full training history, and Figure 3 for loss curves resulting from the training. Results suggest that satisfactory training results could be achieved using Adam optimiser only, because improvements caused by SGD with decaying learning rate are marginal.

The networks are trained to minimise the value of the loss function, which is dependent on the error of segment reconstruction, as defined in Equation 1, as well as the regularisation term shown in Equation 2. Examples of light curve segments from 1s data set, together with their LSTM-VAE reconstructions are shown in Figure 4. Examples for segments from 4s data set are shown in Figure A1.

3.2 Light curve feature extraction

Outputs of `Latent_mean` and `Latent_log_variance` layers of the model correspond to the position and spread of the set of 20 continuous latent variables. The latent variables are a compressed representation of the network's input. For the purpose of further analysis, we do not sample from latent distributions, but take the means of latent distributions, which are representative of the position of each light curve segment within the latent space. The resulting set of latent variables is hereafter referred to as "shape features of light curve segments" (SFoS). Hence, the shape information of each light curve segments is represented by 20 SFoS values.

In order to assess how well SFoS describe the shape of light curve segments, we perform reconstruction of several segments. Figures 4 and A1 show reconstructions of light curve segments of each class of the Belloni et al. system. This set of segments demonstrates how the LSTM-VAE responds to a range of different light curve patterns found in the data sets. The model is able to reproduce the gross features of each segment, but it often does not account for fast count rate changes, which results in reconstructions which are significantly smoother than the input segments. This means that SFoS likely do not account for differences between segments lacking structured variability, where the major difference lies in the root mean square (RMS) deviation from the mean. Reconstructions of those segments would differ only in terms of the shape of random noise. For example, see segments of class ϕ and χ in Figure A1.

In order to remedy this limitation, IFoS (containing information about count rate mean, standard deviation, skewness and kurtosis of each segment) are used in the cluster analysis stage (Section 4) together with SFoS. Segments with indistinguishable shapes and dissimilar RMS can be distinguished based on their standard deviation values.

Figure 5 shows the distribution of IFoS values for segments of ϕ, χ and γ classes from the 1s data set. As expected, segments of class γ generally have larger mean and standard deviation values than the other two classes, which indicates that IFoS would allow for segments of class γ to be distinguished from classes ϕ and χ, even in segments where the characteristic "dip" of class γ is not observed.

Furthermore, projections of SFoS and IFoS in Figure 6 show that IFoS could be used to distinguish classes ϕ, χ, and γ much more reliably than SFoS alone. The projection was produced using Uniform Manifold Approximation and Projection (UMAP) (McInnes et al. 2018) algorithm, which aims to preserve the global structure of the high-dimensional data. In the projection of SFoS, data classified as ϕ, χ, and γ occupied the same region of UMAP space (in the central cluster of the top sub-figure), indicating that those classes are mostly indistinguishable in the SFoS space. Data classified as ρ tends to occupy separate regions of the SFoS space. Class ρ is included in the projection to demonstrate that data associated with characteristic light curve shapes tends to be distinguishable in terms of its position in the SFoS space. IFoS projection uses the same subset of light curve segment data. Classes still show significant overlap in the IFoS projection, but intensity features can clearly provide

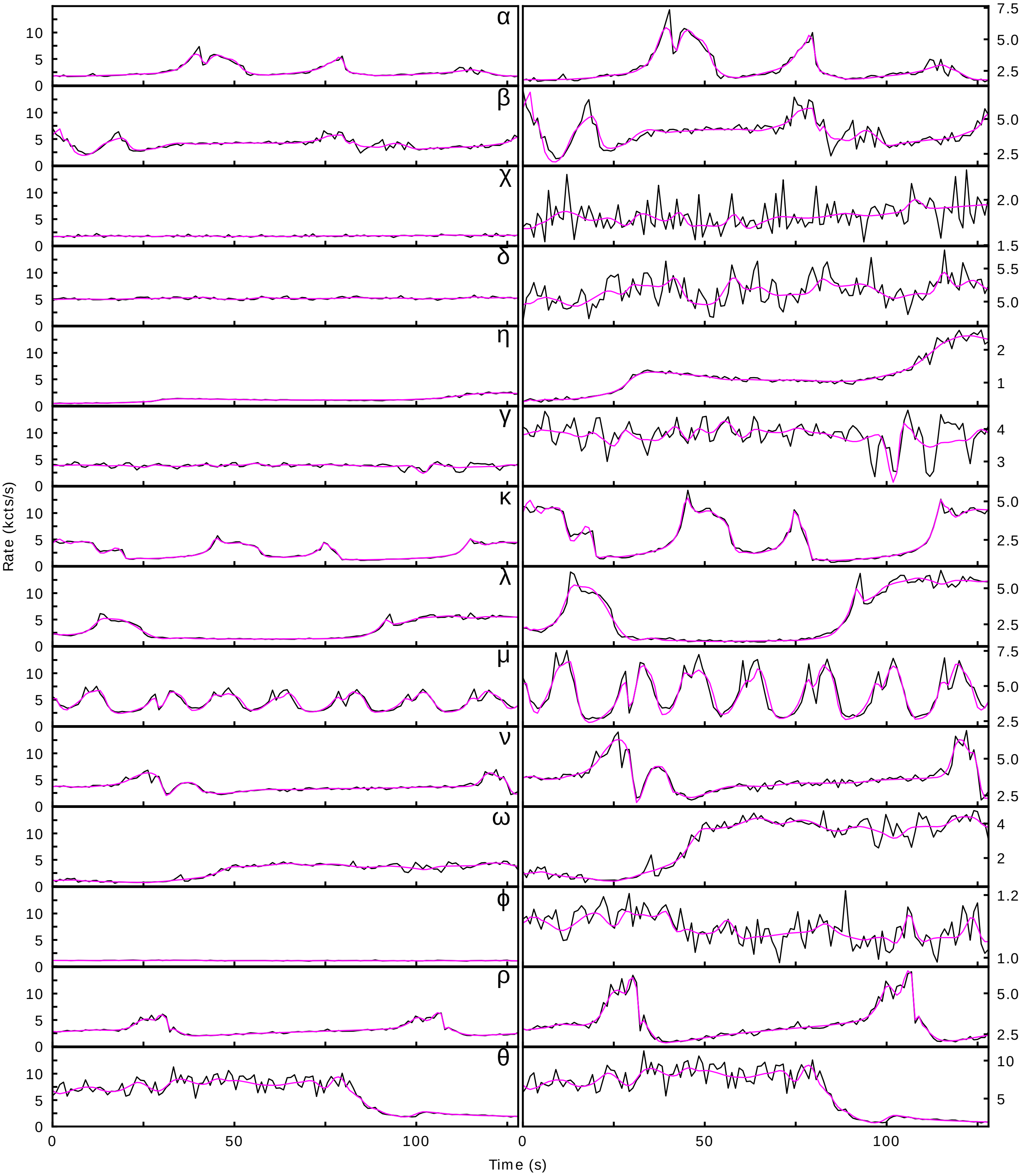


Figure 4. Examples of light curve segments from 1s data set. Segment reconstruction output of the LSTM-VAE is shown in magenta. Segments originate from observations that had been classified according to the Belloni et al. system. We use the curated set of classifications published by Huppenkothen et al. (2017). Both columns shows the same segments, but in the right column the range of the vertical axis is dynamic. All segments come from the testing subset of 1s data set, with exception of class η, which was included only in the training subset.

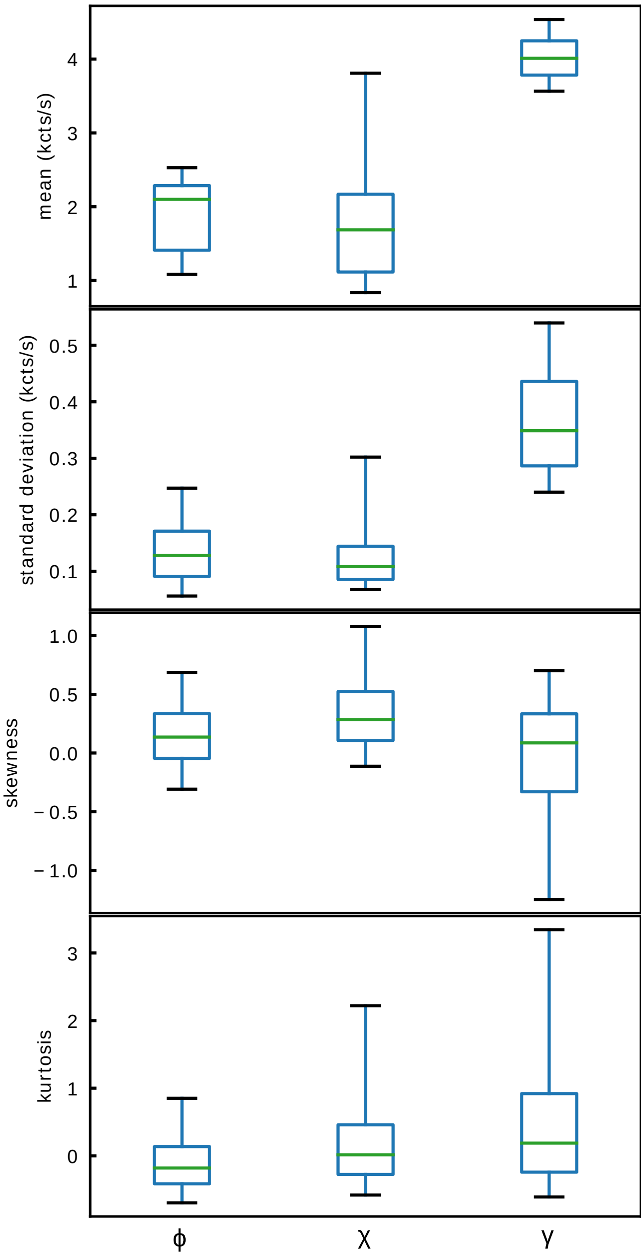


Figure 5. Distributions of the IFoS values from segments assigned to classes ϕ, χ and γ of the Belloni et al. system. Classifications come from the set curated by Huppenkothen et al. (2017). Four IFoS are the mean, standard deviation, skewness and kurtosis of count rate values of each segment. Box extends from the first to third quartile, and the green line shows the median. Whiskers extend from the box to the 5th and 95th percentiles.

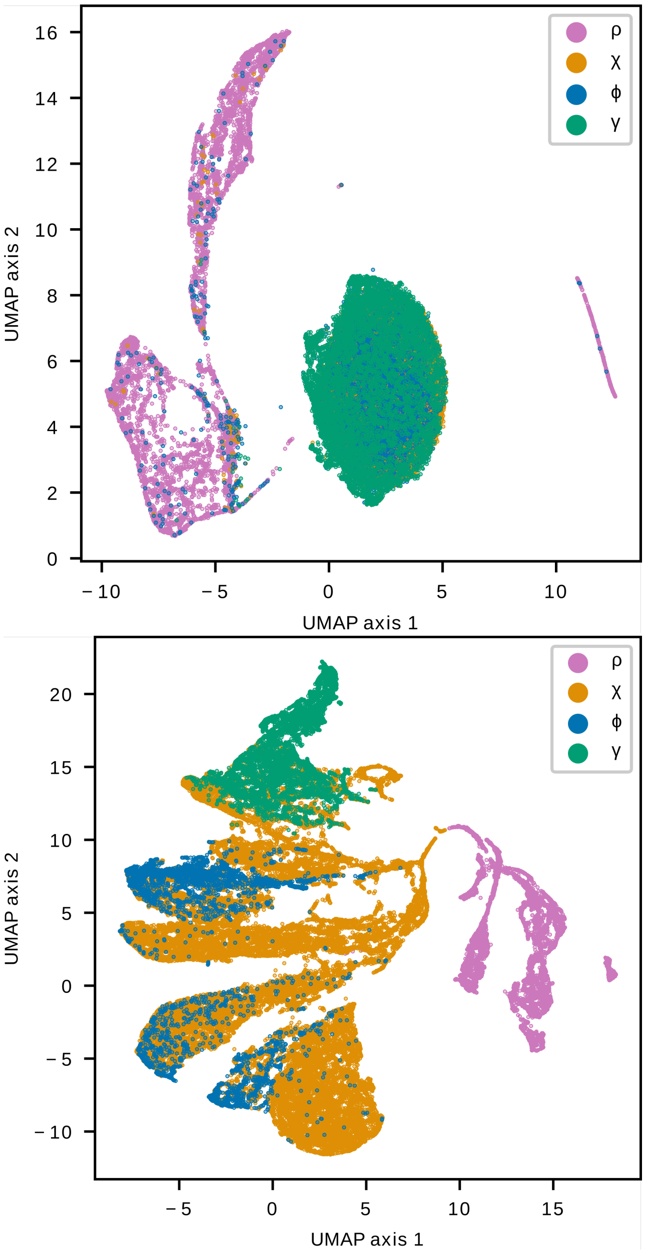


Figure 6. UMAP projection of SFoS (top) and IFoS (bottom) of light curve segments from the 1s data set which have Belloni at al. system classifications of ϕ, χ, γ or ρ. Each point represents a light curve segment, and colour-coding shows their classification according to the Belloni et al. system.

meaningful information about the differences between classes ϕ, χ, and γ.

4 CLUSTER ANALYSIS OF GENERATED FEATURES

4.1 Identifying the set of light curve patterns

Sections 2 and 3 describe the process of feature engineering of four IFoS and 20 SFoS, which encode the shape and intensity information about light curve segments. This set of 24 features is hereafter collectively referred to as “shape and intensity features of light curve segments” (SIFoS). We generate two separate sets of SIFoS, one for the 1s data set, and one for the 4s data set.

In order to find the exhaustive set of light curve pattern templates which have been produced by GRS 1915+105, we perform clustering of the data in the 24 dimensional space of SIFoS. Clustering is performed with an implementation of Gaussian mixture model (GMM) included in the machine learning library scikit-learn (Pedregosa et al. 2011).

GMM uses the Expectation-Maximisation (Dempster et al. 1977) algorithm to approximate the probability distribution of the data using a set of multidimensional Gaussian components. The number of components is a hyper-parameter set by the user, and the mean position of each component is initiated randomly. Position and co-variance matrix of each component are then iteratively optimised to maximise the likelihood of the data under the model. GMM is a “soft” clustering method; likelihood value of each data point is calculated for each Gaussian component, and the data points are assigned to components which give the largest likelihood.

We choose GMM to approximate the shape of data manifold in the SIFoS latent space, with the intention of merging of the Gaussian components which show significant overlap. This way we use combinations of Gaussian components to account for the presence of any extended, curved, non-Gaussian structures in the latent space. We are assuming that those structures correspond to particular light curve patterns, and for each observation of the source, the relative amount of time the source spends showing those patterns allows us to determine the class of that observation.

We perform a hyper-parameter grid search to find the optimal number of Gaussian components for the GMMs in SIFoS space. Figure 7 shows values of the Bayesian information criterion (BIC) of those models as a function of the number of Gaussian components. BIC is a criterion commonly used to select the best model from a set of models fit to the same data set. The number of free parameters is one of the terms of BIC, so it penalises overly complex models. Model which produces the minimum BIC achieves the compromise between complexity and likelihood of the data (see Equation 3).

$$\mathrm{BIC} = k \cdot \log(N) - 2 \cdot \mathcal{L} \qquad (3)$$

where k is the number of free parameters of the model, N is the number of samples of data, and $\mathcal{L}$ is the log-likelihood of the data.

Grid searches indicate that the global BIC minimum for 1s data set is produced by a models with 222 Gaussian components, and for 4s data set with 279 components. We accept those numbers as the optimal numbers of components for GMM, however stochastic nature of the algorithm could cause the numbers to change slightly if the grid-search was to be repeated. The set of clusters resulting from the assignment of light curve segment data points to one of the Gaussian components of GMM is hereafter referred to as “Gaussian clusters”.

Light curve segments showing similar type of variability patterns and count rate distribution are expected to produce similar values of SFoS and IFoS. Consequently, segments showing similar patterns are separated by smaller distances within the SIFoS feature space. Therefore, it is expected that Gaussian clusters contain homogeneous subsets of light curve segments, which are more similar to each other than to segments found in other Gaussian clusters.

Qualitative inspection of populations of the 222 Gaussian clusters of the 1s data set revealed that it is the case indeed. 42 Gaussian clusters contain quasi-periodic flares, characteristic of light curve classes ρ and ν (see Figure 4 for examples). 69 Gaussian clusters contain segments with low RMS and no obvious patterns of variability. Further 15 Gaussian clusters contain segments showing similar type of behaviour, but at higher average count rates. Virtually every one of the remaining Gaussian clusters has some characteristic pattern; irregular flares, dips, negative or positive gradient, flaring followed by quiescence and vice versa. Few Gaussian clusters contain segments whose common patterns of variability are not immediately apparent upon visual inspection of a small random sample of segments, which can indicate that the number of Gaussian components of the GMM is too small.

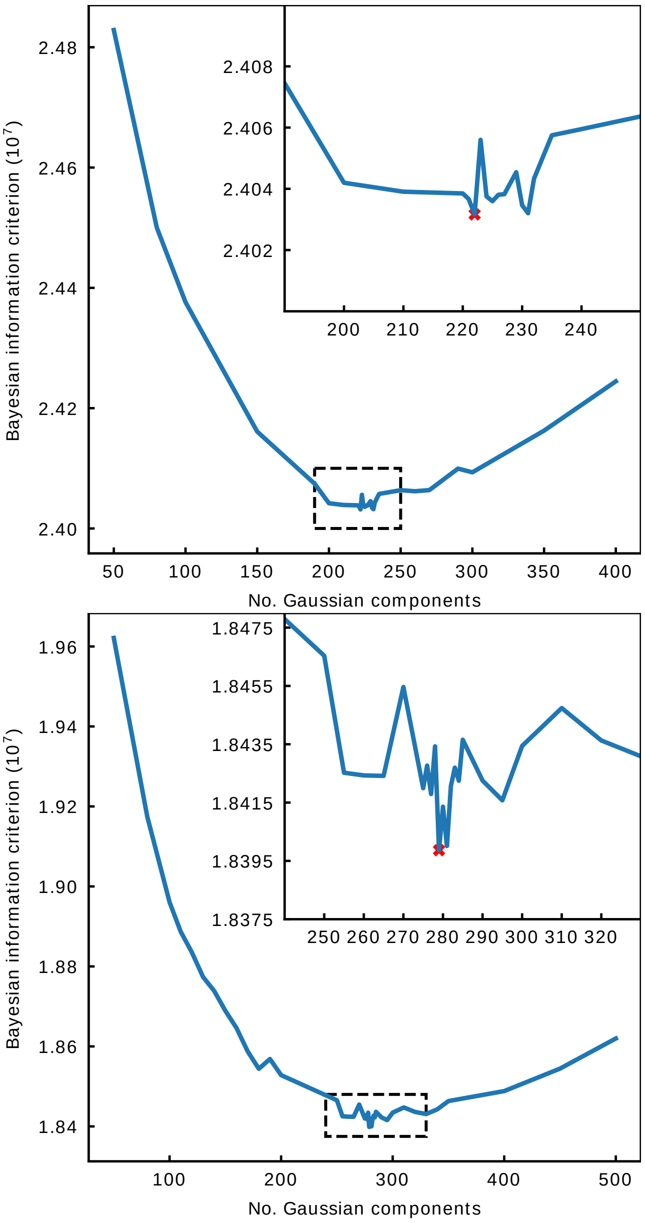


Figure 7. Bayesian information criterion of GMM as a function of the number of Gaussian components. We performed grid searches for the 1s data set (top) and the 4s data set (bottom). Figure insets are zoomed into the regions indicated by the black dotted line boxes, which contain the global minima. Minima were found at 222 Gaussian components for 1s data set and at 279 Gaussian components for 4s data set. Minima are marked by red crosses.

In order to find the minimal, exhaustive set of light curve variability patterns, segments showing the same characteristic patterns should arguably all belong to one cluster. Gaussian clusters produced with the 222 and 279 component GMMs contain apparent degeneracies. The presence of very similar Gaussian clusters is caused by the limitation of the GMM, which approximates the probability density of the data set using multivariate Gaussian components. A single component cannot spread over a curved data manifold; only multiple components can approximate the curvature, as a series of locally flat sections. Light curve segments which show a similar type of variability pattern can vary in more than one SIFoS due to their non-linear interaction within the neural network model.

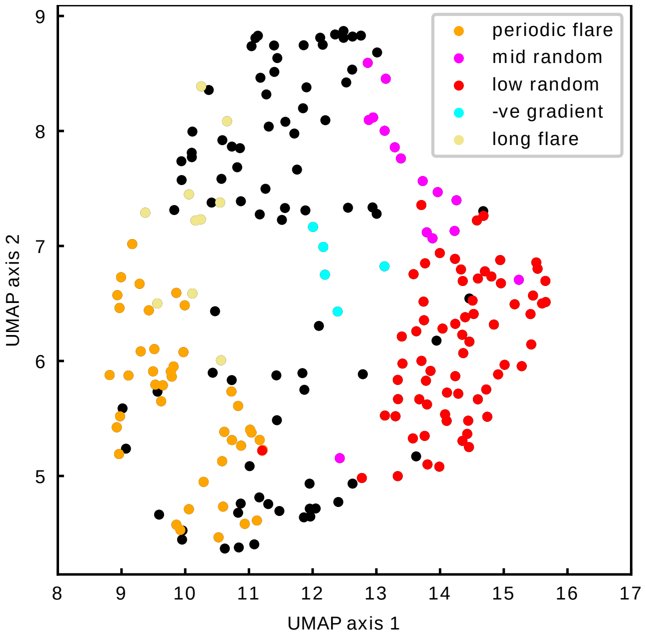


Figure 8. UMAP projection of the 222 means of Gaussian components of the GMM fit to 1s data set. UMAP reduced dimensionality from 24 to 2. Some points are colour coded, and common colours indicate Gaussian clusters containing light curve segments which show common characteristic variability patterns. As mentioned in the text, "periodic flare" stands for behaviour characteristic of light curve classes ρ and ν, "mid random" and "low random" stands for the lack of apparent structured variability (only random noise) at low and medium-high mean count rates, "-ve gradient" stands for slow decrease of mean count rate, whereas "long flare" stands for irregular, long periods of flaring characteristic of classes κ and λ.

Therefore, segments which, for example, vary in the frequency and amplitude of a similar type of pattern, can follow a curved manifold, and hence end up in separate Gaussian clusters. We address that in Section 4.3.

Figure 8 shows a two-dimensional UMAP projection of mean positions of GMM Gaussian components which are fit to the 1s data set. Some of the points are colour coded to indicate the characteristic patterns of light curve segments found in the corresponding Gaussian clusters. Colour coding is produced by manual inspection of data found in Gaussian clusters, which is not a part of the proposed method of unsupervised data analysis. The purpose of this visualisation is to shows that Gaussian clusters sharing the characteristic patterns of behaviour tend to occupy similar regions of the SIFoS space.

4.2 Relating the set of light curve patterns to the Belloni et al. system

In order to transform the two sets of clusters into a sets of observation features, for each observation we count the number of light curve segments assigned to each of the Gaussian components. This results in 222 values per observation in 1s data set and 279 values per observation in 4s data set. For each observation we divide the counts by the sum of all counts for that observation. Feature vectors are independently normalised in order to reduce the impact of variance in the total number of segments extracted from each observation. This variance is caused by the fact that observations vary in total duration and the number of data gaps. Since feature vectors are normalised, they only contain information about the relative abundance of light curve patterns within the corresponding observation. We refer to such 222-vectors and 279-vectors as observation "fingerprints", because they allow identification of distinct classes of light curve variability.

In order to showcase the usefulness of "fingerprint" representation of data, Figure 9 shows the subset of 1s data set which has been human-labelled according to the Belloni et al. system, in terms of the Gaussian clusters described in Section 4. Figure 9 shows combined "fingerprints" for each class of observation ("fingerprints" of observations of the same class were summed to produce the combined "fingerprints"). Rows of this heat map correspond to the 14 classes of observations, and columns correspond to the 222 Gaussian clusters of light curve segments. Particular cells of the heat map reflect the relative abundance of segments of a particular class which have been assigned to a particular Gaussian cluster. Heat map was normalised row-wise, which means that high values indicate the Gaussian clusters which are most closely associated with a particular light curve class. This in turn indicates which light curve patterns are most abundant in light curves of a particular class.

The distinct appearance of the rows of the heat map indicates that the representation of light curves in terms of Gaussian clusters allows us to distinguish observations of different classes of the Belloni et al. system. This indicates that the light curve representation which employs the set of patterns could serve as a viable feature space for supervised classification algorithms.

4.3 Classification of light curves using "fingerprint" representation

In order to test the usefulness of the Gaussian cluster representation in light curve classification, we train random forest classifiers (Breiman 2001) to assign observations to the Belloni et al. system based on their "fingerprints" (note: we choose to disregard the subdivision of the χ class, because the main distinguishing feature of those sub-classes is the position in the colour-colour diagram). We perform classification with the random forest classifier implementation included in the machine learning library scikit-learn (Pedregosa et al. 2011). We train separate classifiers for 1s data set and 4s data set.

In order to address the issue of degeneracy of light curve patterns described in Section 4.1 we merge Gaussian clusters separated by small distances. We use the Mahalanobis distance metric between the mean positions of Gaussian components. Mahalanobis distance is a distance metric used to measure the distance between a point and a distribution, while scaling the distance using the variance of that distribution. Mahalanobis distance is defined as:

$$D_{\mathrm{M}} = \sqrt{(\boldsymbol{u}-\boldsymbol{v})\mathbf{V}^{-1}(\boldsymbol{u}-\boldsymbol{v})^T} \tag{4}$$

where u and v are vectors whose separation is calculated, and V is the co-variance matrix.

The distance threshold between means of Gaussian components which are merged is one of the hyper-parameters included in the grid-search during the training of random forest classifiers. Gaussian components are allowed to merge if the distance calculated for both co-variance matrices is smaller than the threshold hyper-parameter.

We use the training and validation data subsets (described in Section 3.1) for the training of random forest classifiers. We find that the label of observation 10258-01-10-00 from Huppenkothen et al. (2017) disagrees with preceding literature (Klein-Wolt et al.

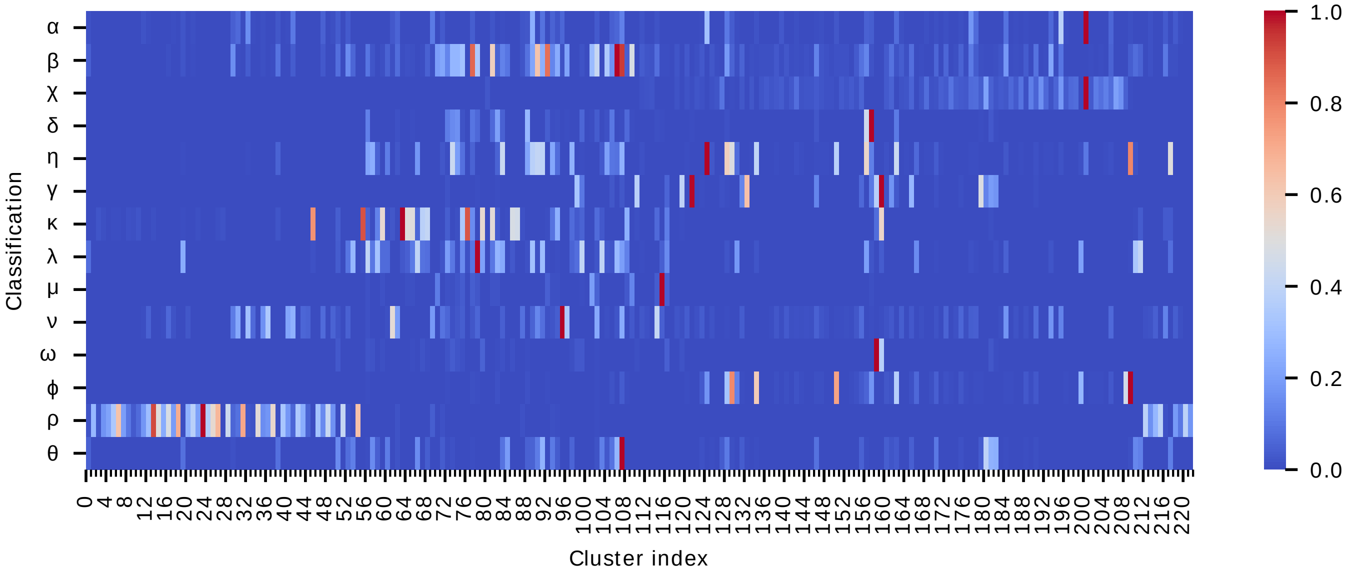


Figure 9. Heat map showing the distribution of light curve segments across the set of 222 Gaussian clusters which were fit to the classified subset of 1s data set. Heat map was normalised row-wise. Colour indicates the relative abundance of light curve segments in the corresponding cluster. Clusters are ordered based on their proximity in the SIFoS space; this was determined using a hierarchical (single linkage) clustering algorithm. Observations' class labels come from Huppenkothen et al. (2017).

Table 3. List of hyper-parameters included in the grid-search of classification experiment described in Section 4.3. Hyper-parameters criterion and max_depth belong to the random forest classifier, as defined in the documentation of scikit-learn library (Pedregosa et al. 2011). "Merge distance" refers to the Mahalanobis distance threshold used in the process of merging Gaussian clusters. We test 100 values evenly spaced between 1.5 and 5. Hyper-parameter producing the largest F1 values for the two data sets are also listed.

Hyperparameter	Possible values	1s data set	4s data set
criterion	gini, entropy	gini	gini
max_depth	None, 5, 10, 15, 25	5	15
merge distance	between 1.5 and 5	3.34	2.84

2002; Belloni & Altamirano 2013), therefore we change the label from λ to μ prior to classifier training.

Due to the limited number of labelled observations, we do not perform n-fold cross-validation, but instead we train the random forest classifier on 137 observations sampled from the training and validation subsets in a random, stratified manner, and we validate the classifier using the remaining 22 observations. We repeat this process 100 times for each combination of hyper-parameters and find the mean of performance scores resulting from the 100 validation trials. In order to account for the class imbalance, the classifiers automatically adjust weights of each training sample to be inversely proportional to class frequencies. Table 3 lists hyper-parameters included in the grid-search.

We keep the values of n_estimators, min_samples_split, min_samples_leaf hyper-parameters constant at 100, 2 and 1 respectively. n_estimators controls the number of decision trees in the random forest ensemble, and increasing the number tends to reduce the variance of predictions, as the forest converges on the answer, but at the expense of increased computation time. Therefore, we keep this number constant for all the grid-search classifiers, and set the hyper-parameter to 1000 for the final classifier used on the testing data subset. We set min_samples_split and min_samples_leaf to their default, minimum values due to the small number of observations relative to the number of classes in our experiment. The hyper-parameters control the splitting of the decision trees, and increasing their values would not allow for the classes with just a single observation to be separated from other classes.

We find that the highest average validation performance scores for the 1s data set are weighted F1 of 0.814 ± 0.065 and accuracy of 0.854 ± 0.054, while the highest average validation scores for the 4s data set are 0.760 ± 0.068 and 0.810 ± 0.056 (reported uncertainty values are equal to one standard deviation calculated from the performance scores of the 100 validation trials). Therefore, we conclude that features derived from 1s data set perform better in the task of light curve classification. Hyper-parameters producing the highest average validation scores are listed in Table 3 for both data sets.

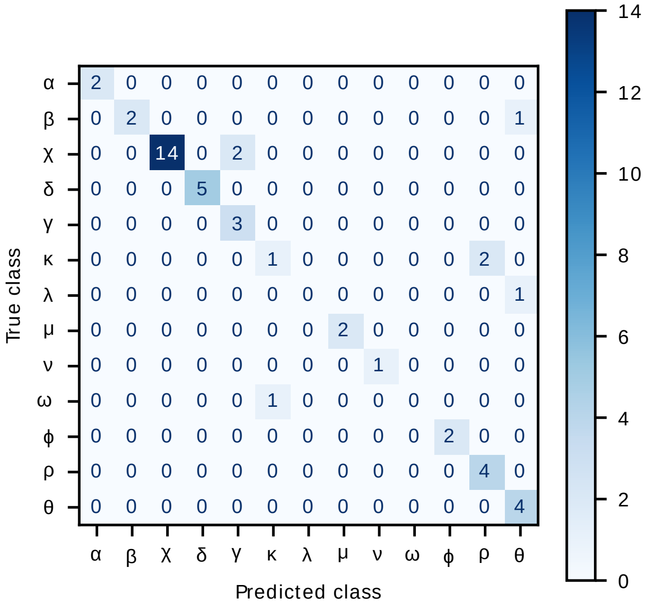
Both F1 and accuracy scores can take values in the range between 0 and 1, the higher the better. Accuracy is the proportion of correct classifications out of all predicted classifications. F1 score is the harmonic mean of recall and precision of classifications of a single class:

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{5}$$

where recall is the proportion of true positives out of the sum of all positives, and precision is the proportion of true positives out of the sum of true positives and false negatives.

Reported F1 scores are weighted averages across the 14 classes. The scores are weighted by the number of observations of the corresponding class in order to account for the class imbalance. In general, weighted F1 is a more reliable performance indicator, because accuracy can be easily biased when observations of one class significantly outnumber observations of other classes.

A random forest classifier with the optimal set of hyper-parameters is trained on the observations of the training subset



	α	β	χ	δ	γ	κ	λ	μ	ν	ω	ϕ	ρ	θ
α	2	0	0	0	0	0	0	0	0	0	0	0	0
β	0	2	0	0	0	0	0	0	0	0	0	0	1
χ	0	0	14	0	2	0	0	0	0	0	0	0	0
δ	0	0	0	5	0	0	0	0	0	0	0	0	0
γ	0	0	0	0	3	0	0	0	0	0	0	0	0
κ	0	0	0	0	0	1	0	0	0	0	0	2	0
λ	0	0	0	0	0	0	0	0	0	0	0	0	1
μ	0	0	0	0	0	0	0	2	0	0	0	0	0
ν	0	0	0	0	0	0	0	0	1	0	0	0	0
ω	0	0	0	0	0	1	0	0	0	0	0	0	0
ϕ	0	0	0	0	0	0	0	0	0	0	2	0	0
ρ	0	0	0	0	0	0	0	0	0	0	0	4	0
θ	0	0	0	0	0	0	0	0	0	0	0	0	4

Figure 10. Confusion matrix showing classification results produced by the random forest classifier with optimal set of hyper-parameters for 47 testing set observations. The matrix shows results for the lowest weighted F1 score out of the one thousand random initiations of the classifier. Weighted F1 and accuracy are 0.834 and 0.851 respectively.

and tested on the testing subset of 1s data set, containing 47 observations. Random initiation of the algorithm is a significant cause of variance in the model performance, therefore training and testing is repeated one thousand times. Mean of weighted F1 and accuracy performance scores of those classifications are 0.878 ± 0.027 and 0.894 ± 0.027 respectively. It should be noted that reported uncertainty values account only for variance caused by changes to the initial random state of the classifier.

Figure 10 shows the classification results with the lowest performance scores out of the set of one thousand testing trails. It seems that our classifications disagrees with classifications from Huppenkothen et al. (2017) the most for classes which have few observations available. Therefore it is likely that classification performance could improve given a larger amount of training data.

Observations of class β get assigned to other classes most likely because of the complex behaviour of its light curves. Some of the patterns of class β can be seen in light curves of other classes (Belloni et al. 2000). Observation with ID 40703-01-35-01 belonging to class β (Klein-Wolt et al. 2002) is assigned to class θ. This observation contains many periods of missing data, and the good time intervals contain dips similar to the class θ. The observation also contains W-shaped intervals which resemble those characteristic of class θ. Furthermore, the classifier predicts that β is the second most probable classification of this observation (top three predictions are θ (37.1%), β (30.4%) and δ (6.1%)).

Two observations of class χ, 10408-01-42-00 and 40703-01-20-03, are assigned to class γ. Both observations show significantly higher count rate and RMS then an average χ observation, which is a possible cause of their classification. Class χ is the second most probable class prediction for both observations; top three predictions for 10408-01-42-00 are γ (21.8%), χ (21.8%) and ϕ (14.8%), whilst for 40703-01-20-03 they are γ (21.9%), χ (21.9%) and ϕ (14.7%).

Two observations of class κ, 40703-01-24-00 and 40703-01-25-00, are assigned to class ρ. Both observations show fairly regular, sharp flares, similar to those characteristic to class ρ, however they are noticeably wider than the canonical ρ flares (see Figure 11 for the light curves of the two observations). Other than the similarity to ρ light curves, another factor influencing the classification of these observations is likely the heterogeneity of the light curve behaviour of observations labelled as κ by Klein-Wolt et al. (2002). More details about the ambiguity of κ classifications is provided in Section 4.4. Furthermore, class κ is the second most probable class prediction for both observations. Top three predictions for 40703-01-24-00 are ρ (16.0%), κ (15.5%) and χ (14.0%), whilst for 40703-01-25-00 they are ρ (15.7%), κ (15.3%) and χ (13.4%).

Observation 20402-01-36-01 belongs to class λ and is assigned to class θ by the random forest classifier. This observation shows behaviour which is very characteristic of class λ; it shows periods of flaring which alternate with low, quiet periods (see λ light curve in Figure 1 for an example). The likely cause of this class assignment is the fact that only one λ observation is included in the training data subset. The classifier predicts that λ is the second most probable classification; top three predictions are θ (30.1%), λ (20.8%) and κ (13.2%).

Observations 10408-01-19-00 and 10408-01-19-02 belong to class ϕ and are assigned to class χ by the random forest classifier. They both show count rate values which are lower than those in training ϕ observations, and since there exist χ observations in that range of count rate, it is a factor that likely influences the classification. As noted by Belloni et al. (2000), the two classes are best told apart based on the hardness of their colour-colour diagrams. Class ϕ is the second most probable class prediction for both observations; top three predictions for 10408-01-19-00 are χ (31.7%), ϕ (31.0%) and γ (14.4%), whilst for 10408-01-19-02 they are χ (31.6%), ϕ (27.3%) and γ (16.5%).

Finally, observation 40703-01-29-01 belonging to class ω is assigned to class κ by the classifier. This observation shows steady flux with no structured variability except a singe W-shaped dip, which is a very typical ω behaviour (see Figure 11 for the light curve). The classifier predicts that ω is the second most probable classification for this observation (top three predictions are κ (29.9%), ω (21.0%) and γ (16.8%)). One cause contributing to this classification is the fact that only one labelled observation of class ω is available in the training data subset. However, another major cause becomes apparent upon inspection of ω and κ observations. Many of the observations labelled as κ by Klein-Wolt et al. (2002) show behaviour which very strongly resembles class ω. We discuss this issue in more detail in Section 4.4.

4.4 Data-driven review of ω and κ classifications

Figure 12 shows the “fingerprint” representation of all ω and κ observations from Huppenkothen et al. (2017). There clearly exist at least two groups of κ observations. Six of the observations (40703-01-14-00/01/02, 40703-01-30-00/01/02) are much more similar to ω observations than the other κ observations. In order to assess the similarity of presented observations, we perform hierarchical clustering of observations in the “fingerprint” space. We use the hierarchical clustering algorithm included in the SciPy package (Virtanen et al. 2020). Clustering is done using the ward method and euclidean metric.

Figure 13 shows a dendogram resulting from the clustering of “fingerprints” of ω and κ observations (see Figure 11 and Figure 14 for their light curves). In the green branch of the dendogram, observations classified as ω by Klein-Wolt et al. (2002) are clustered with

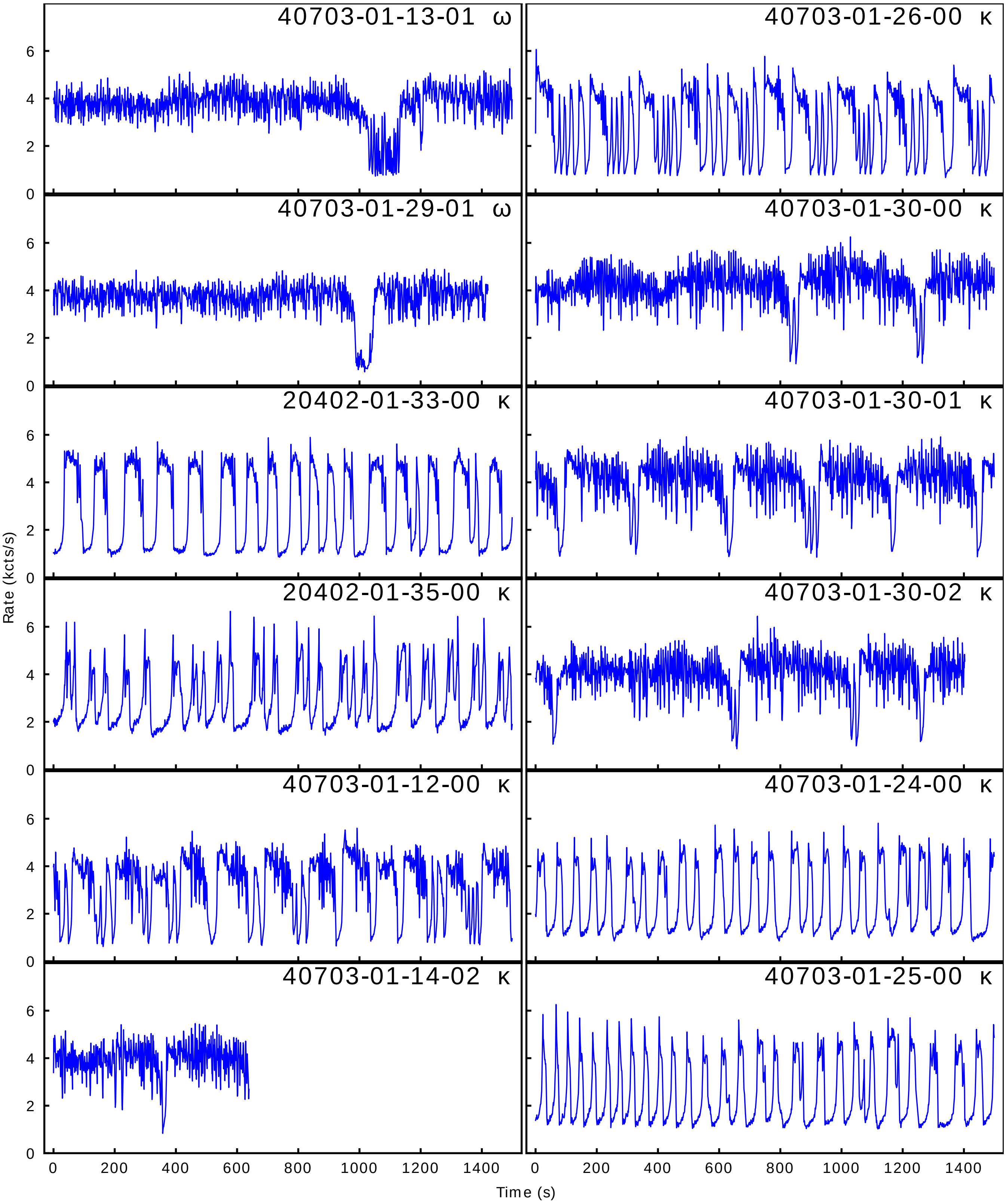


Figure 11. Light curves of ω and κ observations discussed in Section 4.4. Each subfigure contains first 1500 seconds of the light curve or as much as is available in case of shorter observations. Each subfigure contains the observation ID and classification from Huppenkothen et al. (2017). Figure 14 contains light curves which require individual time axes.

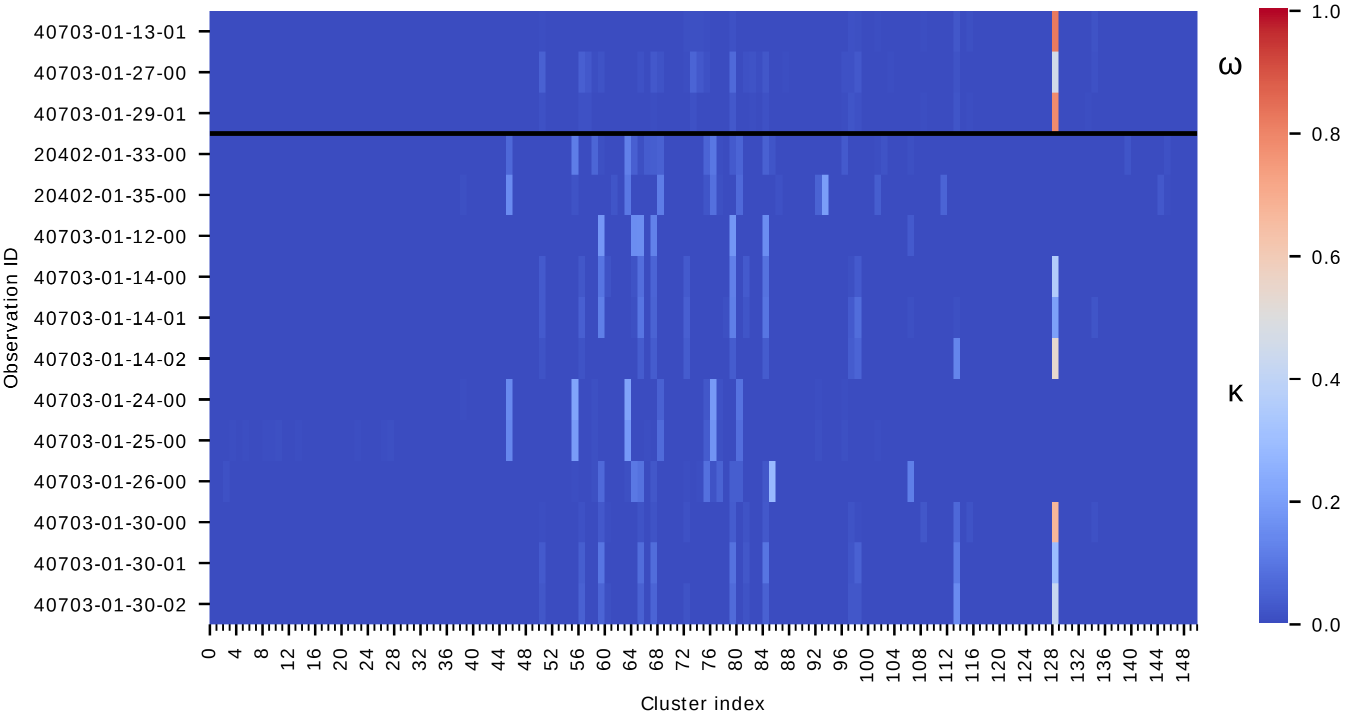


Figure 12. "Fingerprint" representation of ω (above black line) and κ (below black line) observations from Huppenkothen et al. (2017). Clusters are merged using the method described in Section 4.3, using the optimal Mahalanobis distance threshold of 3.34. Colour indicates the relative abundance of light curve segments in the corresponding cluster.

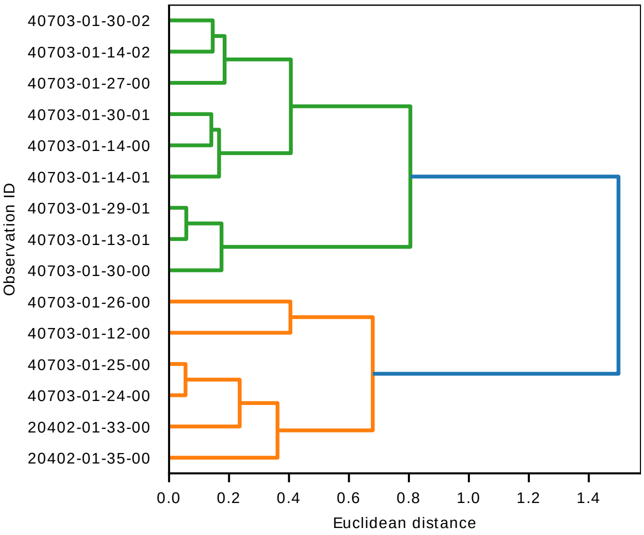


Figure 13. Dendogram resulting from the hierarchical clustering of ω and κ observations based on their "fingerprints" shown in Figure 12. Hierarchical clustering algorithm uses the ward method and euclidean metric. Splitting of branches of the dendogram at smaller values of Euclidean distance indicates that the observations in corresponding leaf nodes are more closely related.

the six κ observations mentioned above. These observations show semi-regular dips with one or more re-flares. Observations 40703-01-13-01/29-01/30-00 clustered in the lower green sub-branch show lower frequency of dipping than the other observations, making them more alike to the canonical ω behaviour. Observations 40703-01-30-02/14-02/27-00 show a slightly higher dipping frequency, whilst observations 40703-01-30-01/14-00/14-01 show the highest frequency of dips in the green branch of the dendogram. Similarity between "fingerprints" of κ observations 40703-01-14-00/01/02, 40703-01-30-00/01/02, and canonical ω observations is the result of the similarity between the light curves of those observations. Based on this similarity, we suggest that those κ observations should be viewed as examples of an intermediate κ/ω state, with a major ω component.

Observations in the orange branch of the dendogram show characteristic quasi-periodic and aperiodic flares of class κ. However, observations 40703-01-12-00/26-00 show a mix of flares with large and small width as opposed to 40703-01-24-00/25-00 and 20402-01-33-00/35-00, show a more consistent flare profile. Moreover, the presence of flares with greater width suggests an intermediate κ/ω state with a major κ component. Furthermore, Belloni & Altamirano (2013) indicate that observations 40703-01-12-00/26-00 belong to class ω, whilst both Pahari & Pal (2010) and Belloni & Altamirano (2013) indicate that observations 40703-01-14-00/30-00 belong to class ω, which shows that ambiguity of κ and ω classification exists in the literature.

Table 4 shows a chronological list of observations captured in the periods when the source was showing κ and ω behaviour, along with the classifications from Klein-Wolt et al. (2002), Pahari & Pal (2010) and Belloni & Altamirano (2013). Observations of class κ and ω were observed in close succession, which supports the notion of intermediate states between those classes. Based on the classification of observations in the two periods shown in Table 4, it seems that the source tends to transition between classes in the order $\kappa \rightarrow \omega \rightarrow \kappa$.

However, clustering results of the "fingerprint" representation

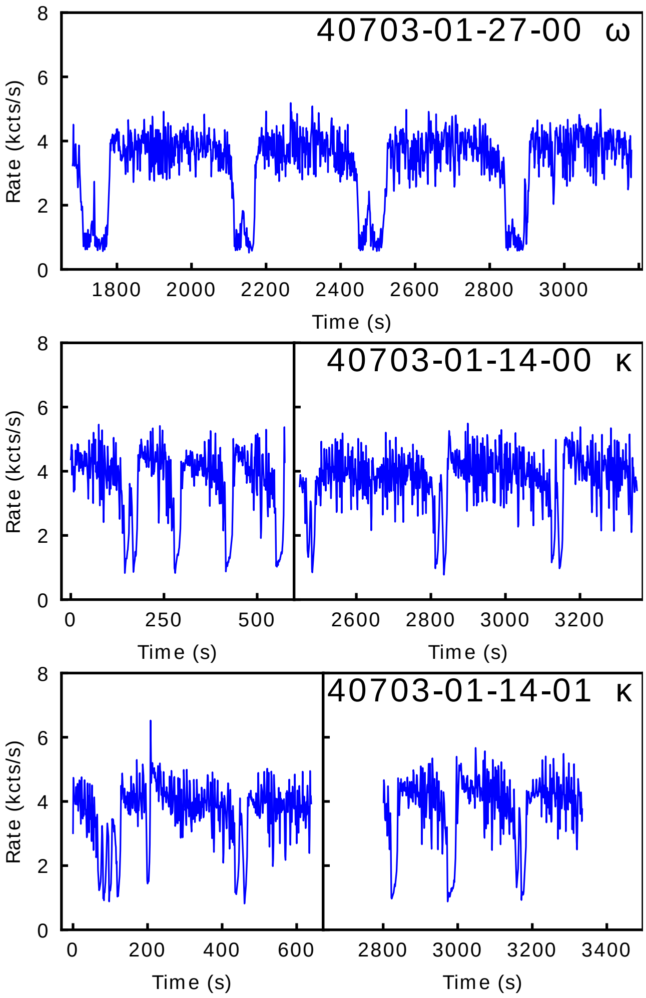


Figure 14. Light curves of ω and κ observations which are mentioned in Section 4.3. Top subfigure contains only the second, longer good time interval of the observation. Middle and bottom subfigures contain two good time intervals of their respective observations, and the data gaps are removed. Each subfigure contains 1500 seconds of the light curve or as much as is available in case of shorter observations. Each subfigure shows the observation ID and classification from Huppenkothen et al. (2017). Figure 11 contains light curves of remaining ω and κ observations.

Table 4. Chronological list of RXTE/PCA observations of GRS 1915+105 during periods 51288-51306 MJD and 51394-51432 MJD. Classifications from Klein-Wolt et al. (2002) (Class. K), Pahari & Pal (2010) (Class. P) and Belloni & Altamirano (2013) (Class. B) are provided. Observations whose classifications seem to be inconsistent are indicated in bold.

Observation ID	Date (MJD)	Class. K	Class. P	Class. B
40703-01-12-00	51288	κ	ω	ω
40115-01-01-00	51288	-	-	-
40403-01-07-00	51291	ω	ω	ω
40703-01-13-00	51299	γ	-	-
40703-01-13-01	51299	ω	ω	ω
40115-01-02-00	51299	-	-	-
40703-01-14-00	51306	κ	ω	-
40703-01-14-01	51306	κ	-	-
40703-01-14-02	51306	κ	-	-
40703-01-24-00	51394	κ	-	-
40703-01-25-00	51399	κ	-	-
40115-01-05-00	51406	-	-	-
40703-01-26-00	51407	κ	-	ω
40703-01-27-00	51413	ω	-	-
40703-01-27-01	51413	γ/ω	ω	-
40703-01-28-00	51418	ω	-	ω
40703-01-28-01	51418	-	-	-
40703-01-28-02	51418	ω	ω	ω
40115-01-06-00/01	51423	-	-	-
40703-01-29-00	51426	ω	ω	ω
40703-01-29-01	51426	ω	-	ω
40703-01-29-02	51426	γ/ω	-	ω
40703-01-30-00	51432	κ	ω	-
40703-01-30-01	51432	κ	-	-
40703-01-30-02	51432	κ	-	-
40703-01-30-03	51432	γ/κ	-	-

should be interpreted with caution due to the geometry of the data sampling space (see Section 5 for more details).

5 CONCLUSIONS

We introduce a data-driven method of light curve feature extraction and test the utility of resulting features by conducting a set of supervised multi-class classification experiments, using a set of human-labelled observations. Light curve classification of data in 1 second resolution resulting in a mean weighted F1 score of 0.878 suggests that the proposed method of unsupervised feature extraction is capable of producing features which represent light curve data in a meaningful way.

In regard to the data set of GRS 1915+105 X-ray light curves, one possible application of the method is the unsupervised exploration of the data space. The shape of the latent data manifold encodes information about modes of activity of the source and its evolution between them. Hence, dimensionality reduction of the data set, followed by clustering of observations in this space could be a way to derive a set of classes of source behaviour, which avoids the biases of human characterisation and annotation of data. Furthermore, we find that Gaussian component merging based on the Mahalanobis distance between them can help to reduce the problem of cluster degeneracy caused by the limitations of GMM, which requires multiple components to follow curved data manifolds.

The Belloni et al. system of classification discussed in this work is not comprehensive, and to some degree, it is arbitrary. As Belloni et al. (2000) point out, their classification system was not intended to exhaustively list mutually exclusive modes of behaviour. This creates a problem for any classification effort that is based on this system. A smaller number of classes could be chosen, because some classes show behaviour that arguably lies on the same continuum. One example of such continuum could be followed by classes λ and κ, which show similar behaviour at slightly different time scales (Belloni et al. 2000). Furthermore, our work shows that classes κ and ω can show very similar light curve behaviour and possibly lie on a continuum as well. A larger number of classes could just as well be required, as it is possible that there exist additional patterns of behaviour that have not been characterised yet (similarly to classes ω and ξ (a.k.a. η) which were added to the Belloni et al. system later than the original 12). Transitions between classes are observed, so ambiguity in classification of observations cannot be avoided.

Taking these considerations into account, it is very difficult to ensure the accuracy of classification for large sets of unknown data, which cannot be standardised to adhere to the assumed classification system. Performance of any supervised classification algorithm greatly depends on the definition of the classes of observations, and any ambiguity is going to affect this performance. Therefore, clustering of the data set can help in deriving a data-driven set of observation classes, which helps to avoid the biases of human characterisation and annotation of data.

However, further work in this direction will need to address the problem of clustering of compositional data. The "fingerprint" representation uses vectors of fixed size, whose values sum up to a positive constant. This is a type of compositional data, and as such it is constrained to the geometry of a simplex (Aitchison & Egozcue 2005), which means that results of clustering of raw compositional data are often unreliable. Clustering methods applied to compositional data should take into account the need of prior transformation of the data into an unbounded, euclidean space. This issue is an open area of research, and it is further complicated by the presence of a large number of zero values in the "fingerprint" compositional data (Aitchison et al. 2000; Martín-Fernández et al. 2003). Study of the appropriate transformation methods for compositional data goes beyond the scope of our work, therefore results of "fingerprint" clustering should be interpreted with caution.

Since the proposed feature extraction method is easily generalisable to different types of time series data, there exists a range of possible applications for the proposed feature extraction pipeline. A similar type of analysis is possible for sources other than GRS 1915+105, and in principle any energy band of light curves.

Derivation of fingerprints which represent the set of light curve patterns observed in a fixed amount of time could be the basis of a live monitoring system, which would alert the user about changes in the behaviour of the source. This could involve classifying observations using a known system of classes, but it could also involve the task of outlier detection, where position of an observation would be tracked within the encoded feature space. Observations producing feature vectors which fall in sparse regions of the feature space would indicate an anomaly.

The main requirement of the proposed feature extraction method is that the light curves must be evenly sampled in time. This requirement is satisfied by data similar in nature to the pointed observations of RXTE/PCA. This includes data captured by the X-ray Telescope aboard the Neil Gehrels Swift Observatory (Swift) (Burrows et al. 2005) or X-ray Timing Instrument aboard the Neutron Star Interior Composition Explorer (NICER) (Gendreau et al. 2016) etc. High-speed optical telescopes ULTRACAM (Dhillon et al. 2007) and HiPERCAM (Dhillon et al. 2016) also produce light curves which are evenly sampled over the time of an exposure, and could be analysed using the method we propose. However, the amount of data produced by ULTRACAM and HiPERCAM is not large enough to justify the use of machine learning analysis. The Optical Timing Camera (OPTICAM) (Castro et al. 2019) will produce a larger amount of data of similar nature, and the proposed method of feature extraction could be appropriate for their analysis.

The proposed method could be used to characterise long-term light curves captured by all-sky X-ray surveys, like those performed with the Gas Slit Camera aboard MAXI (Matsuoka et al. 2009), the Burst Alert Telescope aboard Swift Gehrels et al. (2004) or the All Sky Monitor aboard RXTE, provided that light curves have regular time bins. Interpolation could also be performed if needed.

Pursiainen et al. (2020) interpolated ca. 30,000 light curves of the Dark Energy Survey Supernova Programme using Gaussian Processes, and increased the cadence from the average of 7 days to a constant 0.5 day cadence (see also Wiseman et al. 2020). Light curves generated by several ground-based surveys could be made compatible with our feature extraction method using such interpolation techniques. Those surveys include LSST, ZTF and Asteroid Terrestrial-impact Last Alert System (ATLAS) (Heinze et al. 2018). Optimisation of the type and size of kernel used in Gaussian Processes interpolation would need to be performed by the user prior to feature extraction, and the choice of parameters would depend on the nature of data and the scientific goal.

In addition to the requirement of having even sampling in time, light curves should not have many gaps which cannot be interpolated over, in order to be suitable for analysis using the proposed feature exposure method. During the light curve segmentation stage, segments are extracted using a moving window method, and segments which straddle over data gaps are discarded. Therefore, light curves must have few gaps to allow for a choice of segment size which is large enough to encompass time-scales which are relevant for the variability of the analysed data.

There are several limitations to this work. Figure 5 reveals that classes ϕ and χ are similar in terms of IFoS. The only significant differences are (1) that mean count rate values of segments of class χ span a larger range than the segments of class ϕ, and (2) that segments of class χ can have more positive outliers in terms of kurtosis. Figure 6 shows that the combination of IFoS contains enough information to distinguish many cases of those classes, but significant areas of overlap still exist. The two classes are best distinguished based on the hardness of their colour-colour diagrams (Belloni et al. 2000). Since both classes of observations show no structured X-ray variability patterns, and their IFoS distributions largely overlap, it should be noted that classification attempts based on features presented in this work could disagree with classification of Belloni et al. (2000). The proposed method is only able to capture time series patterns, therefore it would not be able to differentiate between observations which differ only in terms of energy spectra. Nevertheless, users of the method could choose to supplement SIFoS with additional features, dependent on their individual use case.

Another set of limitations involves data pre-processing. The re-binning of light curve data to 1 and 4 second resolution was a compromise between computational tractability and descriptiveness of GRS 1915+105 variability. Even though the shortest timescale of significant X-ray flux changes in this source is ca. 5 seconds (Nayakshin et al. 2000), binning inevitably led to a loss of some of the fast variability information. Furthermore, the choice of light curve segment length was influenced by computational constrains, but also informed by the previous work and knowledge of the time scales of light curve patterns in GRS 1915+105. There is a fairly large degree of tolerance for the choice of these parameters, but some knowledge of the time scales of interest was required. For cases where relevant time scales are difficult to predict, multiple "fingerprints" could be derived from data in a range of temporal resolutions, and subsequently concatenated to create a single feature vector. However such an approach would increase the noise in feature vectors, so further work is required to test this.

Our main assumption was that light curve segments exhibiting similar type of variability patterns and count rate distribution had similar values of SFoS and IFoS. Segments with more similar values of SIFoS were consequently separated by smaller distances within the SIFoS feature space. Therefore, it was expected that Gaussian clusters contained homogeneous subsets of light curve segments, which were more similar to each other than to segments found in other Gaussian clusters. Inspection of Gaussian cluster populations revealed that this assumption was justified to a large degree, however fitting a GMM with a larger number of Gaussian components would provide a greater precision of density estimation of data in the feature space, and hence reduce the risk of non-homogeneity of created clusters. On the other hand, such an approach would likely aggravate the issue of pattern degeneracy in the resulting set of light curve patterns, which would potentially lead to more complex "fingerprints" and ultimately to a more noisy feature set. Merging of clusters based on the Mahalanobis distance can help to reduce the number of degenerate features, but in this work we rely on the classified subset of data to find the optimal distance threshold. In cases where data classification is not possible, unsupervised approaches would need to be used instead. Unsupervised methods of evaluating clustering performance could be used instead, for example Silhouette Coefficient, Calinski-Harabasz Index and Davies-Bouldin Index included in the scikit-learn library (Pedregosa et al. 2011), however computational cost of such an approach might be higher.

DATA AVAILABILITY

The data underlying this article are publicly available in the High Energy Astrophysics Science Archive Research Center (HEASARC) at https://heasarc.gsfc.nasa.gov/db-perl/W3Browse/w3browse.pl, and in the repository of the article of Huppenkothen et al. (2017) at https://github.com/dhuppenkothen/BlackHoleML.

ACKNOWLEDGEMENTS

We thank Tomaso Belloni for sharing his work on GRS 1915+105. We also thank Kevin Alabarta, Mehtap Ozbey Arabaci, Philip Wiseman, Miika Pursiainen and data scientists of HAL24K Data Intelligence Labs for useful discussion and advice. DA and ABH acknowledge support from the Royal Society.

REFERENCES

Aitchison J., Egozcue J. J., 2005, Mathematical Geology, 37, 829
Aitchison J., Barceló-Vidal C., Martín-Fernández J. A., Pawlowsky-Glahn V., 2000, Mathematical Geology, 32, 271
Altamirano D., et al., 2011, The Astrophysical Journal Letters, 742
Armstrong D. J., et al., 2015, Monthly Notices of the Royal Astronomical Society, 456, 2260
Bellm E. C., 2014, in The Third Hot-wiring the Transient Universe Workshop. pp 27–33, http://arxiv.org/abs/1410.8185
Belloni T., 2001, in , Vol. 567, The Neutron Star-Black Hole Connection. Kluwer Academic Publishers, pp 295–300
Belloni T. M., Altamirano D., 2013, Monthly Notices of the Royal Astronomical Society, 432, 10
Belloni T., Méndez M., King A. R., Van Der Klis M., Van Paradijs J., 1997a, The Astrophysical Journal Letters, 479, 145
Belloni T., Méndez M., King A. R., Van Der Klis M., Van Paradijs J., 1997b, The Astrophysical Journal Letters, 488, 109
Belloni T., Klein-Wolt M., Mendez M., van der Klis M., van Paradijs J., 2000, Astronomy and Astrophysics, 355, 271
Benkabou S.-E., Benabdeslem K., Canitia B., 2018, Knowledge and Information Systems, 54, 463
Breiman L., 2001, Machine Learning, 45, 5
Burrows D. N., et al., 2005, Space Science Reviews, 120, 165
Capitanio F., et al., 2006, The Astrophysical Journal, 643, 376
Castro-Tirado A. J., Brandt S., Lund N., Lapshov I., Sunyaev R. A., Shlyapnikov A. A., Guziy S., Pavlenko E. P., 1994, Astrophysical Journal Supplement, 92, 469
Castro A., et al., 2019, Revista Mexicana de Astronomia y Astrofisica, 55, 363
Charnock T., Moss A., 2017, The Astrophysical Journal, 837, L28
Chetlur S., Woolley C., Vandermersch P., Cohen J., Tran J., Catanzaro B., Shelhamer E., 2014, cuDNN: Efficient Primitives for Deep Learning, http://arxiv.org/abs/1410.0759
Chollet F., 2015, Keras, https://keras.io
Dempster A. P., Laird N. M., Rubin D. B., 1977, Journal of the Royal Statistical Society, Series B, 39, 1
Dhillon V. S., et al., 2007, Monthly Notices of the Royal Astronomical Society, 378, 825
Dhillon V. S., et al., 2016, in Proceedings of the SPIE. pp 99080Y–undefined, doi:10.1117/12.2229055, http://arxiv.org/abs/1606.09214http://dx.doi.org/10.1117/12.2229055
Fender R., Belloni T., 2004, Annual Review of Astronomy and Astrophysics, 42, 317
Gehrels N., et al., 2004, The Astrophysical Journal, 611, 1005
Gendreau K. C., et al., 2016, in Space Telescopes and Instrumentation 2016: Ultraviolet to Gamma Ray. SPIE, p. 99051H, doi:10.1117/12.2231304
Glasser C. A., Odell C. E., Seufert S. E., 1994, IEEE Transactions on Nuclear Science, 41, 1343
Hannikainen D. C., et al., 2003, Astronomy & Astrophysics, 411, 415
Hannikainen D. C., et al., 2005, Astronomy & Astrophysics, 435, 995
Heinze A. N., et al., 2018, The Astronomical Journal, 156
Hochreiter S., Urgen Schmidhuber J., 1997, Neural Computation, 9, 1735
Huppenkothen D., Heil L. M., Hogg D. W., Mueller A., 2017, Monthly Notices of the Royal Astronomical Society, 466, 2364
Hyndman R. J., Wang E., Laptev N., 2015, in 2015 IEEE International Conference on Data Mining Workshop (ICDMW). pp 1616–1619, doi:10.1109/ICDMW.2015.104
Ismail Fawaz H., Forestier G., Weber J., Idoumghar L., Muller P. A., 2019, Data Mining and Knowledge Discovery, 33, 917
Ivezic Z., et al., 2019, The Astrophysical Journal, 873, 111
Kingma D. P., Ba J., 2015, in 3rd International Conference on Learning Representations. http://arxiv.org/abs/1412.6980
Kingma D. P., Welling M., 2014, in Proceedings of the 2nd International Conference on Learning Representations (ICLR). http://arxiv.org/abs/1312.6114
Klein-Wolt M., Fender R. P., Pooley G. G., Belloni T., Migliari S., Morgan E. H., van der Klis M., 2002, Monthly Notices of the Royal Astronomical Society, 331, 745
Kuulkers E., Lutovinov A., Parmar A., Capitanio F., Mowlavi N., Hermsen W., 2003, The Astronomer's Telegram, 149, 1
Längkvist M., Karlsson L., Loutfi A., 2014, Pattern Recognition Letters, 42, 11
Mackenzie C., Pichara K., Protopapas P., 2016, The Astrophysical Journal, 820, 138
Mahabal A., Sheth K., Gieseke F., Pai A., Djorgovski S. G., Drake A., Graham M., Collaboration t. C., 2017, in IEEE Symposium Series on Computational Intelligence (SSCI). pp 2757–2764, doi:10.1109/SSCI.2017.8280984
Martín-Fernández J. A., Barceló-Vidal C., Pawlowsky-Glahn V., 2003, Mathematical Geology, 35, 253
Matsuoka M., et al., 2009, Publications of the Astronomical Society of Japan, 61, 999
McInnes L., Healy J., Melville J., 2018, UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, http://arxiv.org/abs/1802.03426
Mirabel I. F., Rodriguezt L. F., 1994, Nature, 371, 46
Naik S., Agrawal P. P. C., Rao P. A. R., Paul B., 2002, Monthly Notices of the Royal Astronomical Society, 330, 487
Naul B., Bloom J. S., Pérez F., Van Der Walt S., 2018, Nature Astronomy, 2, 151
Nayakshin S., Rappaport S., Melia F., 2000, The Astrophysical Journal, 535, 798
Pahari M., Pal S., 2010, Monthly Notices of the Royal Astronomical Society, 409, 903
Pedregosa F., et al., 2011, Journal of Machine Learning Research, 12, 2825

Pieringer C., Pichara K., Catelán M., Protopapas P., 2019, Monthly Notices of the Royal Astronomical Society, 484, 3071
Pursiainen M., et al., 2020, Monthly Notices of the Royal Astronomical Society, 494, 5576
Richards J. W., et al., 2011, The Astrophysical Journal, 733
Singh S., Yassine A., 2018, Energies, 11
Valenzuela L., Pichara K., 2018, Monthly Notices of the Royal Astronomical Society, 474, 3259
Virtanen P., et al., 2020, Nature Methods, 17, 261
Wiseman P., et al., 2020, Monthly Notices of the Royal Astronomical Society, 498, 2575
Yu Y., Si X., Hu C., Zhang J., 2019, A review of recurrent neural networks: Lstm cells and network architectures, doi:10.1162/neco_a_01199
Zhu L., Yu F. R., Wang Y., Ning B., Tang T., 2019, Big Data Analytics in Intelligent Transportation Systems: A Survey, doi:10.1109/TITS.2018.2815678

APPENDIX A: LSTM-VAE RECONSTRUCTIONS OF 4S DATA SET SEGMENTS

This paper has been typeset from a TEX/LATEX file prepared by the author.

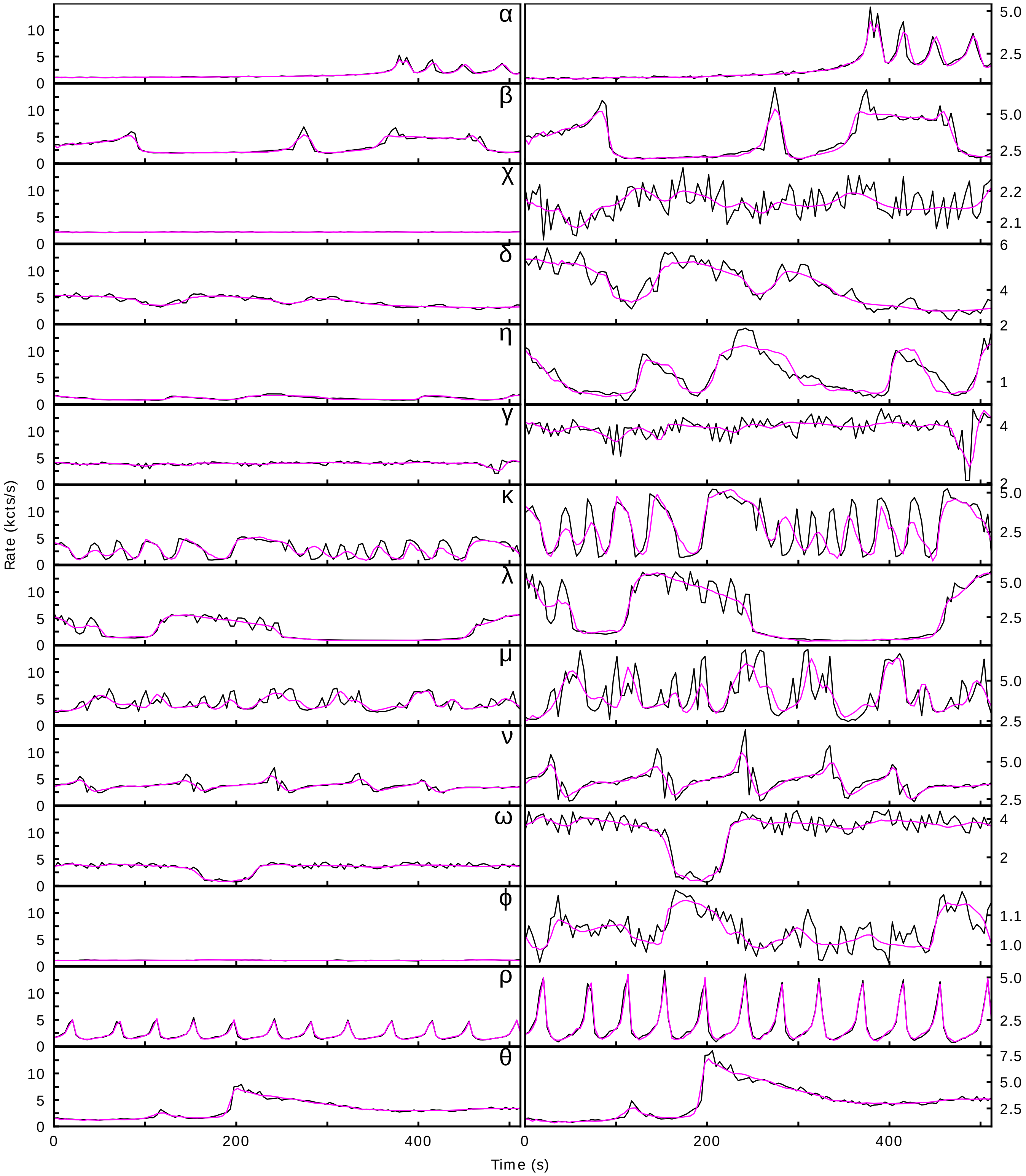


Figure A1. Examples of light curve segments and their LSTM-VAE reconstructions - the counterpart of Figure 4 for 4s data set.