Generalised Interpretable Shapelets

Terry Lyons[1,2]

[1] Mathematical Institute, University of Oxford
[2] The Alan Turing Institute, British Library
{tlyons}@maths.ox.ac.uk

Abstract

The shapelet transform is a form of feature extraction for time series, in which a time series is described by its similarity to each of a collection of 'shapelets'. However existing work has suffered from several limitations, such as expensive training procedures, loss of interpretability, fragility to noise, and a requirement for regularly sampled data. In this work, we demonstrate how these issues may collectively be overcome, and furthermore how the procedure may be generalised in multiple ways. This produces a method that normalises its input data, is straightforward to implement, and whose results are interpretable. We validate our method on several datasets, such as TODO. (TODO: talk about the successes of our method.)

1 Introduction

Shaplets are a form of feature extraction for time series. Given some fixed hyperparameter K, describing how many shapelets we are willing to consider, then each time series is reduced to a vector of length K describing how similar it is to each of the k selected shapelets.

We begin by stating the classical definition of shapelets.

1.1 Classical shapelets

Given N regularly sampled multivariate time series, with D observed channels, where the i-th time series is of length T_n, then the n-th time series is a matrix

$$f^n = (f^n_t)_{t\in\{0,\dots,T_n-1\}} = (f^n_{t,d})_{t\in\{0,\dots,T_n-1\},d\in\{1,\dots,d\}}, \tag{1}$$

with each $f^n_{t,d} \in \mathbb{R}$. We assume without loss of generality that $0, \dots, T_n - 1$ are the times at which each sample is observed, so that the parameterisation t corresponds to the time of an observation.

Fix some hyperparameter $K \in \mathbb{N}$, which will describe the number of shapelets. Fix some $S \in \{0, \dots, \min_{i\in\{1,\dots,N\}} T_n - 1\}$, which will describe the length of each shapelet. We define the k-th shapelet as a matrix

$$w^k = (w^k_t)_{t\in\{0,\dots,S-1\}} = (w^k_{t,d})_{t\in\{0,\dots,S-1\},d\in\{1,\dots,d\}},$$

with each $w^k_{t,d} \in \mathbb{R}$.

Then the discrepancy between f^n and w^k is defined by:

$$\sigma_S(f^n, w^k) = \min_{s\in\{0,\dots,T_n-S\}} \sum_{t=0}^{S-1} \|f^n_{s+t} - w^k_t\|_2^2, \tag{2}$$

Preprint. Under review.

where $\|\cdot\|_2$ describes the L^2 norm on $\mathbb{R}^D$. A small discrepancy implies that f^n and w^k are similar to one another.

Given some already-selected w^k, then this corresponds to sweeping it over f^n, and finding the offset s at which it best matches f^n. The collection of $(\sigma_S(f^n, w^1), \ldots, \sigma_S(f^n, w^K)) \in \mathbb{R}^K$ is now a feature describing f^n. This may now be passed to some model to perform classification or regression.

This method is attractive for two reasons. First, it is invariant to the value of T_n, and as such provides a way to normalise variable-length time series. Second, it is interpretable, as use of a particular feature corresponds to the importance of the similarity to the shapelet w^k, which may for example describe some shape that is characteristic of a particular class; furthermore the value of s gives where the similarity occurs.

1.2 Limitations

However, classical shapelet methods also suffer from a number of limitations.

1. The technique only applies to regularly spaced time series, due to the minimisation over s.
2. The choice of S is a hyperparameter; it is discrete, and choosing it is thus a relatively expensive optimisation procedure.
3. The technique is not robust to irrelevant channels (which will typically exist in many real world datasets, for example medical time series): equation (2) attempts to fit a $w^k_{t,d}$ even for uninformative channels d.
4. Determining the choice of w^k is either expensive, following the procedure of [**?**], or loses interpretability, following the procedure of [**?**].
5. The formulation of equation (2) has essentially made several ad-hoc choices, for example in the choice of L^2 norm on $\mathbb{R}^D$, or the sum-over-s procedure that generalises it to a discrepancy between time series. (Which is not a norm, as it is not multiplicative.) Indeed, there are many other natural notions of discrepancy between time series [**?**, **?**, **?**, **?**] that do not fit this framework.

1.3 Contributions

We demonstrate how classical shapelets may be generalised in multiple ways, so as to address the collection of limitations just described.

By treating the objects in continuous time rather than discrete time, we demonstrate how the shapelet method may be extended to irregularly-sampled time series. Furthermore we demonstrate how this allows for the length of each shapelet to be learnt, individually for each shapelet, in a differentiable manner. (Rather than a single hyperparameter shared amongst all shapelets.)

Next, we generalise to allowing the discrepancy between a shapelet and a time series to a learnt pseudometric. This makes our proposed method robust to noise in unrelated channels, and furthermore this introduces a great deal of flexibility into the method.

Finally, we demonstrate how simple regularisation is enough to achieve shapelets that resemble characteristic features of the data, in order to achieve the desired interpretability.

2 Method

We move on to describing our method, which we present in a general form. In the next section we will discuss the specific choices made in our experiments.

2.1 Continuous-time objects

We interpret a time series as a discretised sample from an underlying process, observed only through the time series. Similarly, the shapelet previously constructed may be thought of as a discretisation of some underlying function. The first important step in our procedure is to construct continuous-time approximations to these underlying objects.

Formally speaking, we assume that for $n \in \{1, \ldots, N\}$ indexing different observations, we observe a collection of time series

$$f^n = (f^n_{t_\tau})_{\tau \in \{1,\ldots,T_n\}},$$

where $t_\tau \in \mathbb{R}$ is the observation time of $f^n_{t_\tau} \in \mathbb{R}^D$. Note that this description allows irregularly sampled time series to be treated on the same footing as regularly sampled time series.

Next, interpolate to get a function $\iota(f^n)\colon [0, T_n - 1] \to \mathbb{R}^D$ such that $\iota(f^n)(t_\tau) = f^n_{t_\tau}$ for $\tau \in \{0, \ldots, T_n - 1\}$. There are many possible choices for interpolations, for example splines [**?**], kernel methods [**?**], or Gaussian processes [**?**, **?**].

The shapelets themselves we are free to control, and so for $k \in \{1, \ldots, K\}$ indexing different shapelets, we take each $w^{k,\rho}\colon [0, 1] \to \mathbb{R}^D$ to be some learnt function depending on learnt parameters ρ. For example, this could be an interpolated sequence of learnt points, an expansion in some basis functions, or a neural network. Then for $S_k > 0$ define $w^{k,\rho,S_k}\colon [0, S_k] \to \mathbb{R}^D$ by

$$w^{k,\rho,S_k}(t) = w^{k,\rho}\left(\frac{t}{S_k}\right).$$

Taking the length S_k to be continuous is a necessary prerequisite to training it differentiably. We will discuss the training procedure in a moment.

2.2 Generalised discrepancy

The core of the shapelet method is that the similarity or discrepancy between f^n and $w^{k,\rho,S}$ is important. In general, we approach this by defining a *discrepancy function* between the two, which will typically be learnt, and which we require only to be a pseudometric, rather than a metric. By relaxing to allow pseudometricity, then the procedure becomes robust to noise in unrelated channels, as the learning procedure may learn to ignore extraneous dimensions.

We denote this discrepancy function by π^A_S; it depends upon the length S and a learnt parameter A, consumes two paths in $\mathbb{R}^D$, and returns a real number describing some notion of closeness between them. We are being deliberately vague about the domain of π^A_S, as it will depend on the regularity of ι.

Given some fixed π^A_S, then we define the discrepancy between f^n and $w^{k,\rho,S}$ to be given by

$$\sigma^A_S(f^n, w^{k,\rho,S}) = \min_{s \in [0, T_n - S]} \pi^A_S(\iota(f^n)|_{[s,s+S]}(s + \cdot), w^{k,\rho,S}). \tag{3}$$

The collection of discrepancies $(\sigma^A_S(f^n, w^{1,\rho,S}), \ldots, \sigma^A_S(f^n, w^{K,\rho,S}))$ is now a feature describing f^n, and is invariant to the length T_n. Use of the particular feature $\sigma^A_S(f^n, w^{k,\rho,S})$ corresponds to the importance of the similarity between f^n and $w^{k,\rho,S}$. In this way, the choice of π^A_S gives a great deal of flexibility: not only may it be selected for reasons of classification performance, but it may also be selected to aid interpretability. For example, in many domains it may be of interest to take Fourier transform-based choices of π^A_S.

A simple example, in analogy to the classical shapelet method of equation (2), is to take

$$\pi^A_S(f, w) = \left(\int_0^S \|f(t) - w(t)\|_2^2 \,\mathrm{d}t\right)^{\frac{1}{2}}, \tag{4}$$

which in fact has no A dependence. If ι is taken to be a piecewise constant 'interpolation' then this will exactly correspond to (the square root of) the classical shapelet approach.

This may be generalised by taking our learnt parameter $A \in \mathbb{R}^{D \times D}$, and then letting

$$\pi^A_S(f, w) = \left(\int_0^S \|Af(t) - Aw(t)\|_2^2 \,\mathrm{d}t\right)^{\frac{1}{2}}. \tag{5}$$

We do not put any conditions on A. In particular, as we have allowed pseudometricity, uninformative dimensions may be shrunk to zero.

An explicitly interpretable example is given by a π^A_S that is based on the logsignature transform, which is a transform on paths, known to characterise its input path whilst extracting statistics which

describe how the path controls differential equations [**?**, **?**, **?**, **?**, **?**]. Machine learning extensions are natural; see for example [**?**, **?**, **?**, **?**, **?**]. Here, we define the *p-logsignature distance* between two functions to be

$$\pi_S^A(f, w) = \|A \operatorname{LogSig}^R(f) - A \operatorname{LogSig}^R(w)\|_p,$$

where $A \in \mathbb{R}^{\beta_{D,R} \times \beta_{D,R}}$, LogSig^R is the depth-R logsignature transform of the path, $\| \cdot \|_p$ is the L^p norm on $\mathbb{R}^{\beta_{D,R}}$, and

$$\beta_{D,R} = \sum_{r=1}^{R} \frac{1}{r} \sum_{\rho | r} \mu\left(\frac{r}{\rho}\right) D^\rho$$

is Witt's formula [**?**], and μ is the Möbius function.

In analogy to classical shapelet methods, we call the map

$$f \mapsto (\sigma_S^A(f, w^{1,\rho,S}), \dots, \sigma_S^A(f, w^{K,\rho,S}))$$

the *generalised shapelet transform*.

2.3 Interpretable regularisation

In [**?**], a procedure is described for selecting shapelets as particular small intervals from particular training samples. But doing so is very expensive, requiring $\mathcal{O}(N^2 \cdot \max_n T_n^4)$ work. As such, [**?**] instead show that the discrepancy σ_S of equation (2) is differentiable with respect to $w^{k,\rho,S}$, and so the shapelets may be selected differentiably, as part of an end-to-end optimisation of the final loss function of the model that consumes the shapelets as features.[1]

However, it has been noted in [**?**] that doing so sacrifices much of the interpretability, as the shapelets that are then selected need not look like any small extracts from the training data. They propose to solve this issue via adversarial regularisation.

We instead propose a much simpler method; train differentiably as before, and simply add on $\sigma_S^A(f^n, w^{k,\rho,s})$ as a regularisation term, so that minimising the discrepancy between f^n and $w^{k,\rho,S}$ is also important. One obvious danger of this is that this introduces a bias towards small values of S, as the discrepancy can then easily be made small because $\iota(f^n)$ is locally almost constant. The solution is to regularise S back towards larger values, with another regularisation term, which we denote $\mathcal{R}(S)$; for example we could take $\mathcal{R}(S) = 1/S$.

2.4 Minimisation objective and training procedure

Overall, suppose we have some parametric model F^θ, some loss function $\mathcal{L}$, and some observed time series $f^1, \dots, f^N$ with targets $y_1, \dots, y_N$.

Then letting $\gamma, \delta > 0$ control the amount of each kind of regularisation, we propose to minimise

$$\frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(y_n, F^\theta(\sigma_{S_1}^A(f^n, w^{1,\rho,S_1}), \dots, \sigma_{S_K}^A(f^n, w^{K,\rho,S_K}))) + \gamma \sum_{k=1}^{K} \min_{n \in \{1,\dots,N\}} \sigma_{S_k}^A(f^n, w^{k,\rho,S_k}) + \delta \sum_{k=1}^{K} \mathcal{R}(S_k) \quad (6)$$

over θ, ρ, A and S_k. We allow the length S_k to vary between different shapelets.

Note the choice of minimisation over n in the first regularisation term, rather than a sum over n. A sum over n would ask that every shapelet should look like every training sample. Taking a minimum instead asks only that every shapelet should be similar to some training sample, not all of them.

We will minimise this via standard stochastic gradient descent based techniques. Some thought is necessary to verify that our constructions are in fact differentiable with respect to S_k, and in general this will depend on the regularity of π_S^A, but this is essentially straightforward analysis. Practically speaking, we would like to implement this in an autodifferentiation framework (for example PyTorch [**?**]), and indeed provided π_S^A is constructed in an autodifferentiable manner, then the rest of our constructions may be as well. For completeness we include sample code to do so in Appendix A.

[1] Although they include a 'softmin' procedure which we believe to be unnecessary, as the minimum function is already almost everywhere differentiable.

3 Experiments

Our generalised shapelet transform, contrasted with the classical shapelet transform, has two extra degrees of freedom: the choice of interpolation scheme ι, and the choice of discrepancy function π_S^A. In our experiments, we consider π_S^A given by either of equations (4) or (5). Meanwhile, we take ι to be piecewise linear interpolation, because efficient algorithms for computing the logsignature transform only exist for piecewise linear paths [**?**].

TODO: we need to consider more disrepancy functions than this.

A Code for the shapelet transform

TODO: include!