Learning Time-Series Shapelets

Josif Grabocka, Nicolas Schilling, Martin Wistuba and Lars Schmidt-Thieme

Information Systems and Machine Learning Lab (ISMLL)
University of Hildesheim, Germany

SIGKDD 2014, 26/08/2014

What are Time-Series Shapelets? (I)

- Definition:
  - **Patterns** whose **minimum** distances to time-series yield **discriminative** predictors [Ye and Keogh(2009), Lines et al.(2012)Lines, Davis, Hills, and Bagnall]

- *Problem*:
  1. **Learn** K discriminative shapelets of length L (denoted as $S \in \mathbb{R}^{K \times L}$).
  2. **From** a dataset that has I time-series instances of length M (denoted as $T \in \mathbb{R}^{I \times M}$), where each series is divided into $J = M - L$ sliding window segments.

What are Time-Series Shapelets? (II)

The minimum distances $M \in \mathbb{R}^{I \times K}$ between shapelets and time-series:

$$M_{i,k} = \min_{j=1,\ldots,J} \frac{1}{L} \sum_{l=1}^{L} (T_{i,j+l-1} - S_{k,l})^2 \quad (1)$$

... yield discriminative predictors:

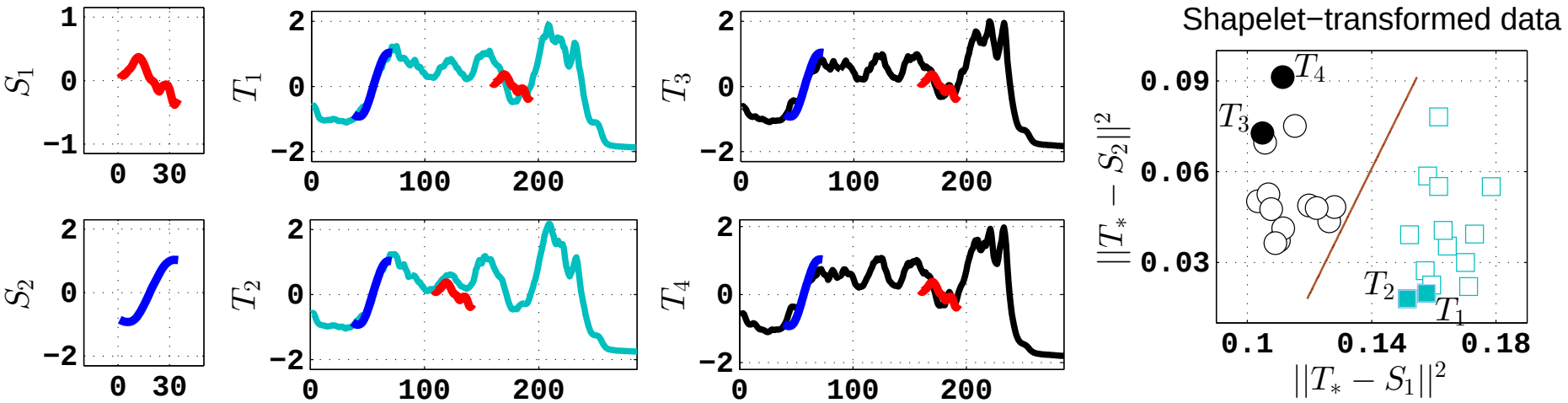

Figure 1: Left: Two shapelets, Middle: Closest Segments, Right: 'Shapelet-Transform'ed data

Related Shapelets Work

All the possible segments (**exhaustively**) of all time-series candidates are potential shapelet candidates:

- Compute the prediction accuracy of minimum distances of each candidate and then **rank** the top-K by prediction accuracy (feature ranking)
- Build a decision tree from the top-K features [Ye and Keogh(2009), Lines et al.(2012)Lines, Davis, Hills, and Bagnall]
- Alternatively, use the new K-dimensional feature representation and use **standard classifiers**
  [Hills et al.(2013)Hills, Lines, Baranauskas, Mapp, and Bagnall].

Exhaustive search is **expensive**, speed-ups were proposed [Mueen et al.(2011)Mueen, Keogh, and Young, Rakthanmanon and Keogh(2013)].

Proposed Method (I)

A linear model of predictors $M \in \mathbb{R}^{I \times K}$ and weights $W \in \mathbb{R}^K$, $W_0 \in \mathbb{R}$: can be used to estimate the target $\hat{Y} \in \mathbb{R}^I$:

$$\hat{Y}_i = W_0 + \sum_{k=1}^{K} M_{i,k} W_k \quad \forall i \in \{1, \ldots, I\} \tag{2}$$

The risk of estimating the true target $Y \in \{-1, +1\}^I$ from approximated target $\hat{Y} \in \mathbb{R}^I$ is the logistic loss $\mathcal{L}(Y, \hat{Y}) \in \mathbb{R}^I$:

$$\mathcal{L}(Y_i, \hat{Y}_i) = -Y_i \ln \sigma(\hat{Y}_i) - (1 - Y_i) \ln \left(1 - \sigma(\hat{Y}_i)\right), \quad \forall i \in \{1, \ldots, I\} \tag{3}$$

Proposed Method (II)

The objective function $\mathcal{F} \in \mathbb{R}$ is a regularized loss function:

$$\underset{S,W}{\text{argmin}}\ \mathcal{F}(S,W) = \underset{S,W}{\text{argmin}} \sum_{i=1}^{I} \mathcal{L}(Y_i, \hat{Y}_i) + \lambda_W ||W||^2 \tag{4}$$

The objective function $\mathcal{F}$ can be decomposed into per-instance objectives $\mathcal{F}_i$:

$$\mathcal{F}_i \quad = \quad \mathcal{L}(Y_i, \hat{Y}_i) + \frac{\lambda_W}{I} \sum_{k=1}^{K} {W_k}^2, \qquad \forall i \in \{1, \ldots, I\} \tag{5}$$

Mission of This Paper: Learn S, W that minimize $\mathcal{F}$.

Differentiable Minimum Function (I)

Approximate the true minimum M with the soft-minimum version $\hat{M}$:

$$M_{i,k} \approx \hat{M}_{i,k} = \frac{\sum_{j=1}^{J} D_{i,k,j}\, e^{\alpha D_{i,k,j}}}{\sum_{j'=1}^{J} e^{\alpha D_{i,k,j'}}},$$
$$\alpha \in (-\infty, 0] \quad \forall i \in \{1, \ldots, I\}, \forall k \in \{1, \ldots, K\} \tag{6}$$

$$D_{i,k,j} := \frac{1}{L} \sum_{l=1}^{L} \left(T_{i,j+l-1} - S_{k,l}\right)^2,$$
$$\forall i \in \{1, \ldots, I\}, \forall k \in \{1, \ldots, K\}, \forall j \in \{1, \ldots, J\} \tag{7}$$

Differentiable Minimum Function (II)

The smooth approximation of the minimum function, allows only the minimum segment to contribute for $\alpha \to -\infty$.

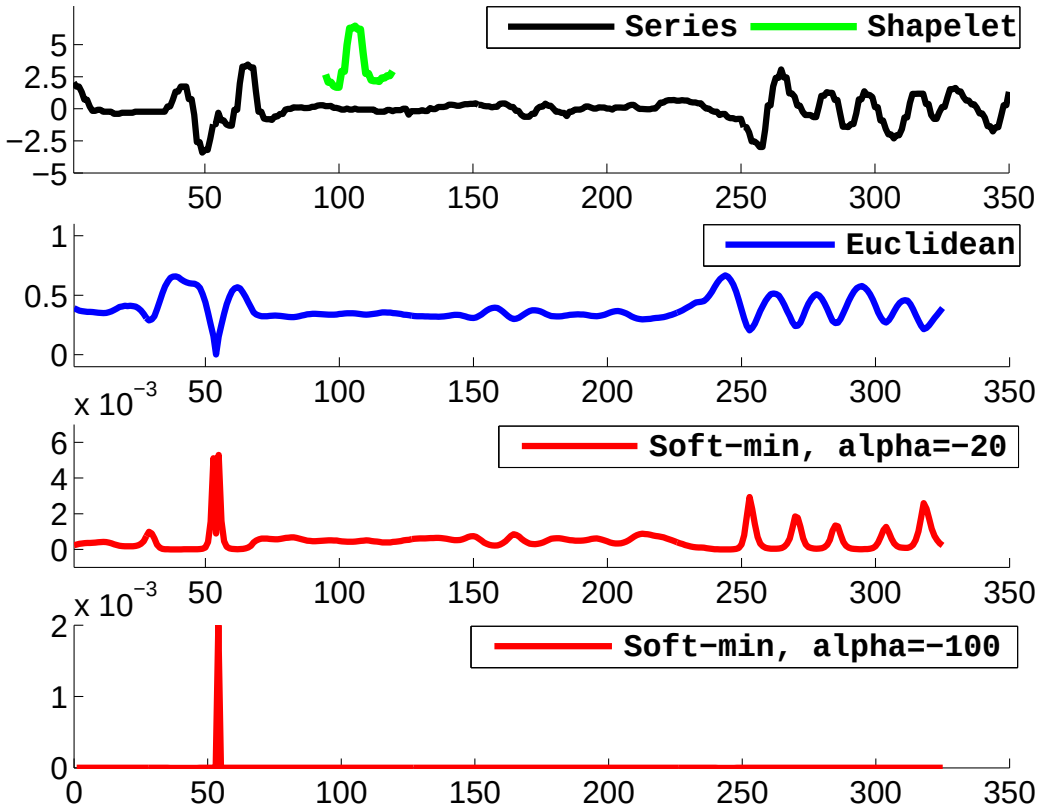

Figure 2: Illustration of the soft minimum between a shapelet (green) and all the segments of a series (black) from the FaceFour dataset

Learning Algorithm (I)

The partial derivative of the per-instance objective function $\mathcal{F}_i$ with respect to the l-th point of the k-th shapelet $S_{k,l}$ is computed using the chain rule of derivation:

$$\frac{\partial \mathcal{F}_i}{\partial S_{k,l}} = \frac{\partial \mathcal{L}(Y_i, \hat{Y}_i)}{\partial \hat{Y}_i} \frac{\partial \hat{Y}_i}{\partial \hat{M}_{i,k}} \sum_{j=1}^{J} \frac{\partial \hat{M}_{i,k}}{\partial D_{i,k,j}} \frac{\partial D_{i,k,j}}{\partial S_{k,l}} \tag{8}$$

$$\frac{\partial \mathcal{F}_i}{\partial W_k} = \frac{\partial \mathcal{L}(Y_i, \hat{Y}_i)}{\partial \hat{Y}_i} \frac{\partial \hat{Y}_i}{\partial \hat{W}_k} + \frac{\partial \mathrm{Reg}(W)}{\partial W_k}, \quad \frac{\partial \mathcal{F}_i}{\partial W_0} = \frac{\partial \mathcal{L}(Y_i, \hat{Y}_i)}{\partial \hat{Y}_i} \tag{9}$$

All the components of the partial derivative are computable as follows:

$$\frac{\partial \mathcal{L}(Y_i, \hat{Y}_i)}{\partial \hat{Y}_i} = -\left(Y_i - \sigma\left(\hat{Y}_i\right)\right), \quad \frac{\partial \hat{Y}_i}{\partial \hat{M}_{i,k}} = W_k, \quad \frac{\partial \hat{Y}_i}{\partial \hat{W}_k} = M_{i,k}, \quad \frac{\partial \mathrm{Reg}(W)}{\partial W_k} = \frac{2\lambda_W}{I} W_k \tag{10}$$

$$\frac{\partial \hat{M}_{i,k}}{\partial D_{i,k,j}} = \frac{e^{\alpha D_{i,k,j}} \left(1 + \alpha \left(D_{i,k,j} - \hat{M}_{i,k}\right)\right)}{\sum_{j'=1}^{J} e^{\alpha D_{i,k,j'}}}, \frac{\partial D_{i,k,j}}{\partial S_{k,l}} = \frac{2}{L}\left(S_{k,l} - T_{i,j+l-1}\right) \tag{11}$$

Learning Algorithm (II)

Require: $T \in \mathbb{R}^{I \times Q}$, Number of Shapelets K, Length of a shapelet L, Regularization λ_W, Learning Rate η, Number of iterations: maxIter

Ensure: Shapelets $S \in \mathbb{R}^{K \times L}$, Classification weights $W \in \mathbb{R}^K$, Bias $W_0 \in \mathbb{R}$

for iteration=$1, \ldots, maxIter$ **do**
  for $i = 1, \ldots, I$ **do**
    $W_0 \leftarrow W_0 - \eta \frac{\partial \mathcal{F}_i}{\partial W_0}$
    for $k = 1, \ldots, K$ **do**
      $W_k \leftarrow W_k - \eta \frac{\partial \mathcal{F}_i}{\partial W_k}$
      for $L = 1, \ldots, L$ **do**
        $S_{k,l} \leftarrow S_{k,l} - \eta \frac{\partial \mathcal{F}_i}{\partial S_{k,l}}$
return S, W, W_0

Illustration of Learning

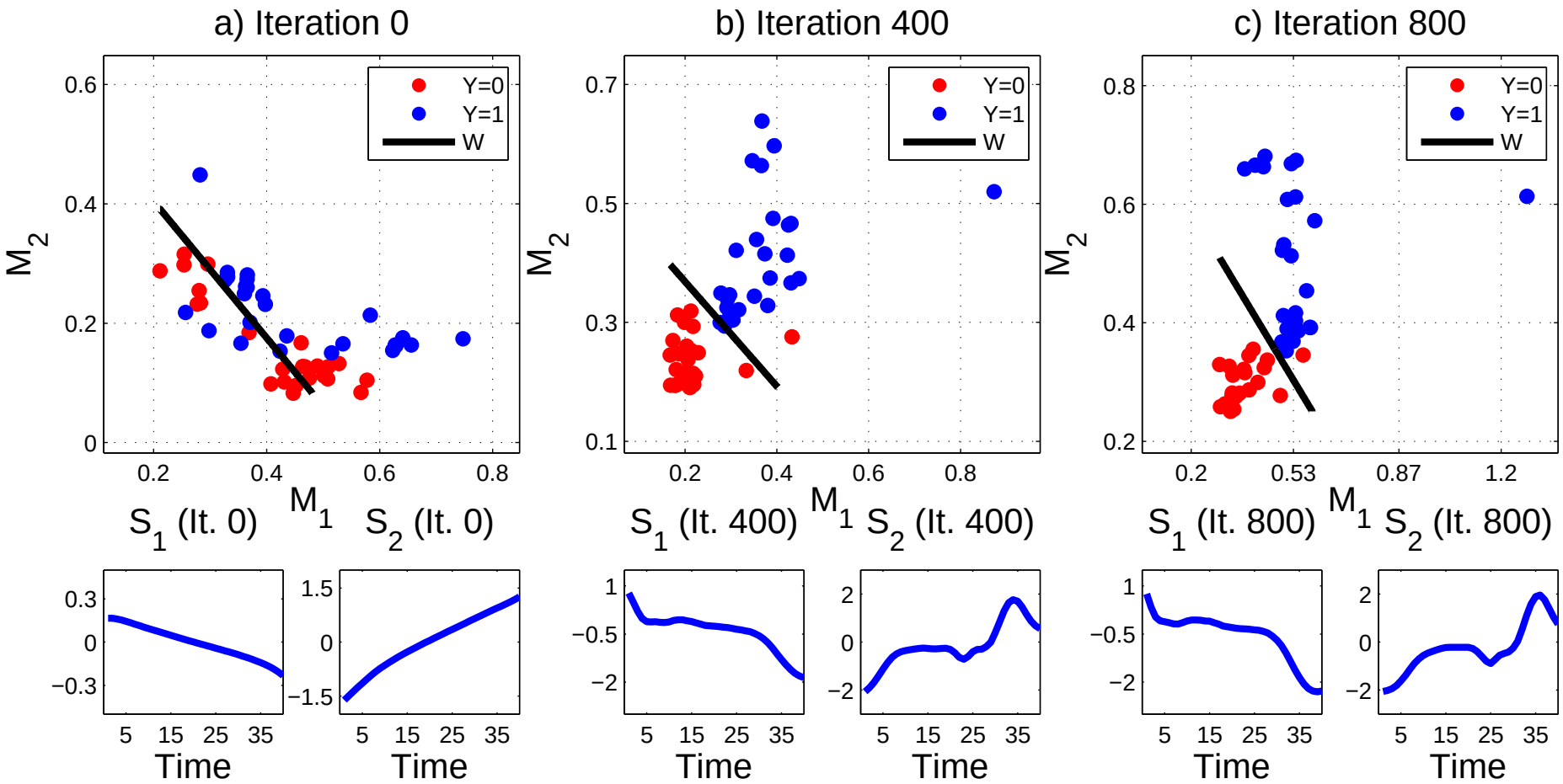

Figure 3: Learning Two Shapelets on the Gun-Point Dataset, ($L = 40, \eta = 0.01, \lambda_W = 0.01, \alpha = -100$)

Advantage over Feature Ranking

1. Discover hidden/latent shapelets
2. Interactions of Variables

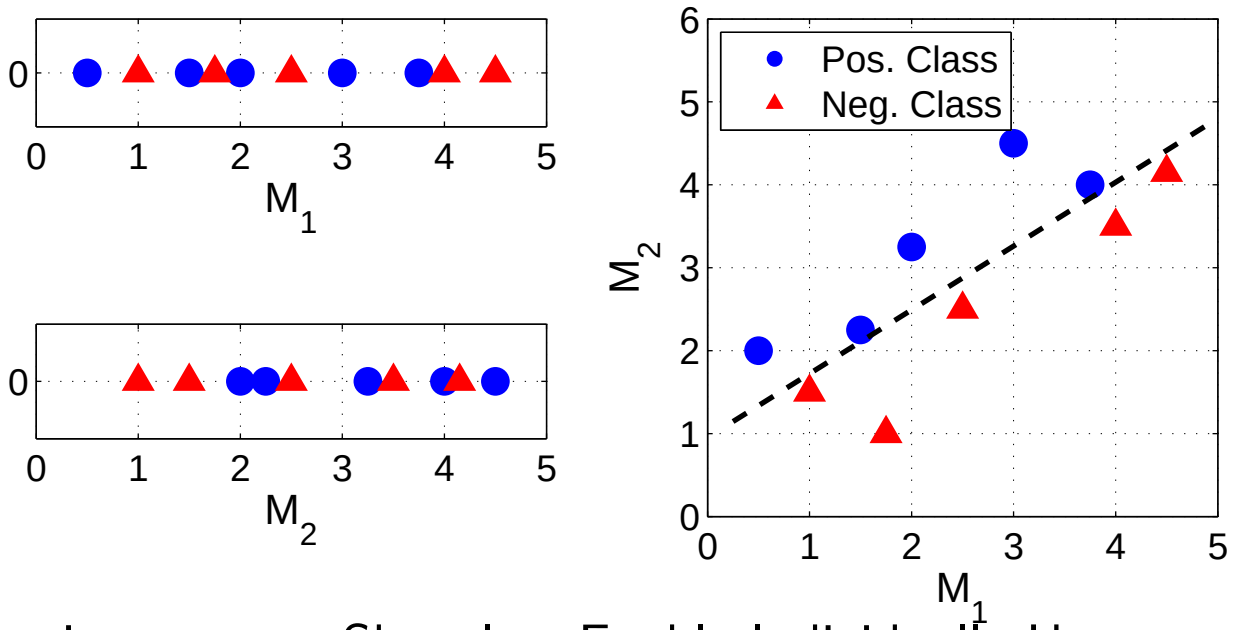

Figure 4: Interactions among Shapelets Enable Individually Unsuccessful Shapelets (left plots) to Excel in Cooperation (right plot)

Experimental Setup

Our method, denoted LS, is compared against 13 baselines using 28 time-series dataset:

- Baselines: **Quality Criteria**: Information gain (IG), Kruskall-Wallis (KW), F-Stats (FT), Mood's Median Criterion (MM); **Using shapelet-transformed data**: Nearest Neighbors (1NN), Naive Bayes (NB), C4.5 tree (C4), Bayesian Networks (BN), Random Forest (RA), Rotation Forest (RO), Support Vector Machines (SV); **Other Related**: Fast Shapelets (FS), **Dynamic Time Warping** (DT).
- Datasets: 28 time-series datasets from the UCR and UEA collections from diverse domains; Provided train/test splits
- Hyper-parameters: are found using grid-search, by testing over a validation split from the training data

Results

	IG	KW	FT	MM	DT	C4	NN	NB	BN	RF	RO	SV	FS	LS
Total Wins	0.0	0.0	1.2	0.3	0.0	0.0	1.5	0.0	1.6	0.2	0.0	4.7	1.3	17.3
LS Wins	28	27	26	26	28	28	23	27	23	26	24	20	26	-
LS Draws	0	0	1	1	0	0	2	0	2	1	1	2	1	-
LS Losses	0	1	1	1	0	0	3	1	3	1	3	6	1	-
Rank Mean	9.8	9.9	9.1	9.5	9.8	10.0	6.2	7.7	5.5	6.3	5.4	4.6	9.2	1.9
Rank C.I.	1.0	1.3	1.3	1.3	1.9	0.8	1.2	1.1	1.2	0.7	0.9	1.2	1.5	0.5
Rank S.D.	2.7	3.4	3.6	3.4	5.0	2.1	3.2	2.9	3.1	2.0	2.4	3.2	4.1	1.4
W.t. p-val.	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	-

Table 1: Comparative Figures of Classification Accuracies over 28 Time-series Datasets

Conclusions

1. Learning shapelets to directly optimize the classification objective improves classification accuracy

2. Supervised Shapelet Learning more accurate than Exhaustive Shapelet Discovery
   - Shapelets are not restricted to series segments
   - Considers interactions among minimum distances features

3. Results against 13 baselines using 28 datasets validate the claim

References

J. Hills, J. Lines, E. Baranauskas, J. Mapp, and A. Bagnall.
Classification of time series by shapelet transformation.
Data Mining and Knowledge Discovery, 2013.

J. Lines, L. Davis, J. Hills, and A. Bagnall.
A shapelet transform for time series classification.
In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

A. Mueen, E. Keogh, and N. Young.
Logical-shapelets: an expressive primitive for time series classification.
In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2011.

T. Rakthanmanon and E. Keogh.
Fast shapelets: A scalable algorithm for discovering time series shapelets.
Proceedings of the 13th SIAM International Conference on Data Mining, 2013.

L. Ye and E. Keogh.
Time series shapelets: a new primitive for data mining.
In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

Back-up Slide: Dependence on Initialization

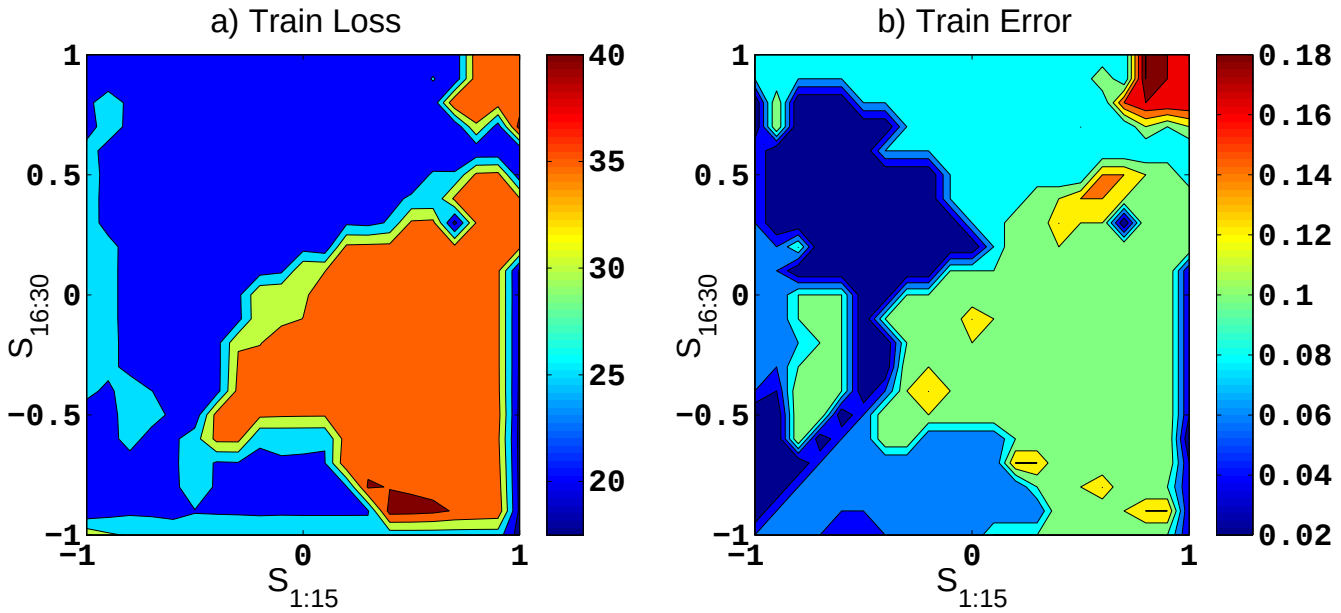

Figure 5: Sensitivity of Shapelet Initialization, Gun-Point dataset, Parameters: $L = 30$, $\eta = 0.01$, $\lambda_W = 0.01$, Iterations$= 3000$, $\alpha = -100$

- We used K-Means centroids as initial shapelets.