A Appendix

A.1 Types of Time Series Anomalies We include six types of time series anomalies, presented in Table 3. Fig. 2 visualizes each type of anomaly and demonstrates how each hyperparameter controls the injection of the anomaly in the time series for a platform anomaly.

A.2 Dataset Details

PhysioNet: The 2017 PhysioNet Challenge dataset [10] comprises a diverse array of real-world 1-lead 300 Hz ECG recordings. We use ECG recordings, each 9 seconds in length with $K = 2700$ time-steps and standardized to have a zero mean and standard deviation of one. Injected anomalies represent 10% of the data. Here, we include three additional controlled TSAD tasks based on the PhysioNet data as shown in Table 4. Table 5 (top) shows the hyperparameter spaces used to train f_{aug} in PhysioNet. The hyperparameters `location` and `length` are normalized by K. Noting that the extremum anomaly always occurs on a single timestamp in the time series, thus `length` is always 1.

MoCap: The CMU Motion Capture (MoCap) dataset[3] includes signal data from various sensors on subjects' bodies as they perform different activities (walking, jumping, or running). As we focus on a univariate setting, only the sensor signal on the left femur is used. Since each data contains the time series data with a short length, we stitch them into a longer one in our experiment. To ensure it is done smoothly, we identified the start and end points of each gait phase for the stitching. We further add random noises to augment the normal samples in the dataset. Each signal is normalized between -1 and 1 and truncated to length $K = 1500$. Constructed anomalies represent 10% of the data. As opposed to PhysioNet, we consider MoCap as a dataset with natural or real anomalies. Therefore, we do not create additional TSAD tasks from this dataset. Table 5 (bottom) shows the hyperparameter spaces used to train f_{aug} in MoCap. The hyperparameters `location` and `length` of platform anomaly are normalized by K. The hyperparameters `location` and `length` of frequency anomaly denote the starting gait phase and the length of gait phases, respectively.

[3]http://mocap.cs.cmu.edu/

Table 3: Types of Time Series Anomalies

Type	Description
Platform	Starting at timestamp `location`, the values of a duration `length` in the time series are **equal to** a constant value `level`.
Mean shift	Starting at timestamp `location`, a constant value `level` is **added to** the values of a duration `length` in the time series.
Amplitude	Starting at timestamp `location`, a constant value `level` is **multiplied with** the values of a duration `length` in the time series.
Trend	Starting at timestamp `location`, a **series of values** *at* **is added to** the duration `length`, where a is the `level` and t is the timestamp in that duration.
Extremum/Spike	A large (either positive or negative) value `level` is **assigned to** a single timestamp `location` in the time series.
Frequency shift	Starting at phase `location`, the frequency of the duration with `length` phases is **increased by** a constant value `level`.

Table 4: Anomaly profile of **additional** TSAD tasks.

Profile	PhysioNet E	PhysioNet F	PhysioNet G
`Type`	Mean shift	Mean shift	Extremum
`Level`	Fixed	Fixed	Fixed
`Location`	Random	Random	Random
`Length`	Random	Fixed	N/A

Table 5: Hyperparameter space for each anomaly type.

PhysioNet			
Anomaly Types	`location`	`length`	`level`
Platform	[100..2000]	[400..600]	$\{0.2k-1 \mid k \in [0..10]\}$
Mean Shift	[100..2000]	[400..600]	$\{0.2k-1 \mid k \in [0..10]\}$
Amplitude	[100..2000]	[400..600]	$\{0.5k+1 \mid k \in [0..10]\}$
Trend	[100..2000]	[400..600]	$\{0.002k-0.01 \mid k \in [0..10]\}$
Extremum/Spike	$\{100k \mid k \in [1..26]\}$	$\{1\}$	$\{3k-15 \mid k \in [0..10]\}$
MoCap			
Anomaly Types	`location`	`length`	`level`
Platform	[200..800]	[100..200]	$\{0.2k-1 \mid k \in [0..10]\}$
Frequency Shift	[1..3]	[1..6]	[1..3]

A.3 Model Configurations See below for details on model configurations (cf. Sec. 4.1).

TSAP configuration: In Table 6, we provide a comprehensive overview of the configuration details for the different components of TSAP.

Table 6: Configuration details for f_{aug} and f_{det} of TSAP

f_{aug} **configuration**	
Encoder$_\phi$	
Conv Layer 1	{In: 1, Out: 64, Kernel: 100, Stride: 4, ReLU, BatchNorm}
Conv Layer 2	{In: 64, Out: 64, Kernel: 100, Stride: 4, ReLU}
Decoder$_\phi$	
TransConv Layer 1	{In: 64, Out: 64, Kernel: 100, Stride: 4, ReLU, BatchNorm}
TransConv Layer 2	{In: 64, Out: 1, Kernel: 100, Stride: 4}
MLP$_\phi$	
MLP Layer 1	{In: 3, Out: 16, ReLU}
MLP Layer 2	{In: 16, Out: dim($\mathcal{Z}_{\text{trn}}$), ReLU}
General Parameters	
Batch Size	64
# Epochs	500
Optimizer	Adam (LR: 0.002)
f_{det} **configuration**	
Encoder$_\theta$	
Conv Layer 1	{In: 1, Out: 32, Kernel: 10, Stride: 2, ReLU, BatchNorm}
Conv Layer 2	{In: 32, Out: 16, Kernel: 10, Dilation: 2, Stride: 2}
Conv Layer 3	{In: 16, Out: 8, Kernel: 10, Dilation: 4, Stride: 4}
Avg Pooling + Flatten	{Kernel: 10, Stride: 3}
Linear Layer	{In: 400, Out: 10}
MLP$_\theta$	
MLP Layer 1	{In: 10, Out: 1}
General Parameters	
Dropout	0.2
Batch Size	64
Warm Start # Epochs f_{det}	3
# Epochs	100
Optimizer for $\mathbf{a}$	Adam (LR: 0.001)
Optimizer for f_{det}	Adam (LR: 0.002)
Mixing Rate (cf. Phase ii)	0.15

Baseline configurations: The details for the baseline configurations are provided in Table 7. Model training was done on a NVIDIA Tesla P100 GPU.

Table 7: Configuration Details for Baselines

Method	Hyperparameter Settings
OC-SVM	We use author-recommended hyperparameters [20].
LOF	We use author-recommended hyperparameters [8].
IF	We use author-recommended hyperparameters [24].
ARIMA	We use AutoARIMA to select hyperparameters [7].
MP	We use author-recommendations to set the window size m [37].
EncDec-LSTM	We downsample our time series to length approx. 200, following [27]. Other hyperparameters follow authors' recommendations.
SR-CNN	We use author-recommended hyperparameters [31].
USAD	We downsample our time series to length approx. 200, following [2]. Other hyperparameters follow authors' recommendations.
NeuTraL-AD	We use author-recommended hyperparameters [29]. We tune augmentation type for each dataset using *labeled* validation data.
TimeGPT	We tune the confidence interval [16] for each dataset using *labeled* validation data.

A.4 Additional Results We present additional results for continuous and discrete augmentation hyperparameter tuning and two additional ablation studies.

A.4.1 Cont. Aug. Hyperparameter Tuning In addition to the results on continuous hyperparameter tuning for PhysioNet A and B (see Fig. 3), we demonstrate TSAP's efficacy in tuning the continuous hyperparameters on five additional TSAD tasks. These include PhysioNet C and D, which feature Trend anomalies (cf. Table 1), as well as PhysioNet E and F, showcasing Mean shift anomalies, and PhysioNet G, illustrating Extremum anomalies (cf. Table 4).

PhysioNet C & D: The tuning process of the continuous hyperparameters for the **Trend anomalies** in PhysioNet C and D is shown in Fig. 6. We observe several initializations for **a** that arrive closely to the true `level` for PhysioNet C, as well as to the true `level` and `length` for PhysioNet D. In turn, those initializations yield a detector f_{det} with high performance on $\mathcal{D}_{\text{val}}$. Note how, for PhysioNet C, $\mathcal{L}_{\text{val}}$ is low across the board. This is likely due to the fact that a Trend anomaly with a subtle slope, has similar characteristics to inliers (see e.g. Fig. 4 bottom left). Yet, TSAP effectively assigns a higher validation loss to the initializations that lead to misaligned cases. This shows the effectiveness of our method even in cases where anomalies are subtle.

PhysioNet E & F: Similarly to Trend anomalies, **Mean shift anomalies** are inherently subtle, especially when the `level` is close to zero. We show in Fig. 7 how TSAP properly tunes the continuous hyperparameters for `level` and `length` in PhysioNet E and F for several initializations.

PhysioNet G: Lastly, Fig. 8 showcases TSAP's ability to tune the `level` of the spike in the **Extremum anomalies** while the location is randomized. Note that Extremum anomalies have no `length` by definition. The ECG recordings in PhysioNet contain many natural spikes. As such, validation loss is low by default. Nonetheless, TSAP successfully tunes two out of four initializations and reflects – though subtly – this difference in the validation loss. This again leads to a well-tuned $f_{\text{det}}(\cdot;\boldsymbol{\theta}^*)$ that performs strongly on $\mathcal{D}_{\text{val}}$.

A.4.2 Discrete Augmentation Hyperparameter Tuning We showcased TSAP's ability to tune the discrete hyperparameter, anomaly type, in Fig. 4 for controlled and natural TSAD tasks. Given the direct applicability and significance of discrete hyperparameter tuning in real-world contexts, we present extended results for discrete hyperparameter tuning.

MoCap B: Fig. 9 shows the **discrete hyperparameter tuning for the unknown anomaly type** in MoCap B. TSAP was initialized twice: first with f_{aug} pre-trained for injecting Frequency shift anomalies, and second with f_{aug} pre-trained for injecting Platform anomalies. The validation loss (left) indicates a strong alignment between $\mathcal{D}_{\text{trn}} \cup \mathcal{D}_{\text{aug}}$ and the unlabeled $\mathcal{D}_{\text{val}}$ when TSAP is initialized with Frequency shift anomalies. This is also reflected in f_{det}'s performance on $\mathcal{D}_{\text{val}}$ (center). Visually, we can indeed confirm that Frequency shift anomalies (right – red) appear to be more similar to the running pattern (right – black) as opposed to Platform anomalies (right – purple).

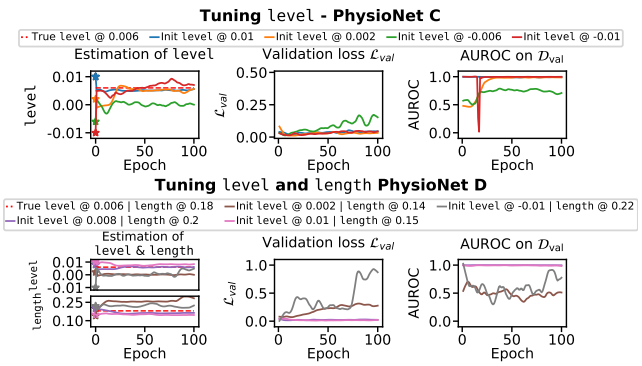

Figure 6: **Tuning the continuous augmentation hyperparameter(s) with TSAP for Trend anomalies**. **Top:** Given Trend anomalies at true `level` (red dashed line), various initializations converge near the true value (left), following the minimized values of val. loss (center), and leading to high detection AUROC performance (right). **Bottom:** Multiple continuous hyperparameters, here both `level` and `length` are tuned to near true values (left), guided by minimizing the val. loss (center), achieving high AUROC (right).

A.4.3 Additional Ablation Studies Following Sec. 4.4, we present additional ablation studies for the second-order optimization within TSAP and the normalization of the embeddings $\{\mathcal{Z}_{\text{trn}}, \mathcal{Z}_{\text{aug}}, \mathcal{Z}_{\text{val}}\}$ obtained through $f_{\text{det}}^{\text{enc}}$.

Second-order Opt. Ablation: Fig. 10 (top) shows the `level`-estimation and performance of f_{det} on PhysioNet C when the second-order optimization is disabled. Note how the estimation process becomes highly unstable when second-order optimization is disabled. In turn, performance of f_{det} on $\mathcal{D}_{\text{val}}$ suffers severely.

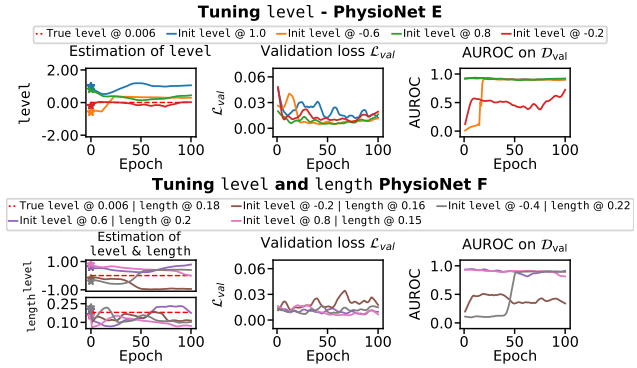

Figure 7: **Tuning the continuous augmentation hyperparameter(s) with TSAP for Mean shift anomalies**. **Top:** Given Mean shift anomalies at true `level` (red dashed line), various initializations converge near the true value (left), following the minimized values of val. loss (center), and leading to high detection AUROC performance (right). **Bottom:** Multiple continuous hyperparameters, both `level` and `length` are accurately tuned to near true values (left), as guided by minimizing the val. loss (center), achieving high AUROC (right). Mean shift anomalies have low val. loss by default due to the subtle nature of Mean shift anomalies.

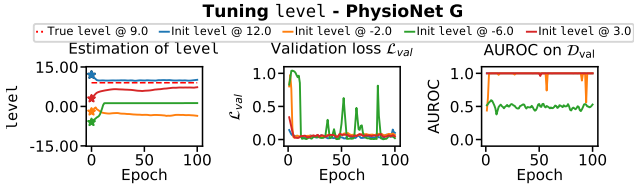

Figure 8: **Tuning the continuous augmentation hyperparameter(s) with TSAP for Extremum anomalies**. Given Extremum anomalies at true `level` (red dashed line), various initializations converge near the true value (left), following the minimized values of val. loss (center), and leading to high detection AUROC performance (right). Note how Extremum anomalies have low val. loss by default due to the natural presence of spikes in ECG data.

Embedding Normalization Ablation: We show the `level`-estimation and performance of f_{det} on PhysioNet C when embedding normalization is disabled in Fig. 10 (bottom). While a initialized at 0.01 eventually leads to the correct `level`, the estimation process is highly volatile compared to when normalization is enabled as shown in Fig. 6 (top).

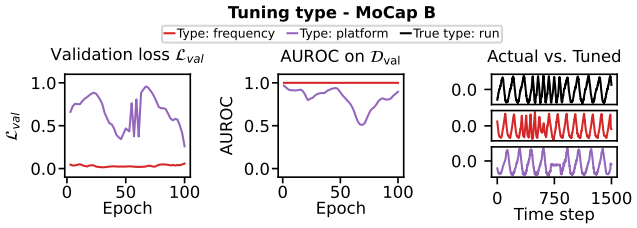

Figure 9: **Tuning discrete hyperparameter (anomaly type) with TSAP**. For Run anomalies in MoCap B with unknown type (black), val. loss favors Frequency shift (red) that leads to high AUROC (center) and mimics well true anomaly (right), and effectively rejects inferior type Platform (purple).

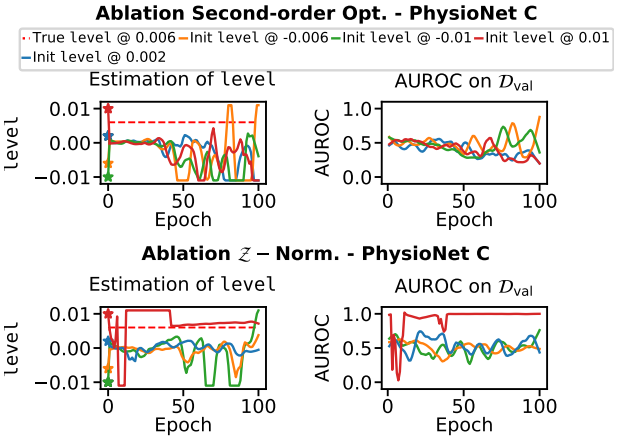

Figure 10: **Overview of additional ablation studies. Top:** We **disable second-order optimization** in TSAP, leading to a highly unstable estimation process of a (left) and poor performance of f_{det} (right). **Bottom:** We **disable normalization** of the embeddings in TSAP. Estimation of a (left) is volatile and does not converge well, in turn, performance of f_{det} on $\mathcal{D}_{\text{val}}$ is poor in most cases (right).