National University of Computer and Emerging Sciences

Data Mining Project
Anomaly Detection

Due Date: 12-05-2024

Instructions:

- Submit your file under the naming **convention iXXXXXX_Section.ipynb (E.g. i210328_A.ipynb) & iXXXXXX_Section.pdf (E.g. i210328_A.pdf).**
- **Do not zip your files.** A 5% penalty will be applied if this is not adhered to.
- Submit the assignment on time. Late submissions will not be considered.
- The deadline will not be extended.
- **Note:** This project is a continuation of Assignment #3. It is highly recommended to read the paper thoroughly before implementing your solution: https://drive.google.com/file/d/1y5ihBtCcKU4-nBHMQEviUU6luq45kaRD/view
- Lastly, have fun :). The goal of the project is to use your critical thinking skills and clustering algorithm knowledge in order to detect anomalies within datasets. Do take your time with understanding the more recent methods of Anomaly detection

---

Problem statement

Given a multivariate time series dataset **X ∈ R^(N×d)**, you need to aim to identify anomalous points without utilizing any known ground truth labels. To detect anomalies, reconstruction errors or equivalent methods are calculated using the input data and should be used to improve your anomaly detection models. Your end goal is to achieve an anomaly detection model with a high F1 Score.

Datasets: https://github.com/elisejiuqizhang/TS-AD-Datasets

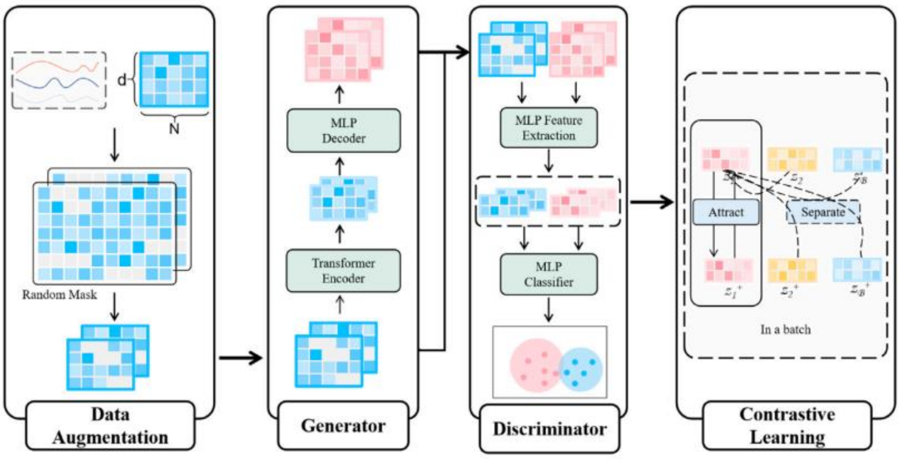
TASK 1:

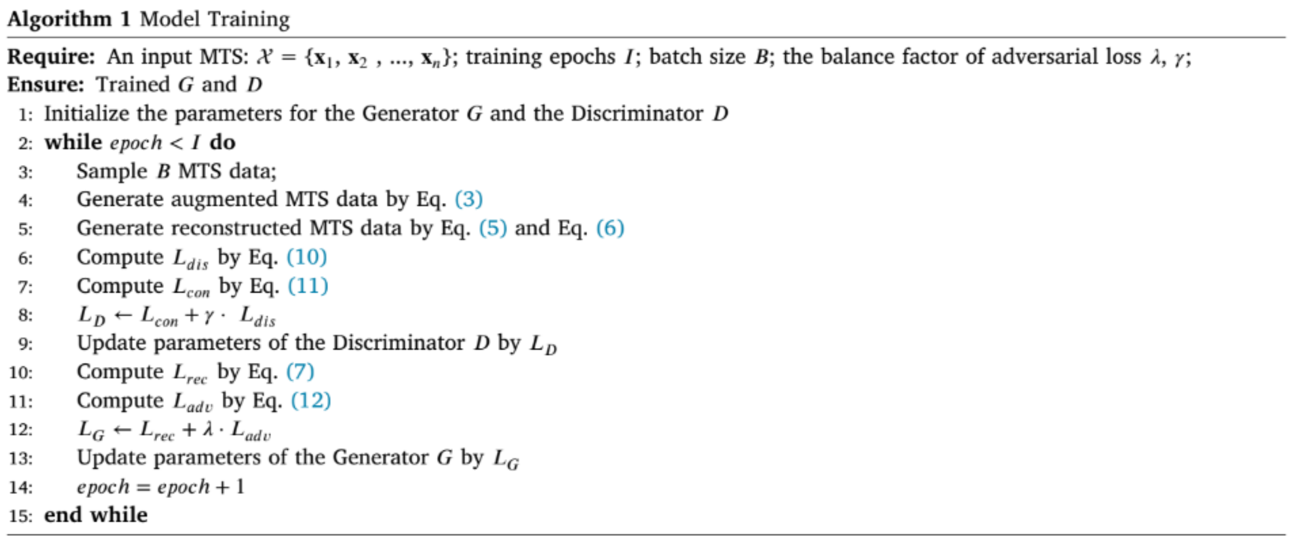
Data Analysis: Load the data into a pandas data frame (or equivalent) and perform Exploratory Data Analysis (EDA) using Matplotlib or Seaborn. Make sure your visualizations are meaningful and visually allow you to interpret where the existing anomalies lie in the dataset.

TASK 2:

Implement the following pipeline for your anomaly detection model:

- **Data Augmentation:** Leverage the properties of MTS (Multivariate Timeseries) and expand the unexplored input space using a novel method called random mask, which follows a geometric distribution.
- **Generator:** This module learns the underlying distribution of normal patterns in MTS and reconstructs it precisely using a Transformer-based autoencoder.
- **Discriminator**: This module imposes constraints on the reconstructions within the GAN framework to better capture normal patterns in MTS.
- **Contrastive Learning:** This module imposes contrastive constraints on representations of MTS to enhance the generalization capability of the model, and facilitates joint training of the Discriminator.





Algorithm 1 Model Training

```
Require: An input MTS: $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$; training epochs $I$; batch size $B$; the balance factor of adversarial loss $\lambda$, $\gamma$;
Ensure: Trained $G$ and $D$
Initialize the parameters for the Generator $G$ and the Discriminator $D$
while $epoch < I$ do
    Sample $B$ MTS data;
    Generate augmented MTS data by Eq. (3)
    Generate reconstructed MTS data by Eq. (5) and Eq. (6)
    Compute $L_{dis}$ by Eq. (10)
    Compute $L_{con}$ by Eq. (11)
    $L_D \leftarrow L_{con} + \gamma \cdot L_{dis}$
    Update parameters of the Discriminator $D$ by $L_D$
    Compute $L_{rec}$ by Eq. (7)
    Compute $L_{adv}$ by Eq. (12)
    $L_G \leftarrow L_{rec} + \lambda \cdot L_{adv}$
    Update parameters of the Generator $G$ by $L_G$
    $epoch = epoch + 1$
end while
```

TASK 3:

Implement further anomaly detection models based on the following concepts:

- Principal Component Analysis (PCA)
- Graph Deviation Network (GDN)
- Anomaly Transformer
- One-Class SVM
- Isolation Forest
- Local Outlier Factor
- DBSCAN
- **Bonus:** Dynamic Time Warping (DTW)
- **Bonus:** Seasonal Hybrid Extreme Studentized Deviate (S-H-ESD):

TASK 4:

Perform Empirical Analysis between all of these methods. Your analysis should include the following details visually: F1 Score, Time, AUC Score. An example of empirical analysis is shown below:

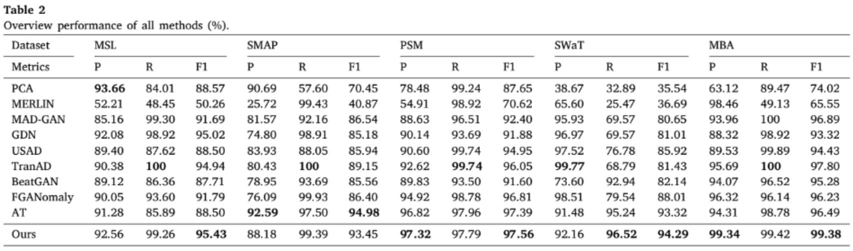


Table 2
Overview performance of all methods (%).

Dataset	MSL			SMAP			PSM			SWaT			MBA		
Metrics	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
PCA	**93.66**	84.01	88.57	90.69	57.60	70.45	78.48	99.24	87.65	38.67	32.89	35.54	63.12	89.47	74.02
MERLIN	52.21	48.45	50.26	25.72	99.43	40.87	54.91	98.92	70.62	65.60	25.47	36.69	98.46	49.13	65.55
MAD-GAN	85.16	99.30	91.69	81.57	92.16	86.54	88.63	96.51	92.40	95.93	69.57	80.65	93.96	100	96.89
GDN	92.08	98.92	95.02	74.80	98.91	85.18	90.14	93.69	91.88	96.97	69.57	81.01	88.32	98.92	93.32
USAD	89.40	87.62	88.50	83.93	88.05	85.94	90.60	99.74	94.95	97.52	76.78	85.92	89.53	99.89	94.43
TranAD	90.38	**100**	94.94	80.43	**100**	89.15	92.62	**99.74**	96.05	**99.77**	68.79	81.43	95.69	**100**	97.80
BeatGAN	89.12	86.36	87.71	78.95	93.69	85.56	89.83	93.50	91.60	73.60	92.94	82.14	94.07	96.52	95.28
FGANomaly	90.05	93.60	91.79	76.09	99.93	86.40	94.92	98.78	96.81	98.51	79.54	88.01	96.32	96.14	96.23
AT	91.28	85.89	88.50	**92.59**	97.50	**94.98**	96.82	97.96	97.39	91.48	95.24	93.32	94.31	98.78	96.49
Ours	92.56	99.26	**95.43**	88.18	99.39	93.45	**97.32**	97.79	**97.56**	92.16	**96.52**	**94.29**	**99.34**	99.42	**99.38**

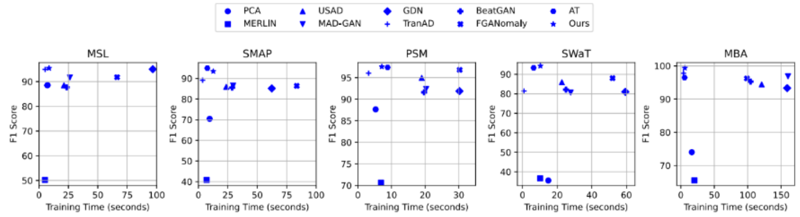
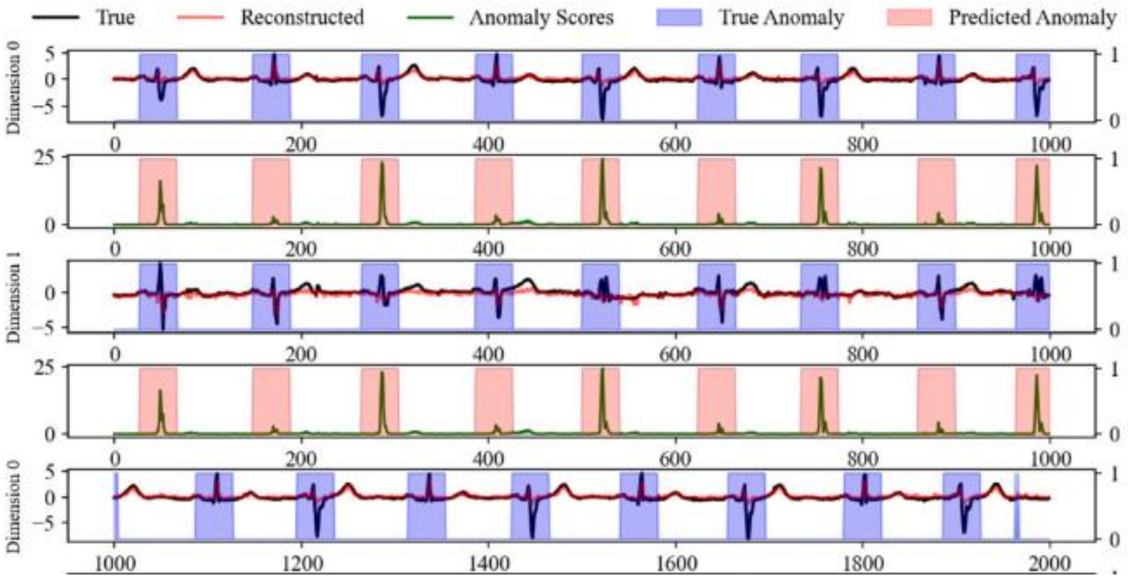


Fig. 4. Performance comparison of computational efficiency and F1 scores.



Bonus: Showcase the distributions between normal data, abnormal data, reconstructed normal data and reconstructed abnormal data. Below is an image to help clarify further:

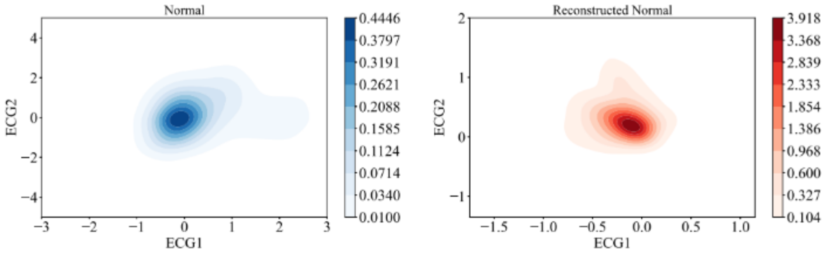


Fig. 8. Distribution of the normal data. The figure on the left illustrates the distribution of the original normal data, while the right illustrates the distribution of the reconstructed normal data.

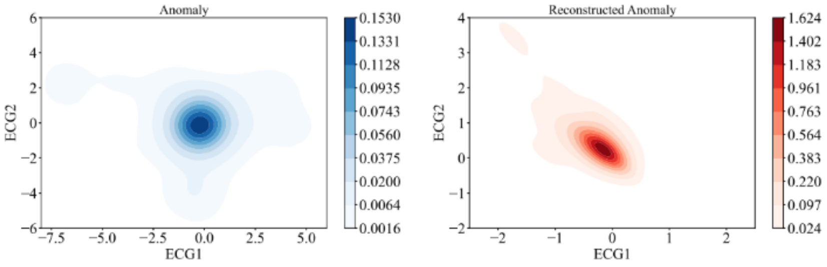


Fig. 9. The distribution of the abnormal data. The figure on the left illustrates the distribution of the original anomaly data, while the right illustrates the distribution of the reconstructed anomaly data.

Submission Guideline

Submit your **.ipynb notebook file** as well as a **.pdf report**. The report should explain implementation of all of the tasks stated above along with **Python code and screenshots**. Justification for techniques needs to be explicitly mentioned in the relevant part of your report.

Best of Luck!