>>> **Privacy-Preserving Data Generation:**
>>> Towards Generating Privacy-Preserving, Synthetic and Useful Time Series ECG Data for Anomaly Detection

KTH x RISE
Sijun John Tu
March 13, 2024

>>> Outline

1. Project introduction
2. Dataset: MITBIH ECG data
3. Privacy-preserving Time Series Data Generation
4. Results
5. Summary
6. References

>>> Outline

1. Project introduction
2. Dataset: MITBIH ECG data
3. Privacy-preserving Time Series Data Generation
4. Results
5. Summary
6. References

>>> Machine Learning Pipeline

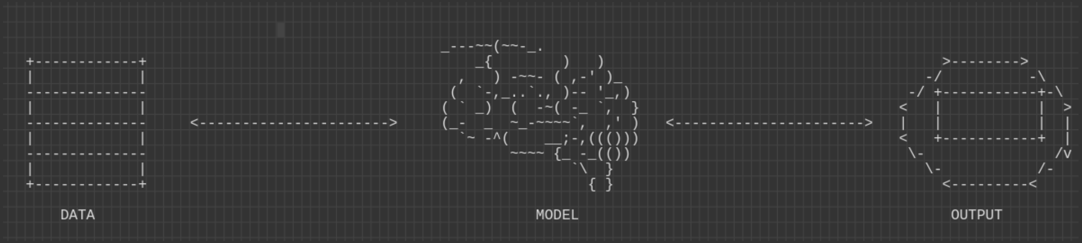


Figure: High-level machine learning pipeline

>>> Anomaly detection using privacy-preserving, synthetic time series data

$ Problem

- ML models are very **data hungry**.
- In many cases sharing data comes with **privacy risks**.

>>> Anomaly detection using privacy-preserving, synthetic time series data

$ Problem
- ML models are very data hungry.
- In many cases sharing data comes with privacy risks.

$ Solution:
- Promising solution: synthetic data with privacy guarantees!
- Synthetic data with differential private (DP) guarantees is a promising solution to ensure privacy independent of downstream task.

>>> Anomaly detection using privacy-preserving, synthetic time series data

$ Problem

- ML models are very `data hungry`.
- In many cases sharing data comes with `privacy risks`.

$ Solution:

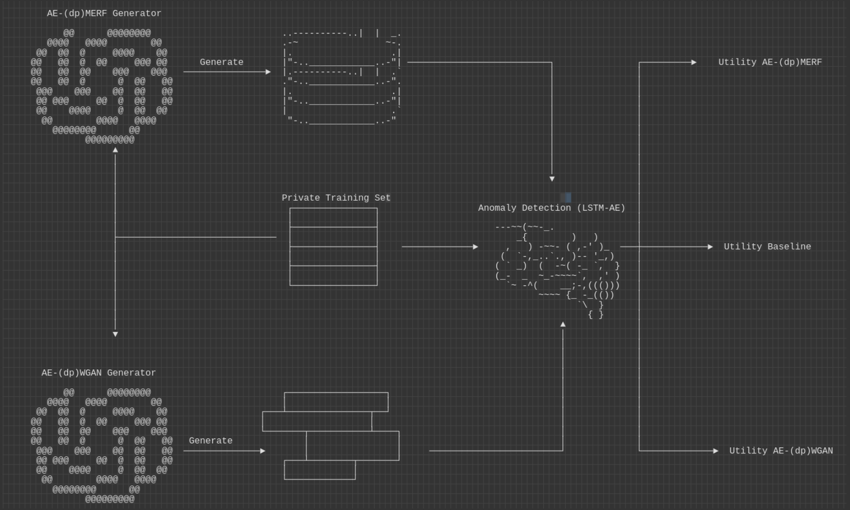
- Promising solution: `synthetic data` with privacy guarantees!
- Synthetic data with `differential private` (DP) guarantees is a promising solution to ensure privacy independent of downstream task.

$ BUT:

- `Privacy-Utility-Tradeoff`: Commonly, a gain in privacy results in a loss of utility.
- For `anomaly detection` this might not be the case (?).

>>> Structure

Figure: Structure of Experiment pipeline



>>> Structure

1. Train **baseline model** for anomaly detection only on **regular, private heartbeat data** using an LSTM-Autoencoder.

>>> Structure

1. Train **baseline model** for anomaly detection only on **regular, private heartbeat data** using an LSTM-Autoencoder.
2. **Generate regular heartbeat** data using two approaches:
   - AE-(dp)MERF
   - AE-(dp)WGAN

>>> Structure

1. Train baseline model for anomaly detection only on regular, private heartbeat data using an LSTM-Autoencoder.
2. Generate regular heartbeat data using two approaches:
   - AE-(dp)MERF
   - AE-(dp)WGAN
3. Train LSTM-Autoencoder for anomaly detection on synthetic data and test on real.
4. Assess utility by measuring performance for anomaly detection (Accuracy, precision, recall, F1).

>>> Structure

1. Train **baseline model** for anomaly detection only on **regular, private heartbeat data** using an LSTM-Autoencoder.
2. **Generate regular heartbeat** data using two approaches:
   - AE-(dp)MERF
   - AE-(dp)WGAN
3. Train LSTM-Autoencoder for **anomaly detection on synthetic data** and **test on real**.
4. Assess **utility** by measuring performance for anomaly detection (Accuracy, precision, recall, F1).
5. **Contaminate** training data with anomalous heartbeats and repeat.

>>> Outline

>>> Heartbeat Arrhythmia

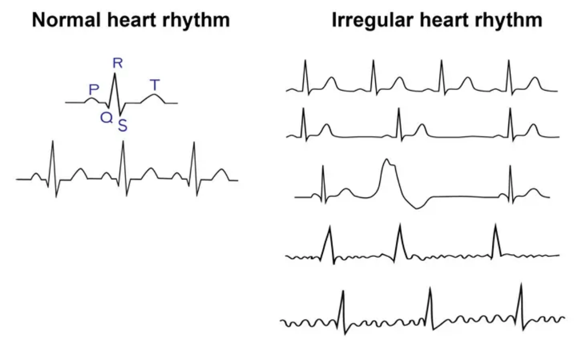


Figure: Different heartbeat arrhythmias [1]

[1] source: https://www.parkwayshenton.com.sg/health-plus/article/arrhythmia-guide

>>> Arrhythmia Detection as an Anomaly Detection Problem

We treat the problem of detecting anomalous heartbeats as an anomaly detection problem from machine learning based on the `reconstruction error`:

>>> Arrhythmia Detection as an Anomaly Detection Problem

We treat the problem of detecting anomalous heartbeats as an anomaly detection problem from machine learning based on the `reconstruction error`:

- $ We train a model `on regular heartbeats` that is able to reconstruct that regular heartbeat.

>>> Arrhythmia Detection as an Anomaly Detection Problem

We treat the problem of detecting anomalous heartbeats as an anomaly detection problem from machine learning based on the `reconstruction error`:

- $ We train a model `on regular heartbeats` that is able to reconstruct that regular heartbeat.
- $ Given an anomalous heartbeat the model should give `higher reconstruction error`.

>>> Arrhythmia Detection as an Anomaly Detection Problem

We treat the problem of detecting anomalous heartbeats as an anomaly detection problem from machine learning based on the `reconstruction error`:

- $ We train a model `on regular heartbeats` that is able to reconstruct that regular heartbeat.
- $ Given an anomalous heartbeat the model should give `higher reconstruction error`.
- $ Based on an optimal `threshold` for that error we classify this heartbeat as either regular or anomalous.

>>> Arrhythmia Detection as an Anomaly Detection Problem

We treat the problem of detecting anomalous heartbeats as an anomaly detection problem from machine learning based on the `reconstruction error`:

- $ We train a model `on regular heartbeats` that is able to reconstruct that regular heartbeat.
- $ Given an anomalous heartbeat the model should give `higher reconstruction error`.
- $ Based on an optimal `threshold` for that error we classify this heartbeat as either regular or anomalous.

Two reasons for this semi-supervised approach: high class imbalancy and no need for labelling.

>>> Baseline Model

Model is a LSTM-AE that is `trained only on regular, private samples` with the goal to reconstruct normal samples. The classification is made based on the reconstruction error.

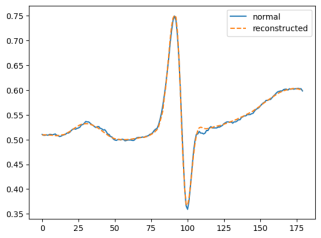


Figure: reconstruction on normal sample

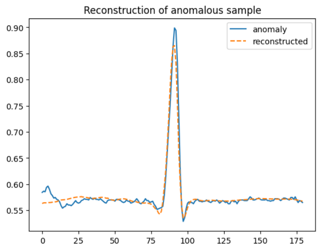


Figure: reconstruction on anomalous sample

>>> Outline

1. Project introduction
2. Dataset: MITBIH ECG data
3. Privacy-preserving Time Series Data Generation
4. Results
5. Summary
6. References

>>> Review: Differential Privacy

Idea. Hide the influence of one particular sample on the output of the model by adding randomness.

>>> Review: Differential Privacy

Idea. Hide the influence of one particular sample on the output of the model by adding randomness.

Definition (Differential Privacy)

A randomised algorithm $\mathcal{M}$ is (ϵ, δ)- differentially private if for all sets of outcomes $S \subset ran\mathcal{M}$ and for all databases x, y, such that they **only differ in one element**, we have

$$\mathbb{P}(\mathcal{M}(x) \in S) \leq e^{\epsilon} \cdot \mathbb{P}(\mathcal{M}(y) \in S) + \delta \quad , \tag{1}$$

where the probability is taken over the randomness of $\mathcal{M}$.

>>> Review: Differential Privacy

Definition (Differential Privacy)

A randomised algorithm $\mathcal{M}$ is (ϵ, δ)- differentially private if for all sets of outcomes $S \subset ran\mathcal{M}$ and for all databases x, y, such that they **only differ in one element**, we have

$$\mathbb{P}(\mathcal{M}(x) \in S) \leq e^{\epsilon} \cdot \mathbb{P}(\mathcal{M}(y) \in S) + \delta \quad , \tag{1}$$

where the probability is taken over the randomness of $\mathcal{M}$.

Informally. Replacing one record in the data will not change the outcome of algorithm $\mathcal{M}$ *too much* (specified via privacy budget ϵ). The lower ϵ the stricter the privacy guarantees.

>>> Examples of DP mechanism

$ Gaussian mechanism

- Add **Gaussian noise** to output of some function.
- For a given function $f : \mathbb{N}^{|\mathcal{X}|} \longrightarrow \mathbb{R}^d$, privacy parameters $\epsilon \in (0, 1)$ and $\delta > 0$ define the gaussian mechanism $F(x)$ as follows:

$$F(x) = f(x) + \xi \quad , \tag{2}$$

where $\xi \sim \mathcal{N}(0, \sigma^2 I)$ and $\sigma \geq \frac{2\Delta f}{\epsilon} \ln(\frac{1.25}{\delta})$ to satisfy DP.

$ DP-SGD

- DP version for training neural networks
- **Add noise to gradients** while training

>>> AE-(dp)MERF: Architecture

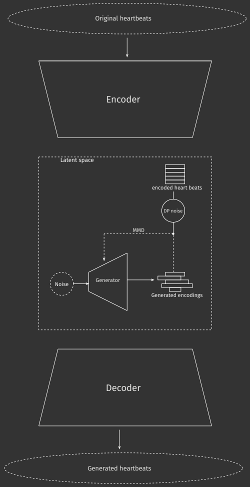


Figure: AE-(dp)MERF architecture

1. Encode dataset X: $X^{enc} = Enc(X)$
2. Generate encodings by sampling from Gaussian noise: $\widetilde{X}^{enc} = Gen(z)$
3. Train generator via loss maximum mean discrepancy (MMD) loss: $MMD(X^{enc}, \widetilde{X}^{enc}) = ||\frac{1}{m}\sum_{i=1}^{m}\hat{\Phi}(x_i^{enc}) - \frac{1}{m}\sum_{j=1}^{m}\hat{\Phi}(\tilde{x}_j^{enc})||_{\mathcal{H}}^2$ where $\hat{\Phi}(x) \in \mathbb{R}^D$ and $\hat{\Phi}_j(x) = \sqrt{\frac{2}{D}} cos(\omega_j^T x)$.
4. Decode generated encodings: $\widetilde{X} = Dec(\widetilde{X}^{enc})$

>>> AE-(dp)MERF: Differential Privacy

We are making AE-MERF differentially private by adding noise to the loss function:

$$MMD(X^{enc}, \widetilde{X}^{enc}) = ||\frac{1}{m}\sum_{i=1}^{m}\hat{\Phi}(x_i^{enc}) - \frac{1}{m}\sum_{j=1}^{m}\hat{\Phi}(\tilde{x}_j^{enc})||_{\mathcal{H}}^2 \quad . \tag{3}$$

>>> AE-(dp)MERF: Differential Privacy

We are making AE-MERF differentially private by adding noise to the loss function:

$$MMD(X^{enc}, \widetilde{X}^{enc}) = ||\frac{1}{m}\sum_{i=1}^{m}\hat{\Phi}(x_i^{enc}) - \frac{1}{m}\sum_{j=1}^{m}\hat{\Phi}(\tilde{x}_j^{enc})||^2_{\mathcal{H}} \quad . \tag{3}$$

>>> AE-(dp)MERF: Differential Privacy

We are making AE-MERF differentially private by adding noise to the loss function:

$$MMD(X^{enc}, \widetilde{X}^{enc}) = ||\frac{1}{m}\sum_{i=1}^{m}\hat{\Phi}(x_i^{enc}) - \frac{1}{m}\sum_{j=1}^{m}\hat{\Phi}(\tilde{x}_j^{enc})||_{\mathcal{H}}^2 \quad . \tag{3}$$

We only need to add noise to the term that “sees” the original data:

>>> AE-(dp)MERF: Differential Privacy

We are making AE-MERF differentially private by adding noise to the loss function:

$$MMD(X^{enc}, \widetilde{X}^{enc}) = ||\frac{1}{m}\sum_{i=1}^{m} \hat{\Phi}(x_i^{enc}) - \frac{1}{m}\sum_{j=1}^{m} \hat{\Phi}(\tilde{x}_j^{enc})||_{\mathcal{H}}^2 \quad . \tag{3}$$

We only need to add noise to the term that "sees" the original data:

$$MMD(X^{enc}, \widetilde{X}^{enc}) = ||\frac{1}{m}\sum_{i=1}^{m} \hat{\Phi}(x_i^{enc}) + \xi - \frac{1}{m}\sum_{j=1}^{m} \hat{\Phi}(\tilde{x}_j^{enc})||_{\mathcal{H}}^2 \quad , \tag{4}$$

where $\xi \sim \mathcal{N}(0, \sigma^2 I)$

>>> AE-(dp)WGAN: Architecture

1. Encode dataset X: $X^{enc} = Enc(X)$
2. Generate encodings by sampling from Gaussian noise: $\widetilde{X}^{enc} = Gen(z)$
3. Discriminator learns to distinguish between real and fake samples.
4. Train Discriminator and Generator jointly.
5. Decode generated encodings: $\widetilde{X} = Dec(\widetilde{X}^{enc})$

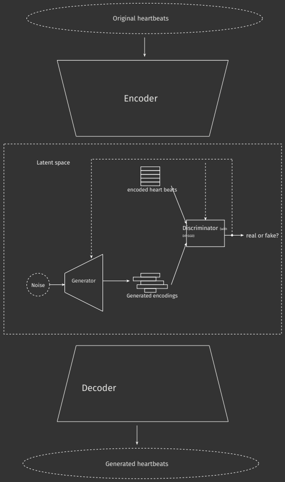


Figure: Architecture of AE-(dp)WGAN

>>> AE-(dp)WGAN: Differential Privacy

1. Encode dataset X: $X^{enc} = Enc(X)$
2. Generate encodings by sampling from Gaussian noise: $\widetilde{X}^{enc} = Gen(z)$
3. Discriminator learns to distinguish between real and fake samples.
4. **Train Discriminator and Generator jointly with DP-SGD.**
5. Decode generated encodings: $\widetilde{X} = Dec(\widetilde{X}^{enc})$

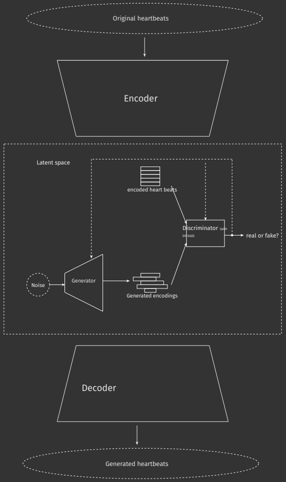


Figure: Architecture of AE-(dp)WGAN

>>> Outline

1. Project introduction
2. Dataset: MITBIH ECG data
3. Privacy-preserving Time Series Data Generation
4. Results
5. Summary
6. References

>>> AE-(dp)MERF

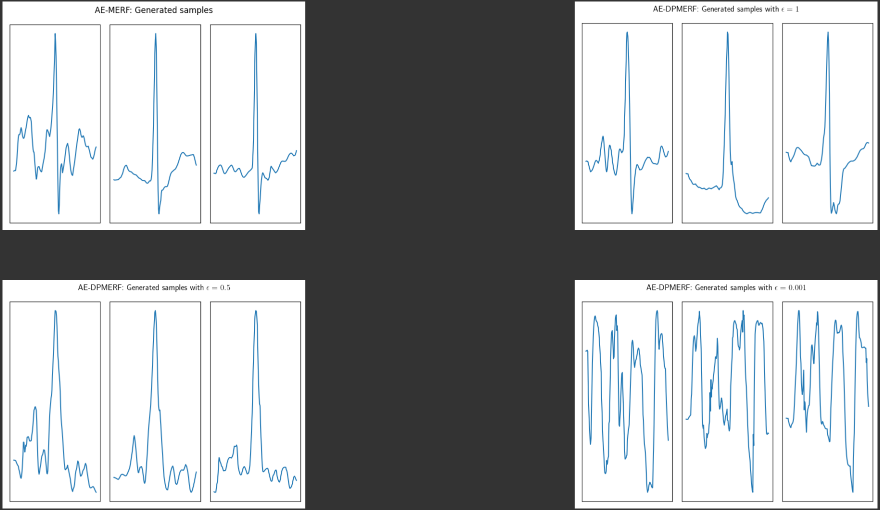


Figure: AE-(dp)MERF generated samples

>>> AE-(dp)MERF: Utility

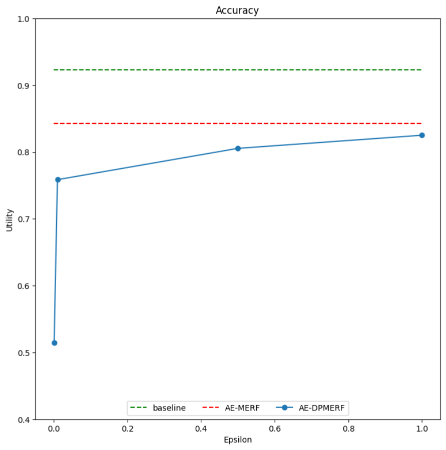


Figure: Results of AE-(DP)MERF with different privacy budgets (lower epsilon means higher privacy)

>>> AE-(dp)WGAN: Utility

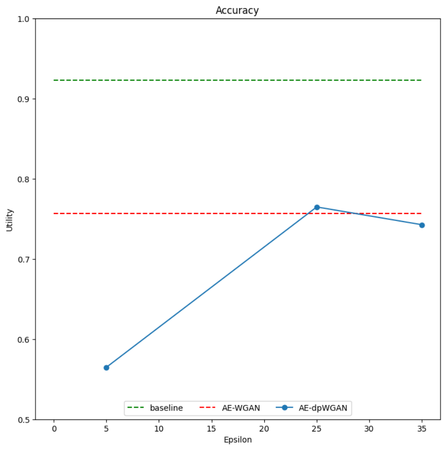


Figure: Results of AE-(dp)WGAN with different privacy budgets (lower epsilon means higher privacy)

>>> Preliminary Results

$ AE-(dp)MERF performs best and is very efficient computationally.

>>> Preliminary Results

- $ **AE-(dp)MERF performs best** and is very efficient computationally.
- $ AE-(dp)MERF can work in **lower epsilon ranges**, which translates to **stronger privacy guarantees**.

>>> Preliminary Results

- $ `AE-(dp)MERF performs best` and is very efficient computationally.
- $ AE-(dp)MERF can work in `lower epsilon ranges`, which translates to `stronger privacy guarantees`.
- $ `AE-(dp)WGAN gives worse utility` and can only work with meaningless privacy budgets ϵ.

>>> Preliminary Results

- $ AE-(dp)MERF performs best and is very efficient computationally.
- $ AE-(dp)MERF can work in lower epsilon ranges, which translates to stronger privacy guarantees.
- $ AE-(dp)WGAN gives worse utility and can only work with meaningless privacy budgets ϵ.
- $ We lose utility when replacing original data with non-private synthetic data.

>>> Preliminary Results

- $ AE-(dp)MERF performs best and is very efficient computationally.
- $ AE-(dp)MERF can work in lower epsilon ranges, which translates to stronger privacy guarantees.
- $ AE-(dp)WGAN gives worse utility and can only work with meaningless privacy budgets ϵ.
- $ We lose utility when replacing original data with non-private synthetic data.
- $ BUT: Adding privacy does not further degrade the utility for anomaly detection too much until too much noise is added.

>>> Contamination

We **contaminate** the train set that only consists of regular samples with **1%, 2%, 5% anomalous samples** (the percentage of heartbeat arrhythmias is estimated to be around max. 5%).

>>> Contamination

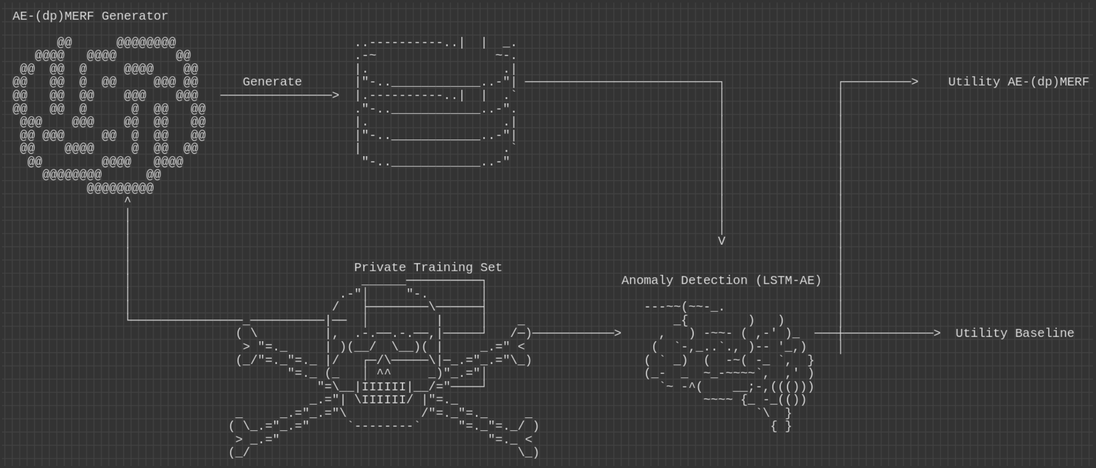


Figure: Structure of Contamination Experiment

>>> Contamination: AE-(DP)MERF

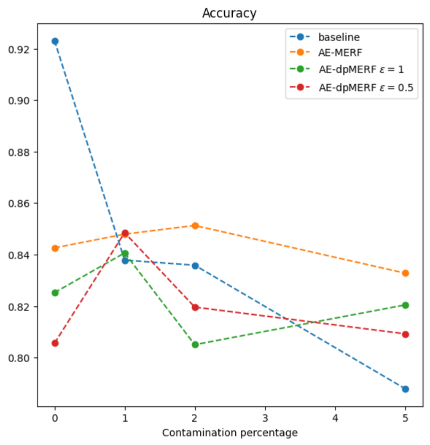


Figure: Contaminated training set: AE-(DP)MERF

>>> Contamination: AE-(DP)MERF

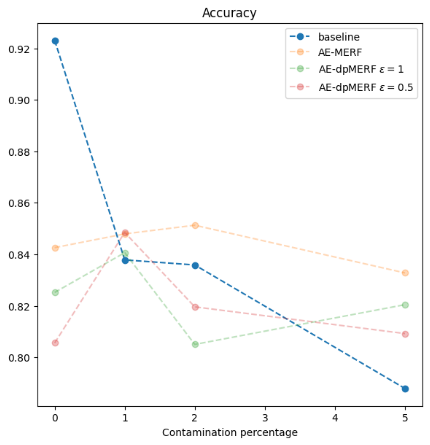


Figure: Contaminated training set: AE-(DP)MERF

$ **Baseline model** performance degrades with increasing contamination percentage.

>>> Contamination: AE-(DP)MERF

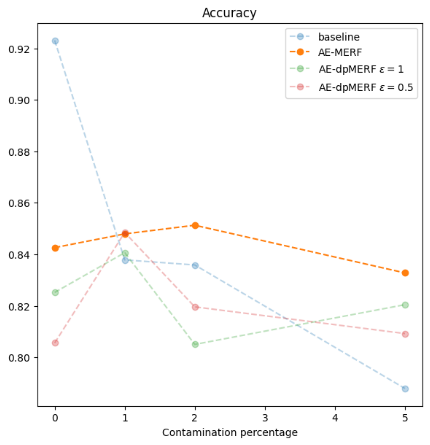


Figure: Contaminated training set: AE-(DP)MERF

- $ **Baseline model** performance degrades with increasing contamination percentage.
- $ **AE-MERF** generated samples retain stable utility.

>>> Contamination: AE-(DP)MERF

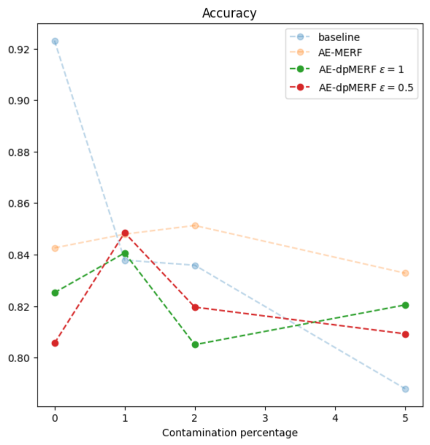


Figure: Contaminated training set: AE-(DP)MERF

- $ **Baseline model** performance degrades with increasing contamination percentage.
- $ **AE-MERF** generated samples retain stable utility.
- $ Utility of **AE-dpMERF** generated samples first increases and then decrease when contamination is too high.

>>> Contamination: AE-(DP)MERF

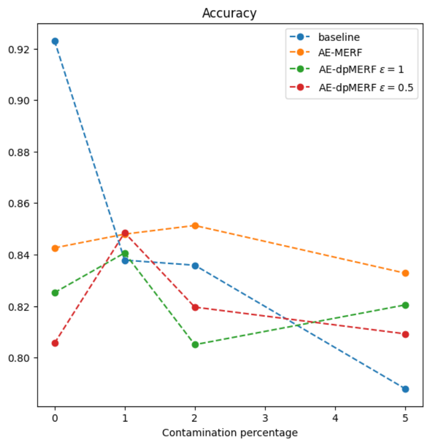


Figure: Contaminated training set: AE-(DP)MERF

- $ **Baseline model** performance degrades with increasing contamination percentage.
- $ **AE-MERF** generated samples retain stable utility.
- $ Utility of **AE-dpMERF** generated samples first increases and then decrease when contamination is too high.
- $ Utility of synthetic data is higher than baseline model

>>> Contamination: AE-(DP)MERF

Hypothesis: Noise added during data generation and DP noise can have a **regularising effect** on the synthetic data which counteracts the contamination.

>>> Outline

1. Project introduction
2. Dataset: MITBIH ECG data
3. Privacy-preserving Time Series Data Generation
4. Results
5. Summary
6. References

>>> Summary

- $ We tested two different `DP times series data generation` models on the MITBIH ECG data set.

>>> Summary

- $ We tested two different `DP times series data generation` models on the MITBIH ECG data set.
- $ We measured the `utility` of the synthetic data via the downstream task of anomaly detection (heartbeat arrhythmia).

>>> Summary

- $ We tested two different `DP times series data generation` models on the MITBIH ECG data set.
- $ We measured the `utility` of the synthetic data via the downstream task of anomaly detection (heartbeat arrhythmia).
- $ We investigated the `robustness` of the data generation by `contaminating` the data set with anomalous samples.

>>> Main Findings

$ AE-(dp)MERF works better than GAN based approach.

>>> Main Findings

- $ AE-(dp)MERF works better than GAN based approach.
- $ The **Privacy-Utility-Tradeoff is more nuanced** and depends on the use case. For anomaly detection, privacy and utility can go hand in hand.

>>> Main Findings

- $ AE-(dp)MERF works better than GAN based approach.
- $ The `Privacy-Utility-Tradeoff is more nuanced` and depends on the use case. For anomaly detection, privacy and utility can go hand in hand.
- $ Synthetic data and DP can add `robustness`.

>>> Future Work

$ Test with other time series data.

>>> Future Work

- $ Test with **other time series data**.
- $ Work with **other use cases**, e.g. classification, regression.

>>> Future Work

- $ Test with **other time series data**.
- $ Work with **other use cases**, e.g. classification, regression.
- $ Further investigate **robustness**.

>>> Future Work

- $ Test with `other time series data`.
- $ Work with `other use cases`, e.g. classification, regression.
- $ Further investigate `robustness`.
- $ Verify theoretical privacy guarantees with empirical tests, e.g. `membership inference attacks`.

>>> Privacy-Preserving Acknowledgement

Thank you Aflsono, Adenrs, Apslotsuo, Hna, Sihahd!

>>> Outline

1. Project introduction
2. Dataset: MITBIH ECG data
3. Privacy-preserving Time Series Data Generation
4. Results
5. Summary
6. References

Ang, Yihao et al. (2023). TSGBench: Time Series Generation Benchmark. arXiv: 2309.03755 [cs.LG].

Dwork, Cynthia, Aaron Roth, et al. (2014). “The algorithmic foundations of differential privacy”. In: Foundations and Trends® in Theoretical Computer Science 9.3–4, pp. 211–407.

Harder, Frederik, Kamil Adamczewski, and Mijung Park (2020). “Differentially Private Mean Embeddings with Random Features (DP-MERF) for Simple & Practical Synthetic Data Generation”. In: CoRR abs/2002.11603. arXiv: 2002.11603. URL: https://arxiv.org/abs/2002.11603.

Hu, Yuzheng et al. (2023). SoK: Privacy-Preserving Data Synthesis. arXiv: 2307.02106 [cs.CR].

Pei, Hengzhi et al. (2021). “Towards Generating Real-World Time Series Data”. In: Proceedings of the 2021 IEEE International Conference on Data Mining (ICDM).

>>> BACKUP

>>> Model Architecture

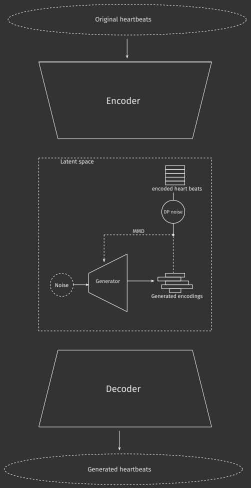


Figure: AE-(dp)MERF architecture

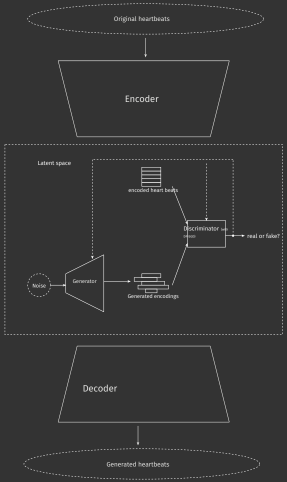


Figure: Architecture of AE-(dp)WGAN

>>> AE-(dp)WGAN

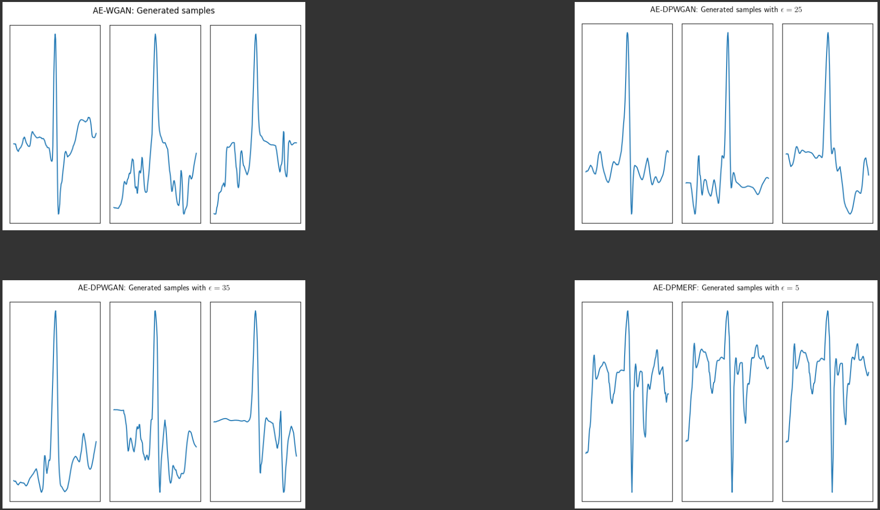


Figure: AE-(dp)WGAN generated samples

>>> Gaussian Mechanism

For a given function $f : \mathbb{N}^{|\mathcal{X}|} \longrightarrow \mathbb{R}^d$, privacy parameters $\epsilon \in (0, 1)$ and $\delta > 0$ define the gaussian mechanism $F(x)$ as follows:

$$F(x) = f(x) + \mathcal{N}(0, \sigma^2) \tag{5}$$

where the variance is calibrated by the sensitivity of f and the given privacy level, such that $\sigma \geq \frac{2\Delta f}{\epsilon} \ln(\frac{1.25}{\delta})$

>>> Performance metrics

- $ Accuracy measures the overall percentage of correct classifications:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad . \tag{6}$$

- $ Precision looks only on the samples that are labelled as anomalies and computes the percentages of correctly detected anomalies:

$$Precision = \frac{TP}{TP + FP} \quad . \tag{7}$$

- $ Recall looks at all true anomalies and computes the percentage of correctly detected anomalies

$$Recall = \frac{TP}{TP + FN} \quad . \tag{8}$$

- $ F1 computes the an average of Precision and Recall

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad . \tag{9}$$

>>> DP Illustrated

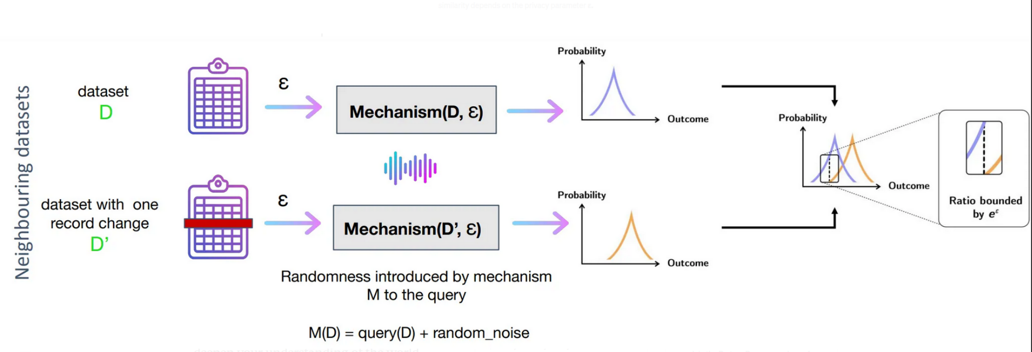


Figure: Illustration of DP[2]

[2] taken from: https://medium.com/dsaid-govtech/protecting-your-data-privacy-with-differential-privacy-an-introduction-abee1d7fcb63

>>> Classification based on Reconstruction Error

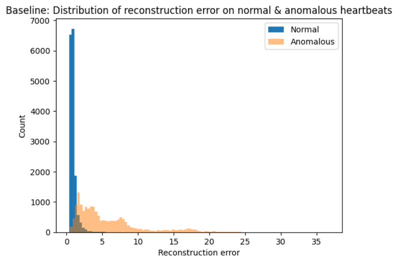


Figure: Distribution of reconstruction error on regular & anomalous samples

We can clearly see a **difference in error distribution** for regular and anomalous samples. We choose the **threshold that maximises the classification accuracy**.

>>> Classification threshold

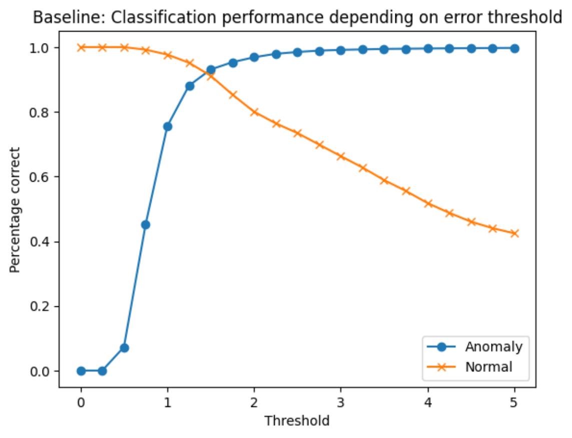


Figure: Adjusting the threshold according to classification accuracy

>>> Models

AE-(dp)MERF

- $ AE-(dp)MERF is based on DP-MERF (best state of the art generator for tabular data).
- $ Simple architecture with mathematically sophisticated loss function.
- $ Does not work with time series data out of the box, but we will modify it so it works.

AE-(dp)WGAN

- $ Model based on GAN network, which are commonly used in image generation.
- $ Based on RTSGAN, which delivers state of the art performance for time series data.
- $ No private counterpart, hence we will implement our own private version.

>>> AE-(dp)MERF: Utility

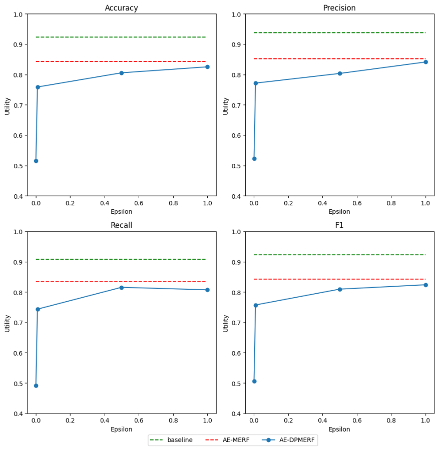


Figure: Results of AE-(DP)MERF with different privacy budgets (lower epsilon means higher privacy)

>>> AE-(dp)WGAN: Utility

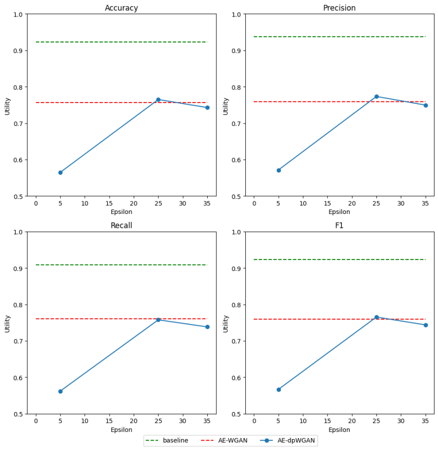


Figure: Results of AE-(dp)WGAN with different privacy budgets (lower epsilon means higher privacy)

>>> Contamination: AE-(DP)MERF

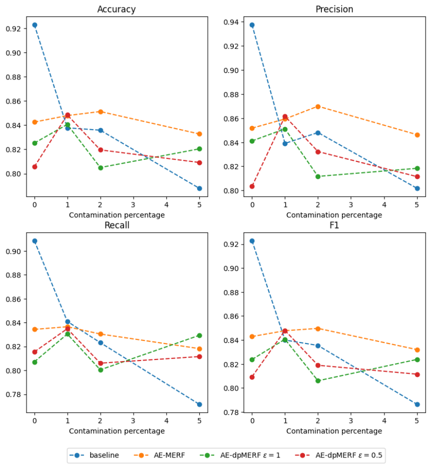


Figure: Contaminated training set: AE-(DP)MERF

- $ **Baseline model** performance degrades with increasing contamination percentage.
- $ **AE-MERF** generated samples retain stable utility.
- $ Utility of **AE-dpMERF** generated samples first increases and then decrease when contamination is too high.
- $ Utility of synthetic data is higher than baseline model