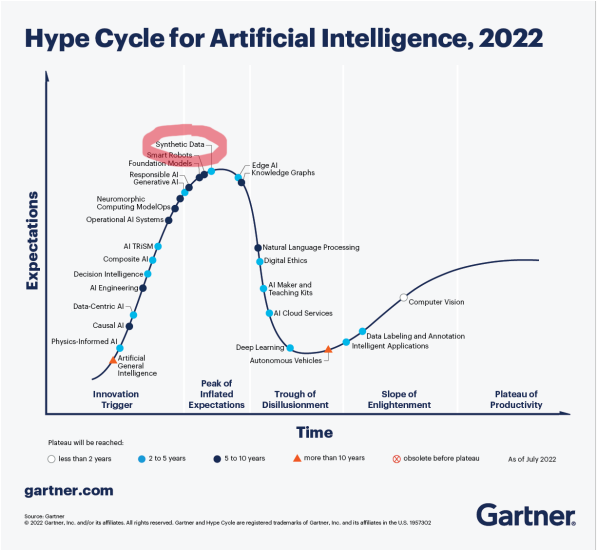
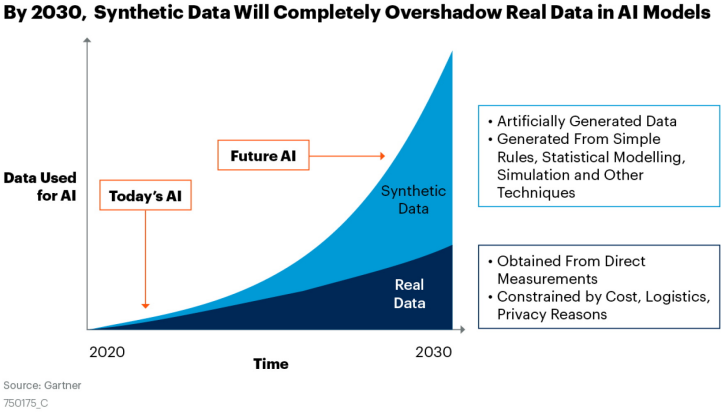
Statistics 414 From Predictive AI to Generative AI
Intro to the Evaluation of Generative Data

Guang Cheng

Department of Statistics & Data Science, UCLA

Trend in synthetic data





Synthetic data is everywhere: image, text, graph and tabular data.

It has different names: simulated data, missing value imputation, and diffusion etc

Can synthetic data create something out of nothing?

Enhancing adversarial robustness [1]

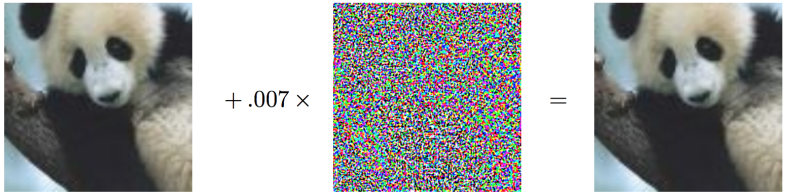
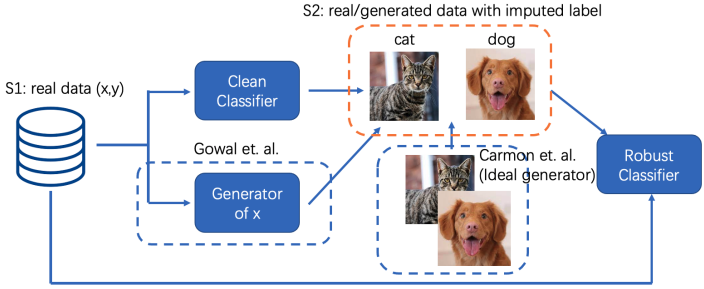


Figure: Adversarial attack: slight data perturbation leads to incorrect prediction

[1] Xing, Song and C. (2022) Why Do Artificially Generated Data Help Adversarial Robustness? *NeurIPS*

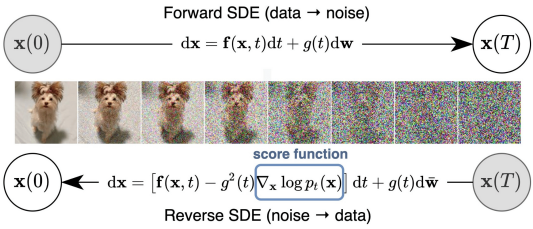


Figure

Adversarial robustness is greatly enhanced by properly mixing true data and synthetic data in the model training.

Imputing missing values[2]

The vanilla diffusion model can be adapted to impute missing values: *generation-as-imputation*.



[2] Ouyang, Xie, Li and C. (2024) MissDiff: Training Diffusion Models on Tabular Data with Missing Values. *arXiv*

Imputing missing values

Miss-Diff

temperature	humidity	person	ac_on	ac_off
23.945565613233800	46.260621462405800	1	0	
	44.54788892074370	1		0
25.36588628043900		0	0	1
	57.533732428941200	0		0
26.143596412902700		1		0
22.364349174453200		1	1	0
	50.3870024038287O	0	0	

Incomplete data

Learning a diffusion model → Diffusion model $f_\theta\colon z \to x$

objective

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} J_{DSM}(\boldsymbol{\theta})$$

$$:= \frac{T}{2}\mathbb{E}_t\Big\{\lambda(t)\mathbb{E}_{\mathbf{x}^{\text{obs}}(0)}\mathbb{E}_{\mathbf{x}^{\text{obs}}(t)|\mathbf{x}^{\text{obs}}(0)}\Big[\Big\|\Big(\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}^{\text{obs}}(t),t) - \nabla_{\mathbf{x}^{\text{obs}}(t)}\log p(\mathbf{x}^{\text{obs}}(t)\mid\mathbf{x}^{\text{obs}}(0))\Big)\odot\mathbf{m}\Big\|_2^2\Big]\Big\},$$

In comparison with traditional statistical imputation, the Miss-Diff can be viewed as a *global* imputation approach by learning the underlying score function of complete data distribution, i.e., masked score matching.

More benefits

- Preserve privacy

More benefits

- Preserve privacy
  - Li, Wang and C. (2023) Statistical Theory of Differentially Private Marginal-based Data Synthesis Algorithms, *ICLR*;

More benefits

- Preserve privacy
  - Li, Wang and C. (2023) Statistical Theory of Differentially Private Marginal-based Data Synthesis Algorithms, *ICLR*;
- Improve utility of downstream tasks

More benefits

- Preserve privacy
  - Li, Wang and C. (2023) Statistical Theory of Differentially Private Marginal-based Data Synthesis Algorithms, *ICLR*;
- Improve utility of downstream tasks
  - Hsieh, Wang and C. (2023) Improve Fidelity and Utility of Synthetic Credit Card Transaction Time Series from Data-centric Perspective, *ACM International Conference on AI in Finance – Workshop*;

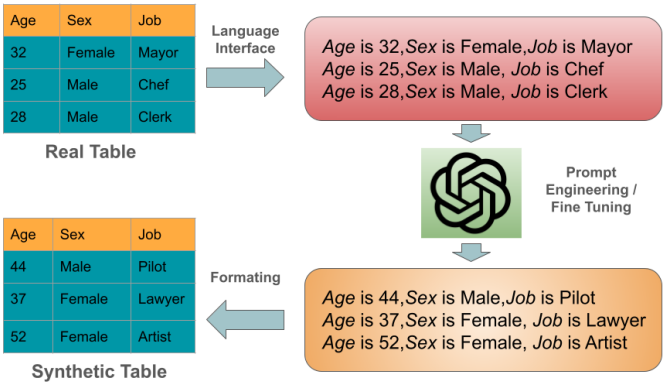
More benefits

- Preserve privacy
  - Li, Wang and C. (2023) Statistical Theory of Differentially Private Marginal-based Data Synthesis Algorithms, *ICLR*;
- Improve utility of downstream tasks
  - Hsieh, Wang and C. (2023) Improve Fidelity and Utility of Synthetic Credit Card Transaction Time Series from Data-centric Perspective, *ACM International Conference on AI in Finance – Workshop*;
- Promote fairness

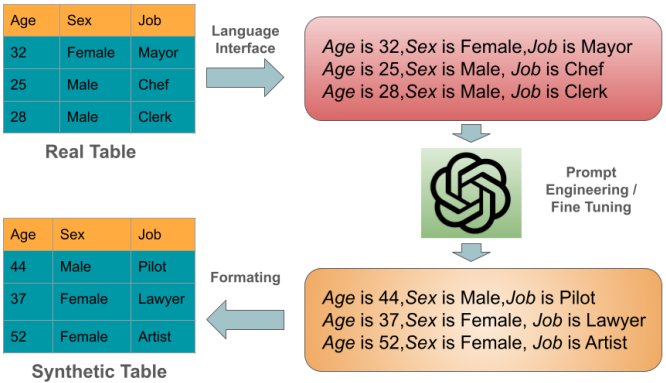
More benefits

- Preserve privacy
  - Li, Wang and C. (2023) Statistical Theory of Differentially Private Marginal-based Data Synthesis Algorithms, *ICLR*;
- Improve utility of downstream tasks
  - Hsieh, Wang and C. (2023) Improve Fidelity and Utility of Synthetic Credit Card Transaction Time Series from Data-centric Perspective, *ACM International Conference on AI in Finance – Workshop*;
- Promote fairness
  - Zeng, C. and Dorbriban. (2024) Bayes-Optimal Fair Classification with Linear Disparity Constraints via Pre-, In-, and Post-processing, *arXiv*.

A recent trend on the use of LLM

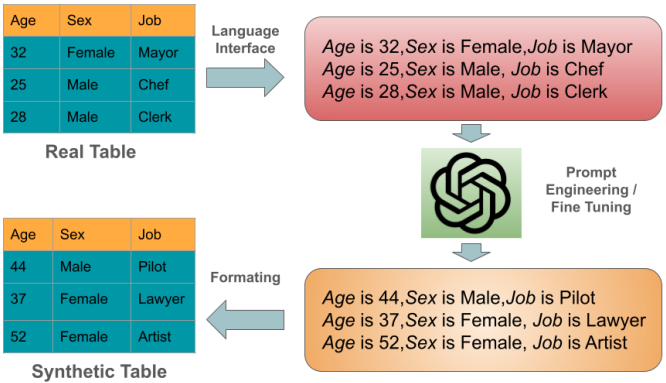


A recent trend on the use of LLM



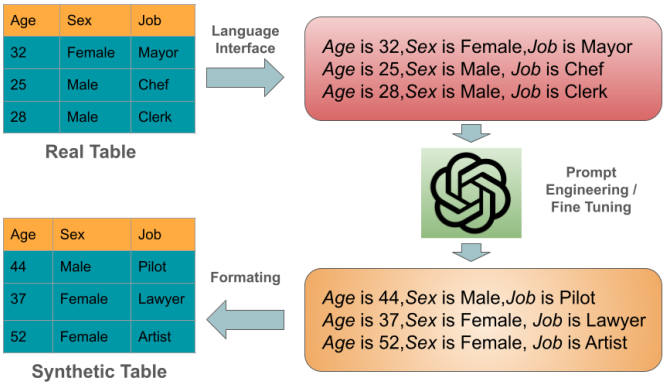
- The fidelity of synthesized tables is *surprisingly* high;

A recent trend on the use of LLM



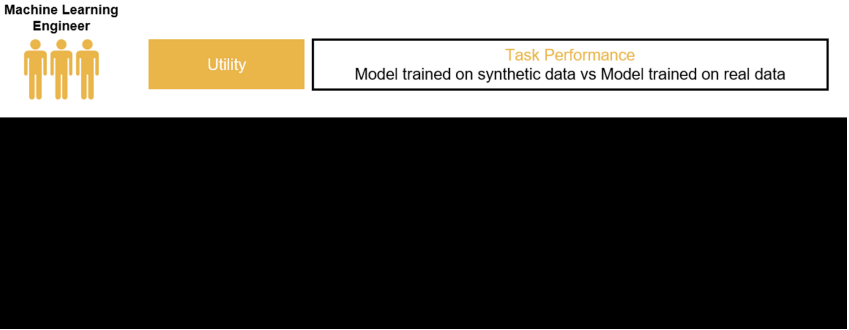
- The fidelity of synthesized tables is *surprisingly* high;
- The table can have mixed-type data and be under some constraints;

A recent trend on the use of LLM

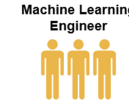


- The fidelity of synthesized tables is *surprisingly* high;
- The table can have mixed-type data and be under some constraints;
- Pre-trained on web-scale data, LLMs can augment small datasets with diverse samples, leveraging their in-context learning ability.

Evaluation of synthetic data: utility, privacy & fidelity



Evaluation of synthetic data: utility, privacy & fidelity



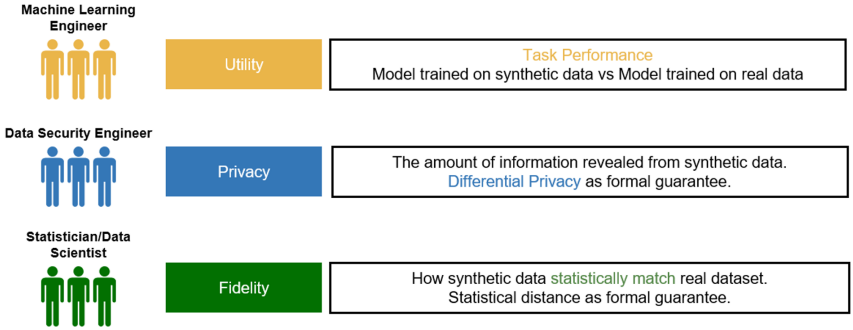
Utility

Task Performance
Model trained on synthetic data vs Model trained on real data

Privacy

The amount of information revealed from synthetic data.
Differential Privacy as formal guarantee.

Evaluation of synthetic data: utility, privacy & fidelity



An emerging field of "Generative Data Science?"

Two concrete examples to illustrate Generative Data Science

Example I. Data-copying[3]

- Data-copying, i.e., memorization, is a type of over-fitting phenomenon in generative models. It may lead to serious privacy leakage in training data;

[3] Bhattacharjee et al (2023) Data-copying in generative models: a formal framework. *ICML*

Example I. Data-copying[3]

- Data-copying, i.e., memorization, is a type of over-fitting phenomenon in generative models. It may lead to serious privacy leakage in training data;
- We will use the most common nonparametric estimate: Kernel Density Estimator (KDE) to illustrate the idea of *"statistical consistency hurts privacy."*

[3] Bhattacharjee et al (2023) Data-copying in generative models: a formal framework. *ICML*

Kernel density estimator 101

- Given a real dataset $D = \{x_1, x_2, \cdots, x_n\}$ with $x_i \sim p(x)$, the KDE is defined as

$$q(x) := \widehat{p}(x) = \frac{1}{n\sigma_n} \sum_{x_i \in D} K\left(\frac{x - x_i}{\sigma_n}\right), \tag{1}$$

  which can be understood as a very naive generative model;

Kernel density estimator 101

- Given a real dataset $D = \{x_1, x_2, \cdots, x_n\}$ with $x_i \sim p(x)$, the KDE is defined as

$$q(x) := \widehat{p}(x) = \frac{1}{n\sigma_n} \sum_{x_i \in D} K\left(\frac{x - x_i}{\sigma_n}\right), \tag{1}$$

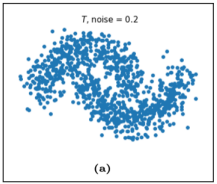
  which can be understood as a very naive generative model;
- According to nonparametric statistics, the statistical consistency of KDE follows as long as the bandwidth

$$\sigma_n \sim n^{-1/5}.$$

Generation of the Halfmoon dataset

KDE : Kernel Density Estimation

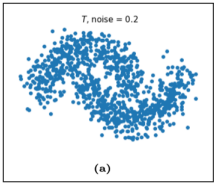
Training Data

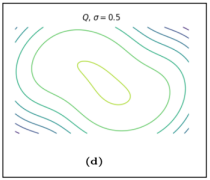


Generation via under-fit KDE

KDE : Kernel Density Estimation

Training Data





Underfit KDE

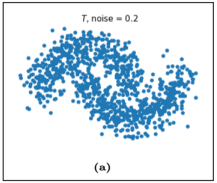
Bad Fidelity

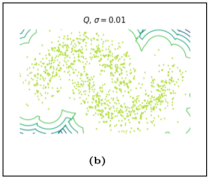
Did Not Catch Pattern

Generation via over-fit KDE (data-copying occurs)

KDE : Kernel Density Estimation

Training Data

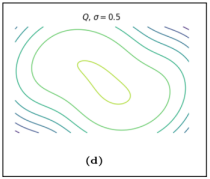




Max Likelihood KDE
(Over Training Dataset)
Great Fidelity

Great Privacy Invasion

(Data Copying)



Underfit KDE

Bad Fidelity

Did Not Catch Pattern

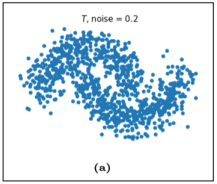
Figure:

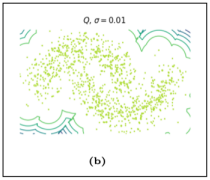
Using training data to estimate σ, giving $\sigma = 0.01$, leads to statistical consistency, but also data-copying (see Appendix for definition).

Generation via holdout KDE

KDE : Kernel Density Estimation

Training Data



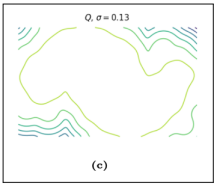


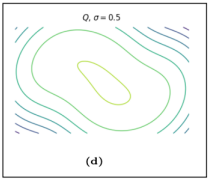
Max Likelihood KDE
(Over Training Dataset)
Great Fidelity

Great Privacy Invasion

(Data Copying)

Max Likelihood KDE
(Over Holdout Dataset)
Good Fidelity
No Privacy Invasion





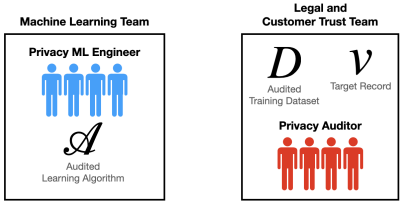
Underfit KDE

Bad Fidelity

Did Not Catch Pattern

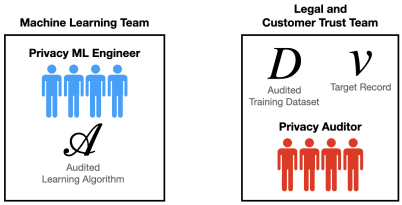
Figure: The holdout KDE has good fidelity and *no privacy invasion*.

Example 2: Auditing generator's data privacy



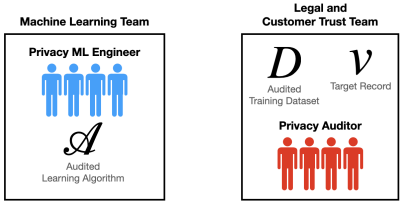
- Privacy ML engineers claim that their algorithm is $\epsilon-$DP with $\epsilon = 2$.

Example 2: Auditing generator's data privacy



- Privacy ML engineers claim that their algorithm is $\epsilon-$DP with $\epsilon = 2$.
- Manager: "Looks good to me. I believe in you."

Example 2: Auditing generator's data privacy



- Privacy ML engineers claim that their algorithm is $\epsilon-$DP with $\epsilon = 2$.
- Manager: "Looks good to me. I believe in you."
- Privacy auditor: "let's get some empirical evidence!"

DP in synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote two synthetic datasets, where the generator $\mathcal{G}$ (such as LLM) was trained on D_0 or D_1;

DP in synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote two synthetic datasets, where the generator $\mathcal{G}$ (such as LLM) was trained on D_0 or D_1;
- We assume D_0 and D_2 are two "neighboring" datasets;

DP in synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote two synthetic datasets, where the generator $\mathcal{G}$ (such as LLM) was trained on D_0 or D_1;
- We assume D_0 and D_2 are two "neighboring" datasets;
- Question 1: if the median income $m(D)$ is of interest, can we show

$$\mathbb{P}\left(m\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(m\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

DP in synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote two synthetic datasets, where the generator $\mathcal{G}$ (such as LLM) was trained on D_0 or D_1;
- We assume D_0 and D_2 are two "neighboring" datasets;
- Question 1: if the median income $m(D)$ is of interest, can we show

$$\mathbb{P}\left(m\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(m\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

- Question 2: if training an income predictor $\theta(\widetilde{D})$, can we show

$$\mathbb{P}\left(\theta\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(\theta\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

DP in synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote two synthetic datasets, where the generator $\mathcal{G}$ (such as LLM) was trained on D_0 or D_1;
- We assume D_0 and D_2 are two "neighboring" datasets;
- Question 1: if the median income $m(D)$ is of interest, can we show

$$\mathbb{P}\left(m\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(m\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

- Question 2: if training an income predictor $\theta(\widetilde{D})$, can we show

$$\mathbb{P}\left(\theta\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(\theta\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

- To answer both questions, we need to have a deep understanding on the design of $\mathcal{G}$ which induces the probability measure $\mathbb{P}$.

Result 1: the choice of utility metric is subtle [4]

Q: Does synthetic dataset need to be a perfect twin of real dataset?

[4] Cheng, Wang, Potluru, Balch and C. (2022)

Result 1: the choice of utility metric is subtle [4]

Q: Does synthetic dataset need to be a perfect twin of real dataset?

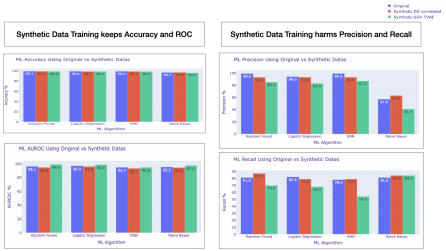


Figure: Fraud detection models trained by real (blue) or synthetic data (red, green)

[4] Cheng, Wang, Potluru, Balch and C. (2022)

Result 1: the choice of utility metric is subtle [4]

Q: Does synthetic dataset need to be a perfect twin of real dataset?

Figure: Fraud detection models trained by real (blue) or synthetic data (red, green)

Observations:

- Train on Synthetic $=$ Train on Real if we care `Accuracy` or `AUROC`
- Train on Synthetic $\neq$ Train on Real if we care `Precision` or `Recall`

[4] Cheng, Wang, Potluru, Balch and C. (2022)

3 Questions for synthetic training data generation

Given the observations

- Train on Synthetic $=$ Train on Real if we care `Accuracy` or `AUROC`
- Train on Synthetic $\neq$ Train on Real if we care `Precision` or `Recall`

3 Questions for synthetic training data generation

Given the observations

- Train on Synthetic $=$ Train on Real if we care `Accuracy` or `AUROC`
- Train on Synthetic $\neq$ Train on Real if we care `Precision` or `Recall`

Natural questions are

Q1 Do the same observations hold *if* we change fraud detection models?

Q2 Is there a best synthesizer *if* we want to maximize `AUROC` (or other utility metric)?

Q3 Can synthetic data improves fraud detection model utility (`AUROC`, `Precision-Recall`)?

3 Questions for synthetic training data generation

Given the observations

- Train on Synthetic $=$ Train on Real if we care `Accuracy` or `AUROC`
- Train on Synthetic $\neq$ Train on Real if we care `Precision` or `Recall`

Natural questions are

Q1 Do the same observations hold *if* we change fraud detection models?

Q2 Is there a best synthesizer *if* we want to maximize `AUROC` (or other utility metric)?

Q3 Can synthetic data improves fraud detection model utility (`AUROC`, `Precision-Recall`)?

Our investigation gives answers

A1 Yes, still hold (We test 9 different models, from easy to complex).

A2 No clear winner, but rule of thumbs (Based on 180 scenarios).

A3 Yes! Train on (1) fully synthetic or (2) real+synthetic DO improves model utility (for certain model classes)!

A1: synthetic data utility is model-dependent

Q1: Do the same observations hold *if* we change fraud detection models?

We investigate 4 different synthesizers across 45 scenarios (5 metrics x 9 models).

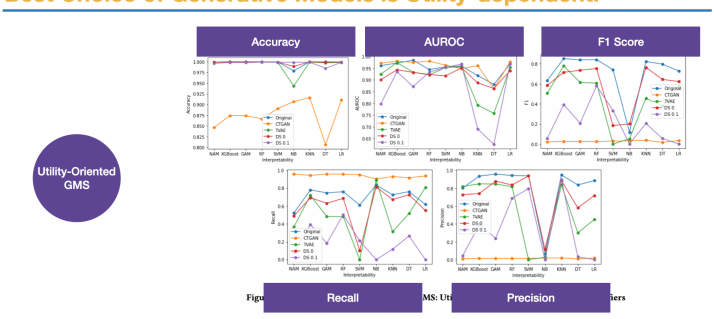


Figure: Benchmark performance is the blue line (Train with real data)

A1: synthetic data utility is model-dependent

Q1: Do the same observations hold *if* we change fraud detection models?

We investigate 4 different synthesizers across 45 scenarios (5 metrics x 9 models).

Best Choice of Generative Models is Utility-dependent.

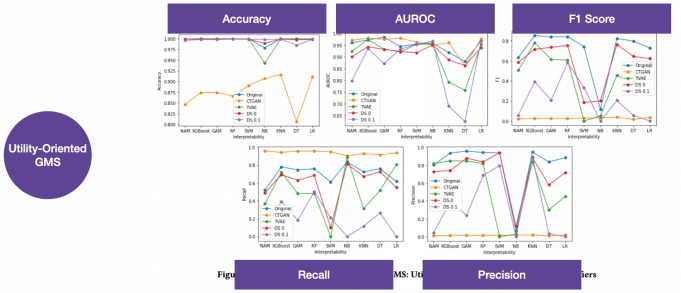
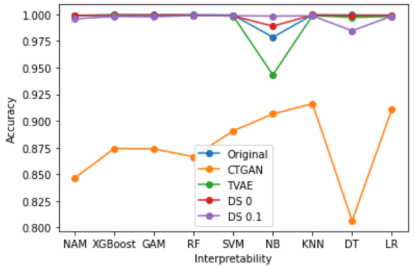


Figure: Benchmark performance is the blue line (Train with real data)

Choosing right synthesizer does improve model utility!

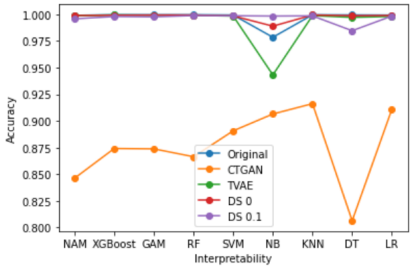
A2: Best synthesizer is utility-dependent (Accuracy)

Q2: Is there a best synthesizer *if* we want to maximize Accuracy ?



A2: Best synthesizer is utility-dependent (Accuracy)

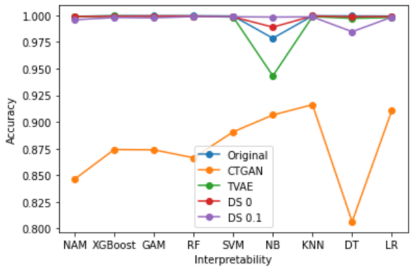
Q2: Is there a best synthesizer *if* we want to maximize Accuracy ?



If utility metric = Accuracy....

A2: Best synthesizer is utility-dependent (Accuracy)

Q2: Is there a best synthesizer *if* we want to maximize Accuracy ?

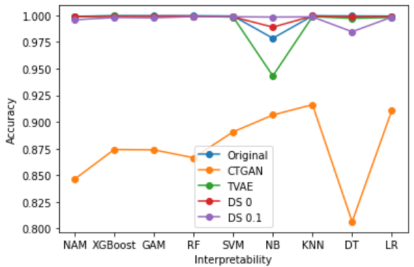


If utility metric = Accuracy....

- Train on Synthetic = Train on Real! (except CTGAN-generated synthetic data)

A2: Best synthesizer is utility-dependent (Accuracy)

Q2: Is there a best synthesizer *if* we want to maximize Accuracy ?

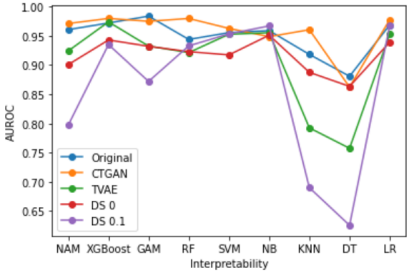


If utility metric = Accuracy....

- Train on Synthetic = Train on Real! (except CTGAN-generated synthetic data)
- However, Accuracy is not a good metric due to training dataset imbalance.

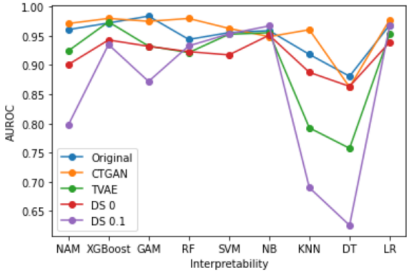
A2: best synthesizer is utility-dependent (AUROC)

Q2: Is there a best synthesizer *if* we want to maximize AUROC ?



A2: best synthesizer is utility-dependent (AUROC)

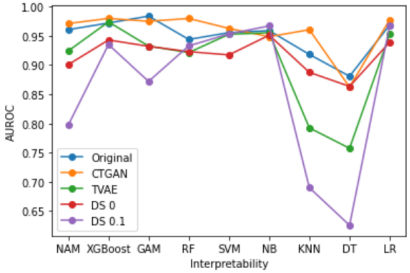
Q2: Is there a best synthesizer *if* we want to maximize AUROC ?



If utility metric = AUROC....

A2: best synthesizer is utility-dependent (AUROC)

Q2: Is there a best synthesizer *if* we want to maximize AUROC ?

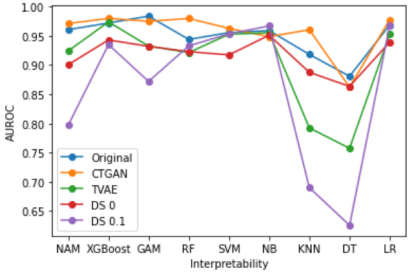


If utility metric = AUROC....

- Train on Synthetic $\neq$ Train on Real! (CTGAN is the best!)

A2: best synthesizer is utility-dependent (AUROC)

Q2: Is there a best synthesizer *if* we want to maximize AUROC ?

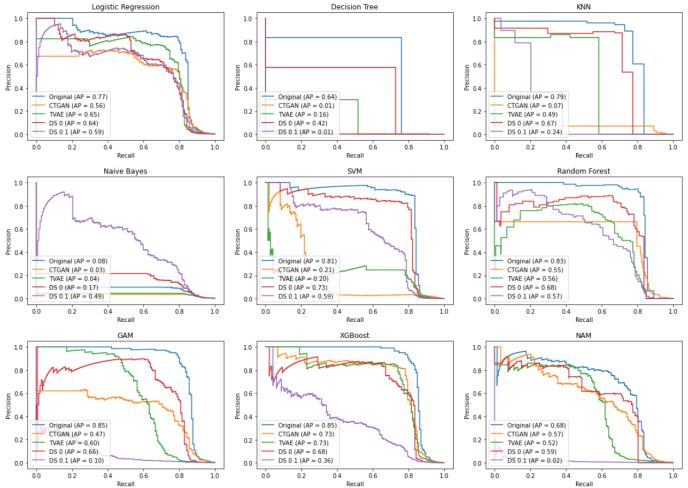


If utility metric = AUROC....

- Train on Synthetic $\neq$ Train on Real! (CTGAN is the best!)
- Training on CTGAN-generated synthetic data DOES improves AUROC than training on real data!

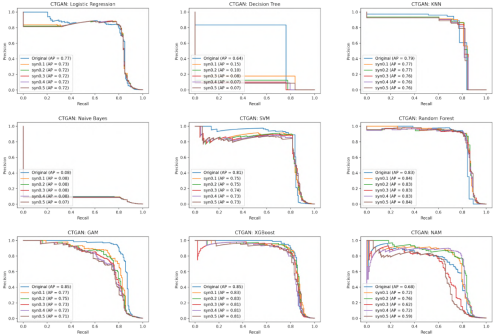
A2: no clear winning synthesizer for precision-recall curve

Q2: Is there a best synthesizer *if* we want to maximize `Precision-Recall` ?



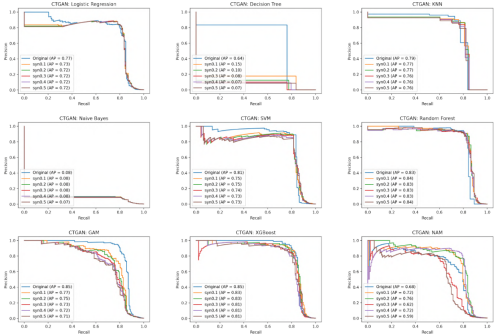
A3: CTGAN-augmented training helps model utility!

Q3: Will mixing real data with CTGAN-generated data improve model `Precision-Recall` ?



A3: CTGAN-augmented training helps model utility!

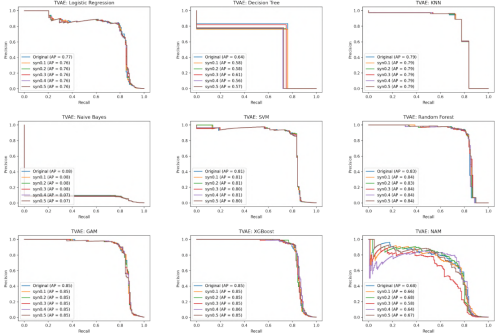
Q3: Will mixing real data with CTGAN-generated data improve model `Precision-Recall` ?



100% Real + 20% Synthetic DOES improve Neural Additive Model Utility

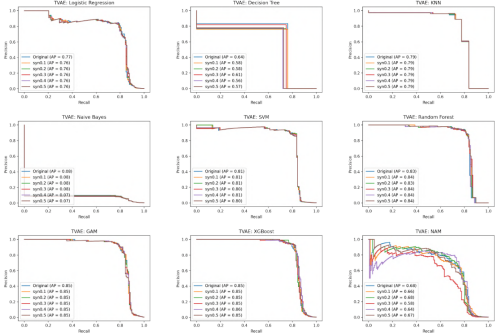
A3: TVAE-augmented training may not help model utility!

Q3: Will mixing real data with TVAE-generated data improve model `Precision-Recall` ?



A3: TVAE-augmented training may not help model utility!

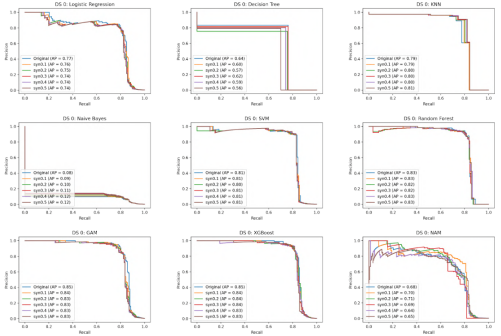
Q3: Will mixing real data with TVAE-generated data improve model `Precision-Recall` ?



No clear improvement for 100% Real + 10 - 50 % TVAE synthetic

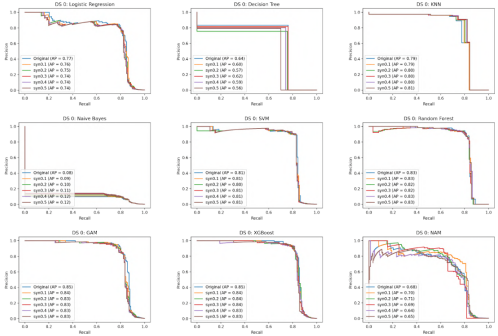
A3: PrivBayes-augmented training helps model utility!

Q3: Will mixing real data with PrivBayes-generated data improves model `Precision-Recall` ?



A3: PrivBayes-augmented training helps model utility!

Q3: Will mixing real data with PrivBayes-generated data improves model `Precision-Recall` ?



100% Real + 30% Synthetic DO improves Neural Additive Model Utility

Summary

Insights on synthetic trained classifier

- BN-based synthetic training data have comparable classification accuracy as NN-based synthetic training data.
- BN-based synthetic training data have better `F1 score` and `Precision` than NN-based synthetic training data.
- NN-based synthetic training data have better `AUROC` and `Recall` than BN-based synthetic training data.

Summary

Insights on synthetic trained classifier

- BN-based synthetic training data have comparable classification accuracy as NN-based synthetic training data.
- BN-based synthetic training data have better `F1 score` and `Precision` than NN-based synthetic training data.
- NN-based synthetic training data have better `AUROC` and `Recall` than BN-based synthetic training data.

Insights on real+synthetic trained classifier

- Mixing real data with synthetic training data does not improve performance if the classifier is intrinsic and medium interpretable.
- Mixing real data with synthetic training data DOES improve performance if the classifier is not-easy interpretable.

Summary

Insights on synthetic trained classifier

- BN-based synthetic training data have comparable classification accuracy as NN-based synthetic training data.
- BN-based synthetic training data have better `F1 score` and `Precision` than NN-based synthetic training data.
- NN-based synthetic training data have better `AUROC` and `Recall` than BN-based synthetic training data.

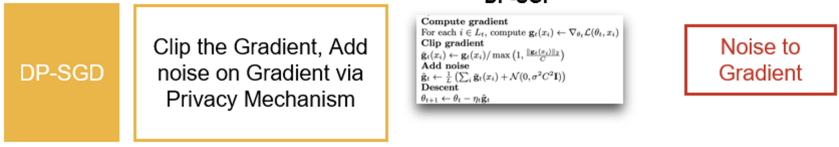
Insights on real+synthetic trained classifier

- Mixing real data with synthetic training data does not improve performance if the classifier is intrinsic and medium interpretable.
- Mixing real data with synthetic training data DOES improve performance if the classifier is not-easy interpretable.

There seems a interpretability-utility trade off when using synthetic data as training data.

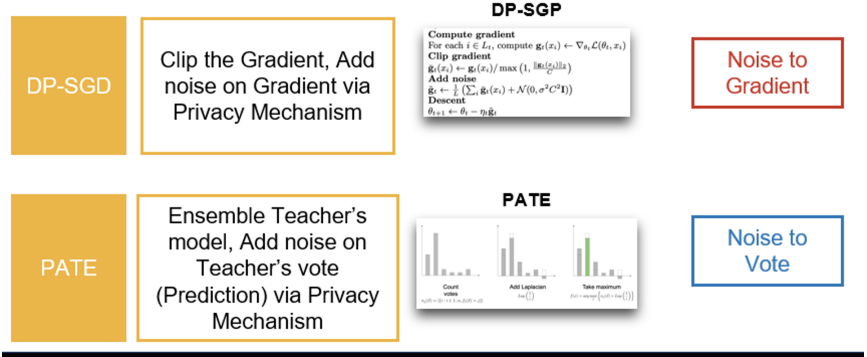
Result 2: the subtle trade-off between utility and privacy[5]

Figure: 3 ways to traing differential private tabular data synthesizers



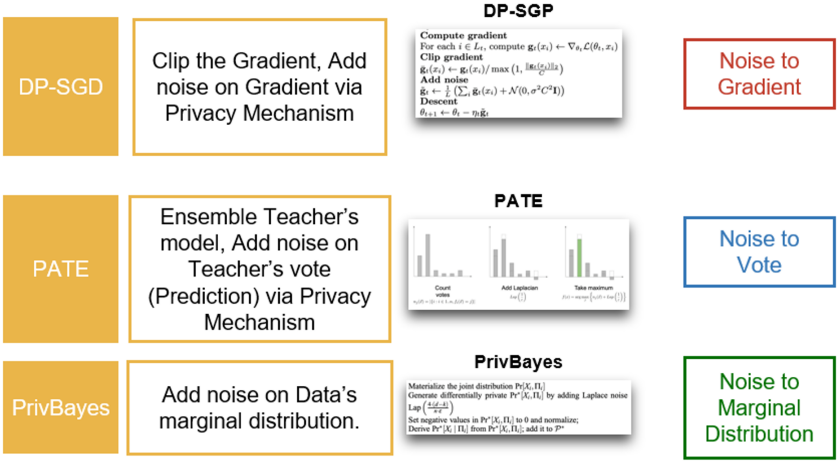
Result 2: the subtle trade-off between utility and privacy[5]

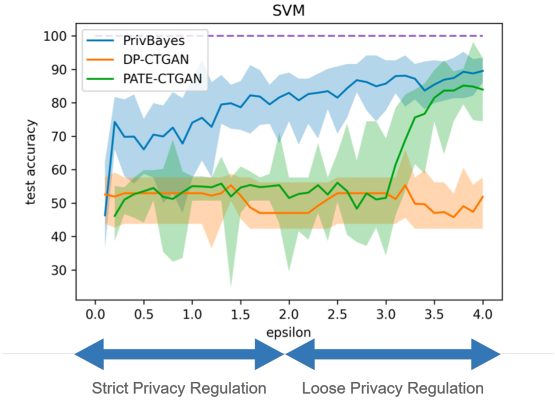
Figure: 3 ways to traing differential private tabular data synthesizers

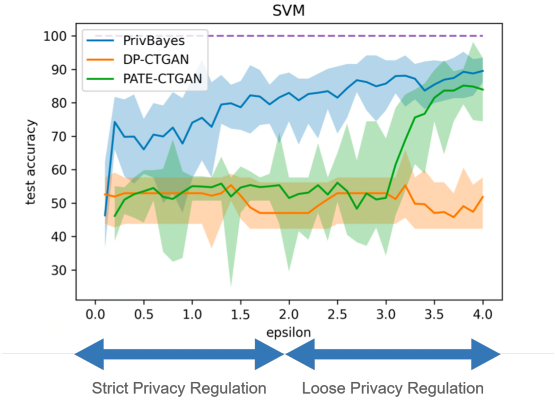


Result 2: the subtle trade-off between utility and privacy[5]

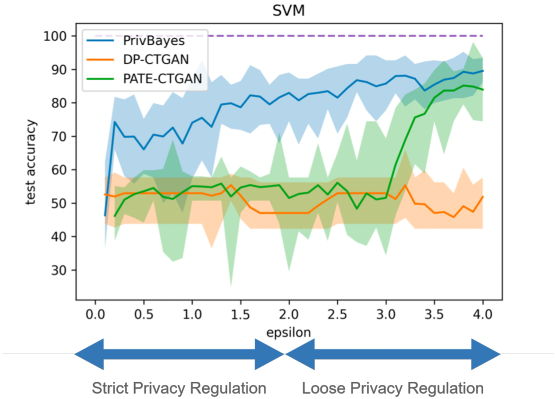
Figure: 3 ways to traing differential private tabular data synthesizers



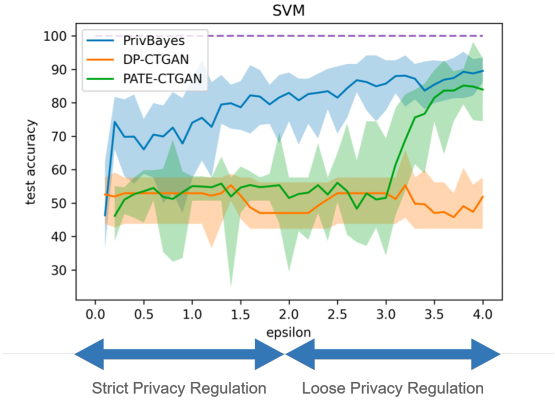

SVM
PrivBayes
DP-CTGAN
PATE-CTGAN
test accuracy
100
90
80
70
60
50
40
30
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
epsilon
Strict Privacy Regulation
Loose Privacy Regulation



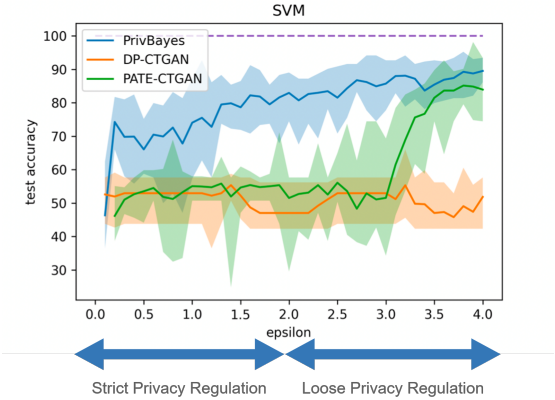
- DP-SGD makes permanent utility damage;



- DP-SGD makes permanent utility damage;
- PATE is good at loose privacy requirement;



- DP-SGD makes permanent utility damage;
- PATE is good at loose privacy requirement;
- PrivBayes is great on both loose and strict privacy requirement;



- DP-SGD makes permanent utility damage;
- PATE is good at loose privacy requirement;
- PrivBayes is great on both loose and strict privacy requirement;
- This finding persists across different classifiers, utility metrics & datasets.

Result 3: generation as imputation[6]

- Missing value is everywhere in tabular data such as EHR MIMIC-IV;

[6] with Ouyang, Xie and Li (2023)

Result 3: generation as imputation[6]

- Missing value is everywhere in tabular data such as EHR MIMIC-IV;
- Traditionally, we apply all kinds of statistical imputation methods to complete tables and possibly train downstream tasks using the imputed ones;

[6] with Ouyang, Xie and Li (2023)

Result 3: generation as imputation[6]

- Missing value is everywhere in tabular data such as EHR MIMIC-IV;
- Traditionally, we apply all kinds of statistical imputation methods to complete tables and possibly train downstream tasks using the imputed ones;
- Rather, we adapt the widely used deep generative models (learned from data with missing values) to generate complete tables.

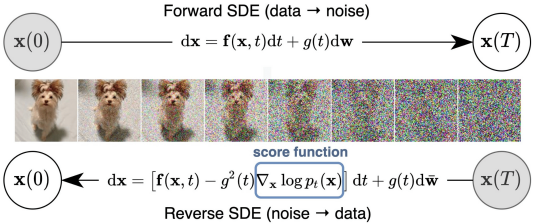
[6] with Ouyang, Xie and Li (2023)

Proposed method: MissDiff

We propose a diffusion based framework, named as MissDiff, that can be learned from *mixed* type data with missing values.

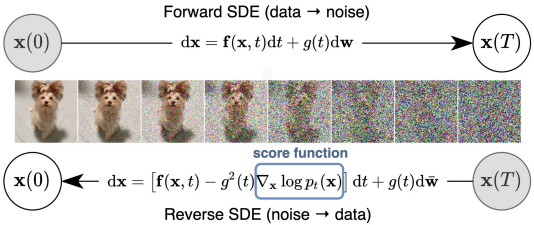
Proposed method: MissDiff

We propose a diffusion based framework, named as MissDiff, that can be learned from *mixed* type data with missing values.



Proposed method: MissDiff

We propose a diffusion based framework, named as MissDiff, that can be learned from *mixed* type data with missing values.



The mathematics behind it: reverse SDE for generation:

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - g^2(t)\nabla_{\mathbf{x}} \log p_t(\mathbf{x})\right] \mathrm{d}t + g(t)\mathrm{d}\mathbf{w}.$$

Experimental results

temperature	humidity	person	ac_on	ac_off
24.976851624413100	61.087371059280000	1	0	
	50.72691842406700	1		0
22.958243715316800		0	2	0
26.39676331897740	49.1446379155066	0		0
24.081463874895100	51.332752941686600	0		0
25.423667405985400		1	1	0
	50.20260187457130	1	0	0

(d) Raw data

temperature	humidity	person	ac_on	ac_off
24.976851624413100	61.087371059280000	1	0	1
28.491321352474500	50.72691842406700	1	2	0
22.958243715316800	55.5970516321978	0	2	0
26.39676331897740	49.1446379155066	0	2	0
24.081463874895100	51.332752941686600	0	2	0
25.423667405985400	60.01977691767880	1	1	0
21.317568123232800	50.20260187457130	1	0	0
22.515180692904800	56.77211519251470	1	0	1
26.88515290224740	55.37722194837660	1	1	1

(e) Generated data

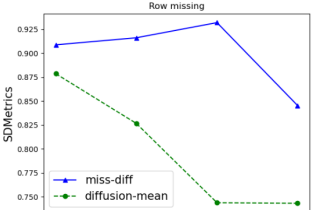
Experimental results

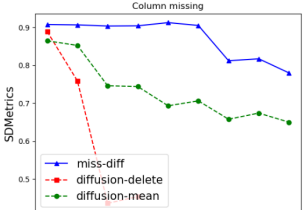
temperature	humidity	person	ac_on	ac_off
24.976851624413100	61.087371059280000	1	0	
	50.72691842406700	1		0
22.958243715316800		0	2	0
26.39676331897740	49.1446379155066	0		0
24.081463874895100	51.332752941686600	0		0
25.423667405985400		1	1	0
	50.20260187457130	1	0	0

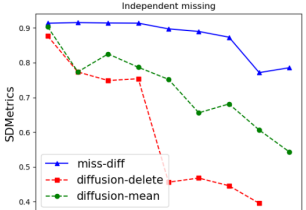
(d) Raw data

temperature	humidity	person	ac_on	ac_off
24.976851624413100	61.087371059280000	1	0	1
28.491321352474500	50.72691842406700	1	2	0
22.958243715316800	55.5970516321978	0	2	0
26.39676331897740	49.1446379155066	0	2	0
24.081463874895100	51.332752941686600	0	2	0
25.423667405985400	60.01977691767880	1	1	0
21.317568123232800	50.20260187457130	1	0	0
22.515180692904800	56.77211519251470	1	0	1
26.88515290224740	55.37722194837660	1	1	1

(e) Generated data







Experimental results

Table: Inputation results on Census dataset with RMSE criterion.

Method	RMSE
Mean /Mode	0.120
MICE(linear)	0.101
MissForest	0.112
GAIN	0.123
CSDI_T	0.099
MissDiff	**0.087**

I. Differential Privacy of Synthesizer

DP in general

- DP is a property of information-releasing systems.[7]

[7] Boudewijn et al. *NeurIPS 2023 Workshop*

DP in general

- DP is a property of information-releasing systems.[7]
- When sharing *structured/un-structured data contents*, the shared information is subject to *re-identification risk*, that is, adversary may leverage shared information and auxiliary knowledge to re-identify individual in the source real data.

[7] Boudewijn et al. *NeurIPS 2023 Workshop*

DP in general

- DP is a property of information-releasing systems.[7]
- When sharing *structured/un-structured data contents*, the shared information is subject to *re-identification risk*, that is, adversary may leverage shared information and auxiliary knowledge to re-identify individual in the source real data.
- DP serves as a standard to quantify/mitigate such re-id risk.

[7] Boudewijn et al. *NeurIPS 2023 Workshop*

DP for synthesizers

- How to define DP for synthesizers is still *controversial*.

DP for synthesizers

- How to define DP for synthesizers is still *controversial*.
- In general, DP is purpose-specific, e.g., DP for returning queried statistics.

DP for synthesizers

- How to define DP for synthesizers is still *controversial*.
- In general, DP is purpose-specific, e.g., DP for returning queried statistics.
- However, DP for synthesizers is *purpose-agnostic* since we cannot control how synthetic data will be used.

DP for synthesizers

- How to define DP for synthesizers is still *controversial*.
- In general, DP is purpose-specific, e.g., DP for returning queried statistics.
- However, DP for synthesizers is *purpose-agnostic* since we cannot control how synthetic data will be used.
- Say user 1 applies synthetic data to calculate median income of certain population; User 2 trains income predictor with synthetic data.

DP for synthesizers

- How to define DP for synthesizers is still *controversial*.
- In general, DP is purpose-specific, e.g., DP for returning queried statistics.
- However, DP for synthesizers is *purpose-agnostic* since we cannot control how synthetic data will be used.
- Say user 1 applies synthetic data to calculate median income of certain population; User 2 trains income predictor with synthetic data.
- How can we measure privacy leakage in these two scenarios in a unified manner?

Our attempt to define DP for synthesizers

- DP's original definition: for any information-releasing mechanism $\mathcal{M}$

$$\mathbb{P}(\mathcal{M}(D_0) \in S) \leq e^{\epsilon}\mathbb{P}(\mathcal{M}(D_1) \in S) + \delta$$

for any pair of *neighboring dataset* D_0, D_1.

Our attempt to define DP for synthesizers

- DP's original definition: for any information-releasing mechanism $\mathcal{M}$

$$\mathbb{P}(\mathcal{M}(D_0) \in S) \leq e^{\epsilon}\mathbb{P}(\mathcal{M}(D_1) \in S) + \delta$$

  for any pair of *neighboring dataset* D_0, D_1.
- Let $\mathcal{G}$ denote the synthesizer such as Bayesian network, GAN, diffusion, LLM, etc.

Our attempt to define DP for synthesizers

- DP's original definition: for any information-releasing mechanism $\mathcal{M}$

$$\mathbb{P}(\mathcal{M}(D_0) \in S) \leq e^{\epsilon}\mathbb{P}(\mathcal{M}(D_1) \in S) + \delta$$

  for any pair of *neighboring dataset* D_0, D_1.
- Let $\mathcal{G}$ denote the synthesizer such as Bayesian network, GAN, diffusion, LLM, etc.
- Given a real database D, we have the synthetic dataset written as

$$\widetilde{D} \sim \mathcal{G}(D).$$

Our attempt to define DP for synthesizers

- DP's original definition: for any information-releasing mechanism $\mathcal{M}$

$$\mathbb{P}(\mathcal{M}(D_0) \in S) \leq e^{\epsilon}\mathbb{P}(\mathcal{M}(D_1) \in S) + \delta$$

  for any pair of *neighboring dataset* D_0, D_1.
- Let $\mathcal{G}$ denote the synthesizer such as Bayesian network, GAN, diffusion, LLM, etc.
- Given a real database D, we have the synthetic dataset written as

$$\widetilde{D} \sim \mathcal{G}(D).$$

- Intuition: if a generator $\mathcal{G}$ is DP, then it should be difficult for users to infer from its synthetic data whether $\mathcal{G}$ was trained from D_0 or D_1 (w/o auxiliary information).

DP in downstream tasks of synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote the synthetic dataset generated from the generator $\mathcal{G}$ trained on D_0 and D_1, respectively.

DP in downstream tasks of synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote the synthetic dataset generated from the generator $\mathcal{G}$ trained on D_0 and D_1, respectively.
- Example 1: if the median income $m(D)$ is of interest, can we show

$$\mathbb{P}\left(m\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(m\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

DP in downstream tasks of synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote the synthetic dataset generated from the generator $\mathcal{G}$ trained on D_0 and D_1, respectively.
- Example 1: if the median income $m(D)$ is of interest, can we show

$$\mathbb{P}\left(m\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(m\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

- Example 2: if training an income predictor $\theta(\widetilde{D})$, can we show

$$\mathbb{P}\left(\theta\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(\theta\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

DP in downstream tasks of synthetic data

- Let $\widetilde{D}_0 \sim \mathcal{G}(D_0)$ and $\widetilde{D}_1 \sim \mathcal{G}(D_1)$ denote the synthetic dataset generated from the generator $\mathcal{G}$ trained on D_0 and D_1, respectively.
- Example 1: if the median income $m(D)$ is of interest, can we show

$$\mathbb{P}\left(m\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(m\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

- Example 2: if training an income predictor $\theta(\widetilde{D})$, can we show

$$\mathbb{P}\left(\theta\left(\widetilde{D}_0\right) \in S\right) \leq e^{\epsilon}\mathbb{P}\left(\theta\left(\widetilde{D}_1\right) \in S\right) + \delta?$$

- To answer both questions, we need to have a deep understanding on the design of $\mathcal{G}$ which induces the probability measure $\mathbb{P}$.

Why synthesizer's DP is still not there yet?

- Ideally, the definition of DP for synthesizers is *generator-agnostic* and *downstream task-agnostic* in the sense that DP depends on neither generators nor downstream tasks.

Why synthesizer's DP is still not there yet?

- Ideally, the definition of DP for synthesizers is *generator-agnostic* and *downstream task-agnostic* in the sense that DP depends on neither generators nor downstream tasks.
- Of course, we can always define synthesizer's DP in a narrow sense if we have full knowledge on generator and downstream tasks.

Why synthesizer's DP is still not there yet?

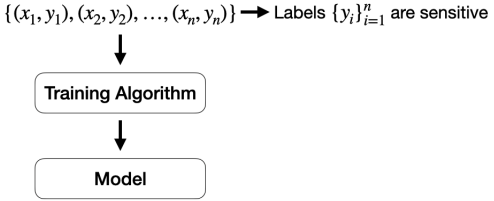
- Ideally, the definition of DP for synthesizers is *generator-agnostic* and *downstream task-agnostic* in the sense that DP depends on neither generators nor downstream tasks.
- Of course, we can always define synthesizer's DP in a narrow sense if we have full knowledge on generator and downstream tasks.
- To have a general purpose DP, we may approach it from a *privacy auditing* perspective that gives a lower bound on ϵ.

Another example: Achieve (Local) Label DP via Randomizing Response

Xu, Wang, Sun and C. (*TMLR*, 2023)

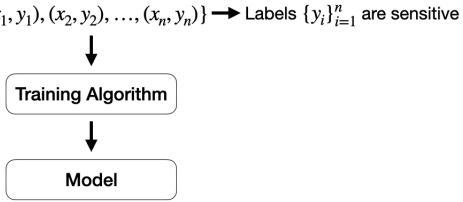
Learning with Label DP

- Learning from a dataset where labels are sensitive



Learning with Label DP

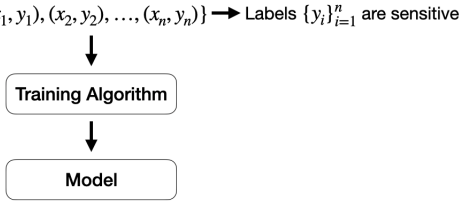
- Learning from a dataset where labels are sensitive



- In survey, basic demographic info is non-sensitive but income is sensitive; In Advertisement, impression data is non-sensitive, but conversion rate is sensitive;

Learning with Label DP

- Learning from a dataset where labels are sensitive

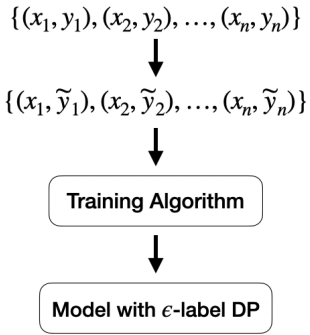


- In survey, basic demographic info is non-sensitive but income is sensitive; In Advertisement, impression data is non-sensitive, but conversion rate is sensitive;
- Label Differentially Privacy: For any two datasets D_1 and D_2 differing in a single label, then a randomized mechanism satisfies ϵ-label DP if

$$\frac{\mathbb{P}(\mathcal{A}(D) \in S)}{\mathbb{P}(\mathcal{A}(D') \in S)} \leq \exp(\epsilon)$$

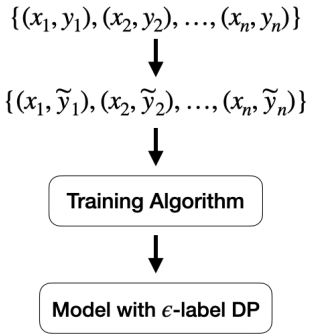
Learning with Private Labels

- Learning with private labels



Learning with Private Labels

- Learning with private labels



- We need to perturb labels $\widetilde{y}_i$'s in a proper way to preserve ϵ-label differentially private.

Privacy-Protection of Randomized Response

- For the binary label, we can flip it randomly as (Warner, 1965)

$$\mathcal{A}_\theta(Y) = \begin{cases} Y, & \text{with probability } \theta, \\ -Y, & \text{with probability } 1-\theta, \end{cases}$$

Privacy-Protection of Randomized Response

- For the binary label, we can flip it randomly as (Warner, 1965)

$$\mathcal{A}_\theta(Y) = \begin{cases} Y, & \text{with probability } \theta, \\ -Y, & \text{with probability } 1-\theta, \end{cases}$$

- It is straightforward to verify that this randomized response mechanism satisfies ϵ-label DP if setting

$$\theta = \frac{e^\epsilon}{1+e^\epsilon}.$$

Differentially Private Classifier

- A dataset with privatized labels $\{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ can be used for learning a differentially private classifier:

$$\widetilde{f}_n = _{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \phi(f(\boldsymbol{x}_i)\widetilde{y}_i) + \lambda_n J(f),$$

where ϕ is any margin loss function and $J(f)$ is a regularization term.

Differentially Private Classifier

- A dataset with privatized labels $\{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ can be used for learning a differentially private classifier:

$$\widetilde{f}_n = _{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \phi(f(\boldsymbol{x}_i)\widetilde{y}_i) + \lambda_n J(f),$$

  where ϕ is any margin loss function and $J(f)$ is a regularization term.
- We evaluate the performance of the private classifier via

$$R(\widetilde{f}_n) = \mathbb{E}\left[I(\widetilde{f}_n(\boldsymbol{X}) \neq Y) \right].$$

Differentially Private Classifier

- A dataset with privatized labels $\{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ can be used for learning a differentially private classifier:

$$\widetilde{f}_n = {}_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \phi(f(\boldsymbol{x}_i)\widetilde{y}_i) + \lambda_n J(f),$$

  where ϕ is any margin loss function and $J(f)$ is a regularization term.
- We evaluate the performance of the private classifier via

$$R(\widetilde{f}_n) = \mathbb{E}\left[I(\widetilde{f}_n(\boldsymbol{X}) \neq Y) \right].$$

- Interestingly, we prove that $R(\widetilde{f}_n) \longrightarrow R(f^*)$ as long as $\epsilon^2 \approx \frac{\log n}{n}$ with $R(f^*)$ being the Bayes risk.

Application to Deep Neural Network Classifier

Theorem

Let $\mathcal{P}_{\gamma,\beta}$ be a class of probability measures on $\mathcal{X} \times \{-1, 1\}$ under certain assumptions. For a deep neural network plug-in classifier $\widetilde{s}_{nn}$ with layers $\asymp \log(\kappa_\epsilon n/\log(n))$ and hidden units $\asymp (\kappa_\epsilon n/\log(n))^{\frac{2p}{2\beta+p}}$, we have

$$\left(\frac{1}{n\kappa_\epsilon^2}\right)^{\frac{\beta(\gamma+1)}{\beta(\gamma+2)+p}} \lesssim \sup_{\pi\in\mathcal{P}_{\gamma,\beta}} \mathbb{E}\left[R(\widetilde{s}_{nn}) - R(f^*)\right] \lesssim \left(\frac{\log n}{n\kappa_\epsilon^2}\right)^{\frac{2\beta(\gamma+1)}{2\beta(\gamma+2)+p(\gamma+2)}},$$

where $\kappa_\epsilon = \frac{e^\epsilon-1}{e^\epsilon+1}$

Application to Deep Neural Network Classifier

Theorem

Let $\mathcal{P}_{\gamma,\beta}$ be a class of probability measures on $\mathcal{X} \times \{-1, 1\}$ under certain assumptions. For a deep neural network plug-in classifier $\widetilde{s}_{nn}$ with layers $\asymp \log(\kappa_\epsilon n/\log(n))$ and hidden units $\asymp (\kappa_\epsilon n/\log(n))^{\frac{2p}{2\beta+p}}$, we have

$$\left(\frac{1}{n\kappa_\epsilon^2}\right)^{\frac{\beta(\gamma+1)}{\beta(\gamma+2)+p}} \lesssim \sup_{\pi\in\mathcal{P}_{\gamma,\beta}} \mathbb{E}\left[R(\widetilde{s}_{nn}) - R(f^*)\right] \lesssim \left(\frac{\log n}{n\kappa_\epsilon^2}\right)^{\frac{2\beta(\gamma+1)}{2\beta(\gamma+2)+p(\gamma+2)}},$$

where $\kappa_\epsilon = \frac{e^\epsilon-1}{e^\epsilon+1}$

An important implication: as ϵ decreases (more privacy), the optimal neural network requires less layers and hidden units.

Experiments

- $\widetilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ is constructed with $\boldsymbol{x}_i$ being uniformly generated and y_i being generated via (so that $\theta = e^\epsilon/(1+e^\epsilon)$)

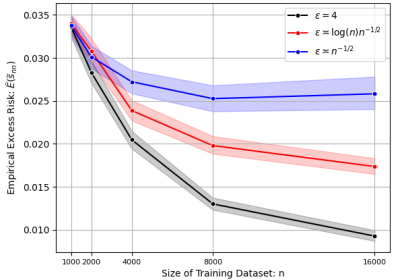
$$y_i = \begin{cases} 1 & \text{with probability } \sum_{j=1}^4 \sin(2\pi x_{ij})/8 + 1/2, \\ -1 & \text{with probability } 1/2 - \sum_{j=1}^4 \sin(2\pi x_{ij})/8. \end{cases}$$

Experiments

- $\widetilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ is constructed with $\boldsymbol{x}_i$ being uniformly generated and y_i being generated via (so that $\theta = e^\epsilon/(1+e^\epsilon)$)

$$y_i = \begin{cases} 1 & \text{with probability } \sum_{j=1}^4 \sin(2\pi x_{ij})/8 + 1/2, \\ -1 & \text{with probability } 1/2 - \sum_{j=1}^4 \sin(2\pi x_{ij})/8. \end{cases}$$

- In DNN classifiers, we empirically confirm our theory that $\epsilon \approx n^{-1/2}$ is a critical point in determining the convergence of excess risk.



Potential application I: ads measurement data

FORBES > INNOVATION > ENTERPRISE TECH

Google Commits To Third-Party Cookies Deprecation In 2024

Forrester Contributor ⓘ

Follow

0

Jan 16, 2024, 01:43pm EST

Today, Google began rolling out Tracking Protection, a new feature that limits the use of cross-site tracking by deprecating third-party cookies by default. Part of Google's broader Privacy Sandbox initiative, this change affects 1% of Chrome users globally — about 30 million people. And it's a milestone that many thought might never happen: Per Forrester, 51% of the global marketers we surveyed **did not believe** that Google would deprecate the third-party cookie.

Potential application I: ads measurement data

FORBES > INNOVATION > ENTERPRISE TECH

Google Commits To Third-Party Cookies Deprecation In 2024

Forrester Contributor ⓘ

Follow

0

Jan 16, 2024, 01:43pm EST

Today, Google began rolling out Tracking Protection, a new feature that limits the use of cross-site tracking by deprecating third-party cookies by default. Part of Google's broader Privacy Sandbox initiative, this change affects 1% of Chrome users globally — about 30 million people. And it's a milestone that many thought might never happen: Per Forrester, 51% of the global marketers we surveyed **did not believe** that Google would deprecate the third-party cookie.

- GDPR rules websites can no longer rely on *implicit* opt-in, and must capture opt-in consent before any analytics or web tracking cookies are placed on a browser.

Potential application I: ads measurement data

FORBES > INNOVATION > ENTERPRISE TECH

Google Commits To Third-Party Cookies Deprecation In 2024

Forrester Contributor ⓘ

Follow

Jan 16, 2024, 01:43pm EST

Today, Google began rolling out Tracking Protection, a new feature that limits the use of cross-site tracking by deprecating third-party cookies by default. Part of Google's broader Privacy Sandbox initiative, this change affects 1% of Chrome users globally — about 30 million people. And it's a milestone that many thought might never happen: Per Forrester, 51% of the global marketers we surveyed **did not believe** that Google would deprecate the third-party cookie.

- GDPR rules websites can no longer rely on *implicit* opt-in, and must capture opt-in consent before any analytics or web tracking cookies are placed on a browser.
- Digital Markets Act (DMA) imposes restrictions on online advertising services owned by large digital corporations.

Potential application I: ads measurement data

FORBES > INNOVATION > ENTERPRISE TECH

Google Commits To Third-Party Cookies Deprecation In 2024

Forrester Contributor ⓘ

Follow

Jan 16, 2024, 01:43pm EST

Today, Google began rolling out Tracking Protection, a new feature that limits the use of cross-site tracking by deprecating third-party cookies by default. Part of Google's broader Privacy Sandbox initiative, this change affects 1% of Chrome users globally — about 30 million people. And it's a milestone that many thought might never happen: Per Forrester, 51% of the global marketers we surveyed **did not believe** that Google would deprecate the third-party cookie.

- GDPR rules websites can no longer rely on *implicit* opt-in, and must capture opt-in consent before any analytics or web tracking cookies are placed on a browser.
- Digital Markets Act (DMA) imposes restrictions on online advertising services owned by large digital corporations.
- Under these regulations, advertisement data becomes unavoidably fragmented, having huge impacts on marketers for their data analytics and machine learning tasks.

Potential application I: ads measurement data

FORBES > INNOVATION > ENTERPRISE TECH

Google Commits To Third-Party Cookies Deprecation In 2024

Forrester Contributor ⓘ

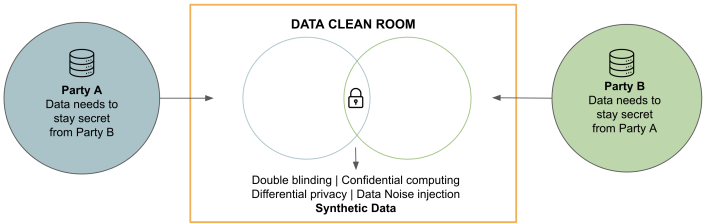
Follow

Jan 16, 2024, 01:43pm EST

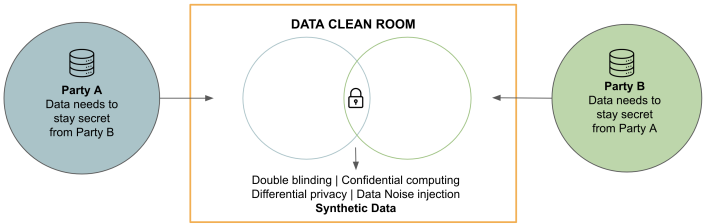
Today, Google began rolling out Tracking Protection, a new feature that limits the use of cross-site tracking by deprecating third-party cookies by default. Part of Google's broader Privacy Sandbox initiative, this change affects 1% of Chrome users globally — about 30 million people. And it's a milestone that many thought might never happen: Per Forrester, 51% of the global marketers we surveyed **did not believe** that Google would deprecate the third-party cookie.

- GDPR rules websites can no longer rely on *implicit* opt-in, and must capture opt-in consent before any analytics or web tracking cookies are placed on a browser.
- Digital Markets Act (DMA) imposes restrictions on online advertising services owned by large digital corporations.
- Under these regulations, advertisement data becomes unavoidably fragmented, having huge impacts on marketers for their data analytics and machine learning tasks.
- Privacy compliant synthetic data is a promising solution to mitigate such impact on marketing industry.

Potential application II: data clean room

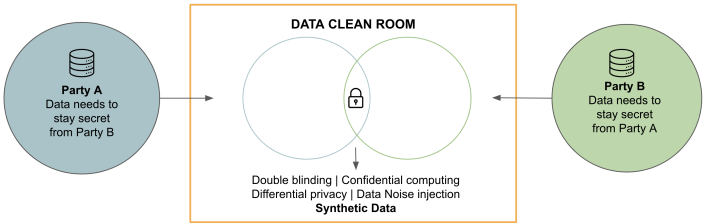


Potential application II: data clean room



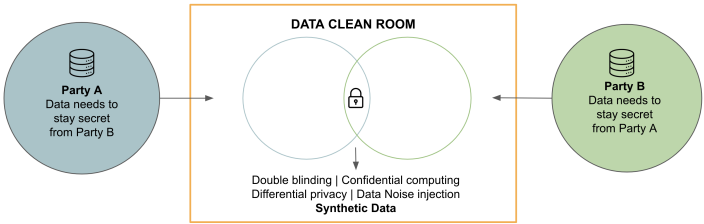
- A Data Clean Room (DCR) is a secure environment that allows multiple parties to collaboratively use shared (aggregate) data without exposing personal customer information.

Potential application II: data clean room



- A Data Clean Room (DCR) is a secure environment that allows multiple parties to collaboratively use shared (aggregate) data without exposing personal customer information.
- Classical DCR techniques include double blinding, confidential computing, differential privacy, data noise injection etc.

Potential application II: data clean room



- A Data Clean Room (DCR) is a secure environment that allows multiple parties to collaboratively use shared (aggregate) data without exposing personal customer information.
- Classical DCR techniques include double blinding, confidential computing, differential privacy, data noise injection etc.
- Synthetic data could be an alternative DCR technique solution that allows access to individual (synthetic) data with best utility.


GRAZAS
DANKIE
QUJAN
GRAZAS
MERCI XIE XIE
THANK YOU
THANK YOU DANKIE
THANKS
TAKK
ASANTE
GRAZAS
MAHALO
THANK YOU
GRACIAS
OBRIGADO APACHE
TODA
MERCI
OBRIGADA
THANK YOU
GRACIAS
OBRIGADO
MUCHAS GRACIAS
YEWO
DANKIE
TAKK
THANK YOU
QUIANARSUAQ
TODA
ASANTE
TACK
MITELA
XIE XIE
TAKK
XIE XIE
MITELA
TODA
KIITOS
SPASIBO
ДЯКУЮ
DANKE
ДЯКУЮ
HVALA
GRAZIE
DEKUJI
TAKK
TODA
KIITOS
FOOLE TUNK
THANK YOU
DANKIE
SPAS
ASANTE
WILIWNI
BAIE DANKIE
NGIYABONGA
FAKAAUE
NGIYABONGA
HVALA
SUKRIA
TAKK
PAXMAT
СПАСИБО
TESEKKÜR EDERIM
ДЯКУЙ
GRAZAS
THANK YOU
PAXMET!
多謝
唔該
DO JEH
SPAS
XIE XIE
SUKRIA
ASANTE
KOP KUR
SPASIBI
SUKRIA
XIE XIE
TODA
THANK YOU
多謝
TA