Diffusion Models Bootcamp

For Single-Table Tabular Datasets

(Part 1)

Applied AI Projects

August 7, 2024

VECTOR INSTITUTE

Agenda

1 What is Tabular Data?

2 Synthetic Tabular Data

3 Diffusion Model for Tabular Data

4 Synthetic Data Evaluation

What is Tabular Data?

VECTOR INSTITUTE

- **Everything in a database or spreadsheet**
  - Data can be represented as columns and row
- **Contains mixed type of data**
  - Numerical
  - Categorical
  - Dates
  - Text

	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3	PAY_4	PAY_5	...	BILL_AMT4	BILL_AMT5	BILL_AMT6	PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4
0	20000.0	female	university	married	24.0	payment delay for three months	payment delay for three months	payment delay for one month	payment delay for one month	payment delay for one month	...	18457.0	21381.0	18914.0	0.0	1500.0	1600.0	1646.0
1	200000.0	female	university	married	39.0	payment delay for four months	payment delay for three months	payment delay for three months	payment delay for three months	payment delay for three months	...	125357.0	121853.0	124731.0	0.0	6216.0	10000.0	0.0
2	230000.0	female	university	single	23.0	pay duly	pay duly	pay duly	pay duly	pay duly	...	1045.0	12525.0	12219.0	1444.0	14019.0	1045.0	12525.0
3	50000.0	female	graduate school	married	35.0	payment delay for one month	payment delay for one month	payment delay for one month	unknown	unknown	...	0.0	0.0	0.0	2400.0	0.0	0.0	0.0
4	160000.0	male	graduate school	married	39.0	unknown	unknown	unknown	unknown	unknown	...	0.0	2920.0	0.0	35.0	0.0	0.0	2920.0
5	20000.0	male	high school	others	59.0	payment delay for three months	payment delay for one month	payment delay for one month	payment delay for one month	payment delay for one month	...	18055.0	18755.0	20299.0	1596.0	1600.0	1300.0	1000.0
6	50000.0	male	university	single	42.0	payment delay for one month	payment delay for one month	payment delay for one month	payment delay for one month	payment delay for one month	...	17029.0	10575.0	9478.0	2500.0	2000.0	2500.0	500.0
7	130000.0	female	graduate school	single	26.0	payment delay for one month	pay duly	pay duly	pay duly	unknown	...	-884.0	-6332.0	-9333.0	1298.0	6730.0	900.0	5448.0

Source: https://github.com/SanDiegoMachineLearning/talks

What is Tabular Data?

VECTOR INSTITUTE

- **Everything in a database or spreadsheet**
  - Data can be represented as columns and row
- **Contains mixed type of data**
  - **Numerical**
  - **Categorical**
  - Dates
  - Text

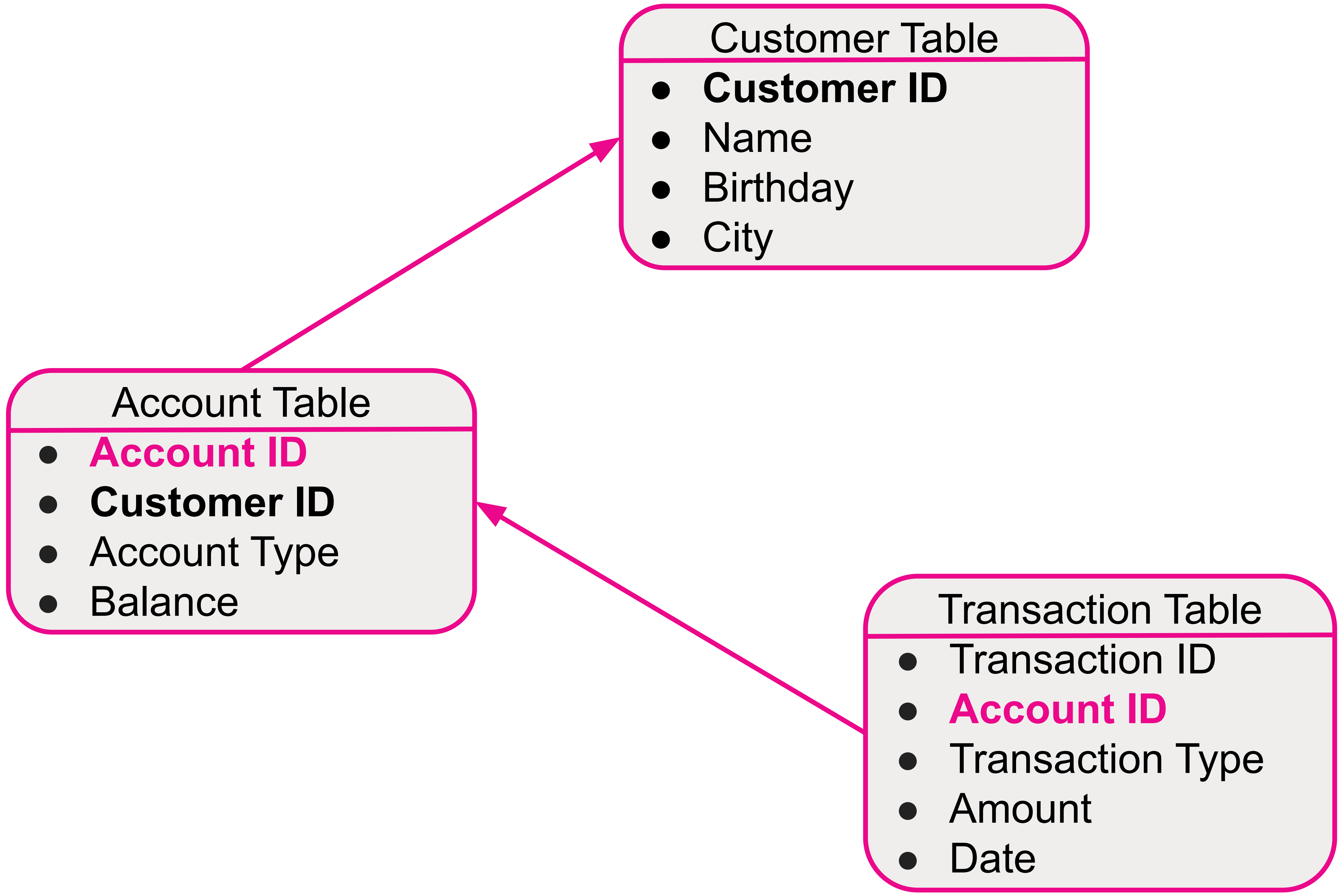
	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3	PAY_4	PAY_5	...	BILL_AMT4	BILL_AMT5	BILL_AMT6	PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4
0	20000.0	female	university	married	24.0	payment delay for three months	payment delay for three months	payment delay for one month	payment delay for one month	payment delay for one month	...	18457.0	21381.0	18914.0	0.0	1500.0	1600.0	1646.0
1	200000.0	female	university	married	39.0	payment delay for four months	payment delay for three months	payment delay for three months	payment delay for three months	payment delay for three months	...	125357.0	121853.0	124731.0	0.0	6216.0	10000.0	0.0
2	230000.0	female	university	single	23.0	pay duly	pay duly	pay duly	pay duly	pay duly	...	1045.0	12525.0	12219.0	1444.0	14019.0	1045.0	12525.0
3	50000.0	female	graduate school	married	35.0	payment delay for one month	payment delay for one month	payment delay for one month	unknown	unknown	...	0.0	0.0	0.0	2400.0	0.0	0.0	0.0
4	160000.0	male	graduate school	married	39.0	unknown	unknown	unknown	unknown	unknown	...	0.0	2920.0	0.0	35.0	0.0	0.0	2920.0
5	20000.0	male	high school	others	59.0	payment delay for three months	payment delay for one month	payment delay for one month	payment delay for one month	payment delay for one month	...	18055.0	18755.0	20299.0	1596.0	1600.0	1300.0	1000.0
6	50000.0	male	university	single	42.0	payment delay for one month	payment delay for one month	payment delay for one month	payment delay for one month	payment delay for one month	...	17029.0	10575.0	9478.0	2500.0	2000.0	2500.0	500.0
7	130000.0	female	graduate school	single	26.0	payment delay for one month	pay duly	pay duly	pay duly	unknown	...	-884.0	-6332.0	-9333.0	1298.0	6730.0	900.0	5448.0

Source: https://github.com/SanDiegoMachineLearning/talks

Tabular Data Special Cases

VECTOR INSTITUTE

- **Information that is across many tables → Multi-Relational Dataset**
  - The relationships between these entities are captured through **foreign keys**, establishing connections between records in different tables.
  - Such as Individual transactions that need to be grouped to be meaningful



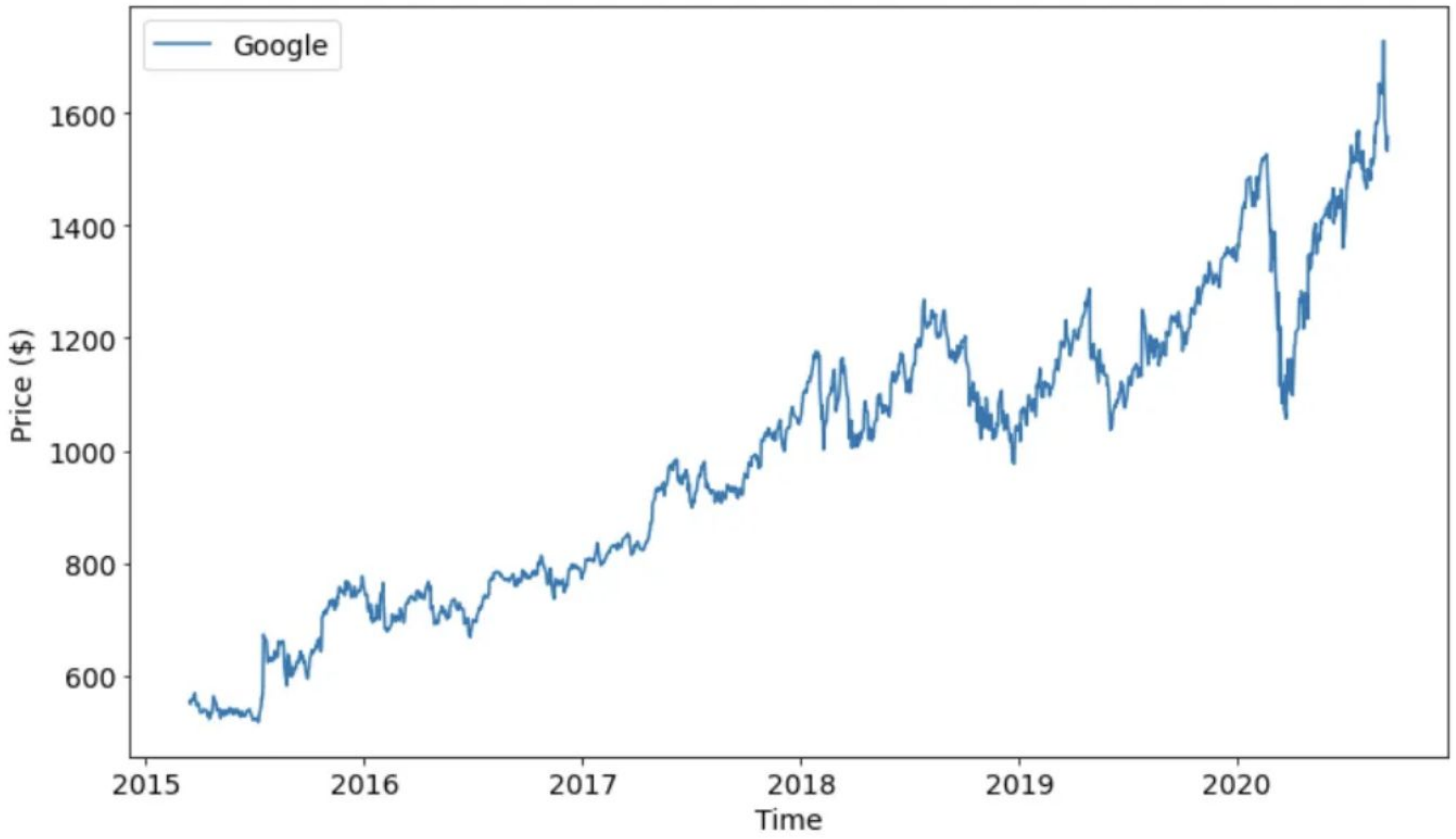
Will Be Covered in Tomorrow's Morning Session

Source: https://github.com/SanDiegoMachineLearning/talks

Tabular Data Special Cases

- **Information that is gathered along time → Time Series Dataset**
  - **Univariate time series:** A dataset with a single variable recorded over time.
  - **Multivariate time series:** A dataset with multiple variables recorded over time, where the variables may be interdependent.
  - Such as stock prices and weather

Will Be Covered in Tomorrow's Afternoon Session



Source: https://github.com/SanDiegoMachineLearning/talks

Synthetic Tabular Dataset

VECTOR INSTITUTE

- **What is tabular synthetic data?**
  - Artificially generated data that imitates real-world tabular data for testing and training models.

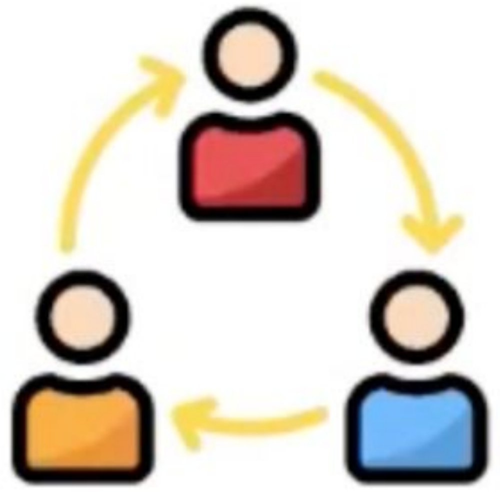
Source: https://github.com/SanDiegoMachineLearning/talks

Synthetic Tabular Dataset

VECTOR INSTITUTE

- **What is tabular synthetic data?**
  - Artificially generated data that imitates real-world tabular data for testing and training models.

- **Why synthetic data?**



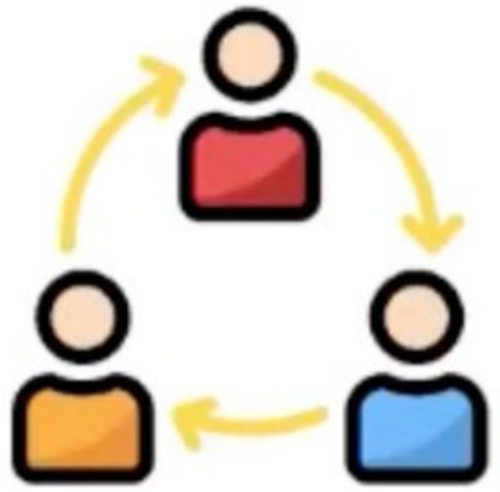
Source: https://www.youtube.com/live/IzMfWkWNtmA?si=5Eeq_LYhDq-MK8wk

Synthetic Tabular Dataset

VECTOR INSTITUTE

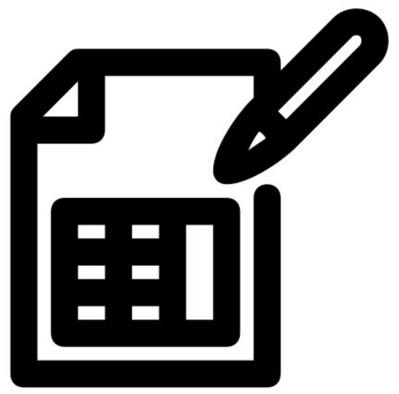
- **What is tabular synthetic data?**
  - Artificially generated data that imitates real-world tabular data for testing and training models.

- **Why synthetic data?**



Lack of Data

Imbalance Data

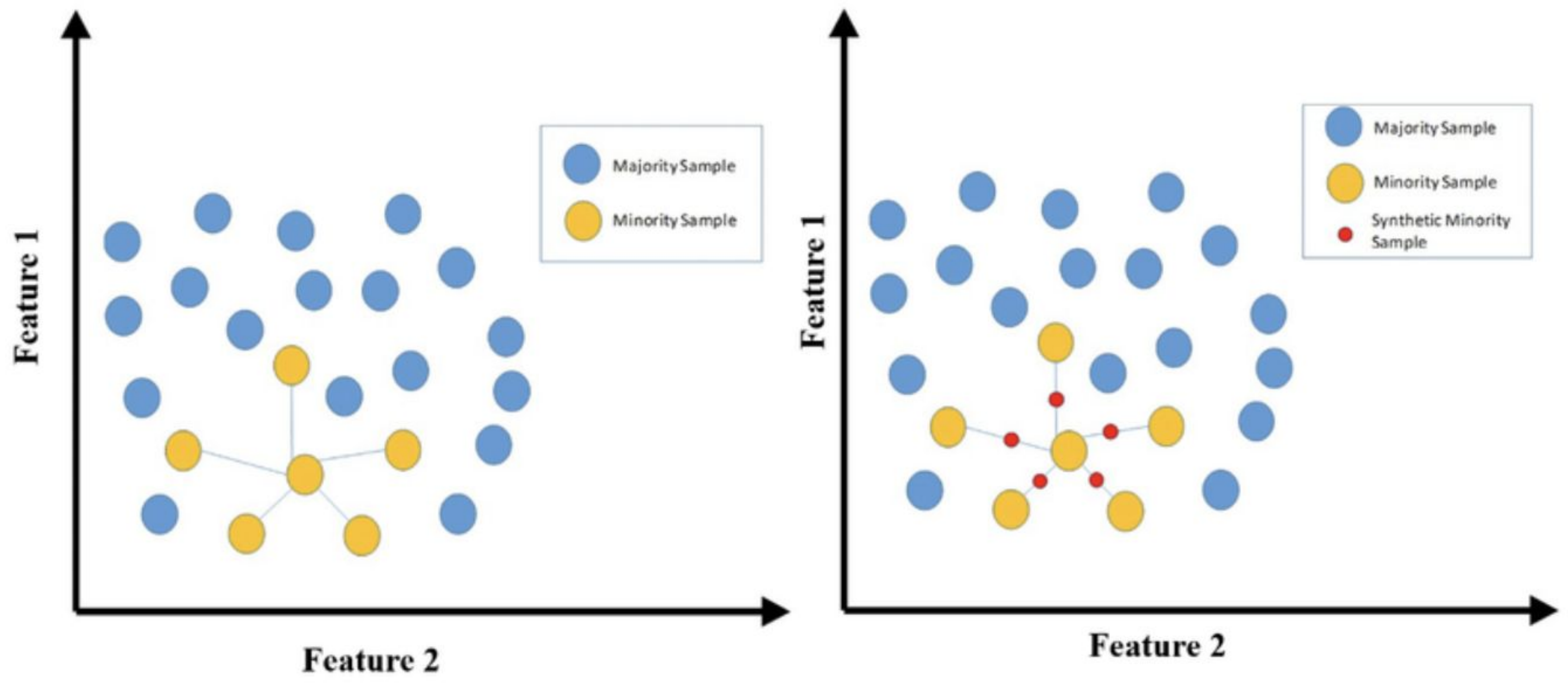


Missing Data

Source: https://www.youtube.com/live/IzMfWkWNtmA?si=5Eeq_LYhDq-MK8wk

Tabular Data Synthesis Methods

- **Traditional Generation**

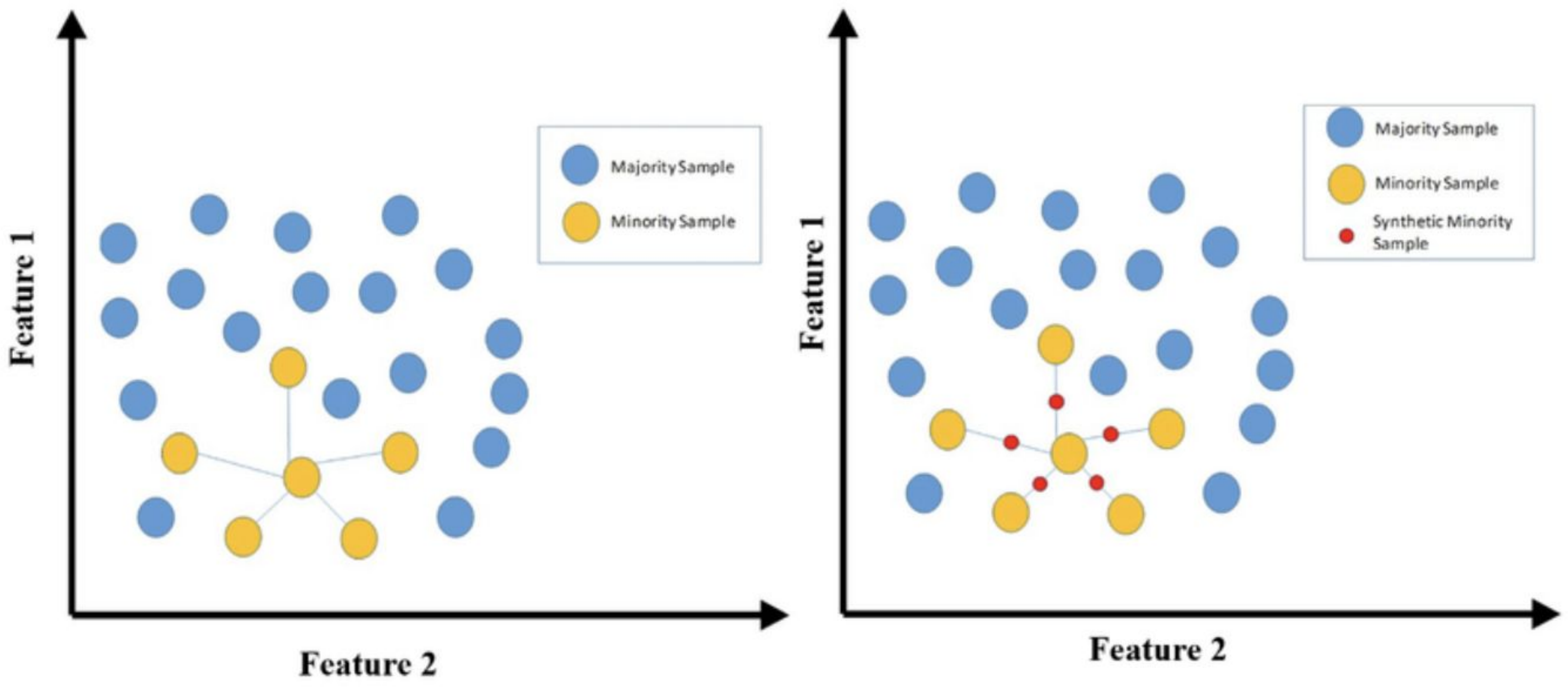


Oversampling Methods (Ex. SMOTE)

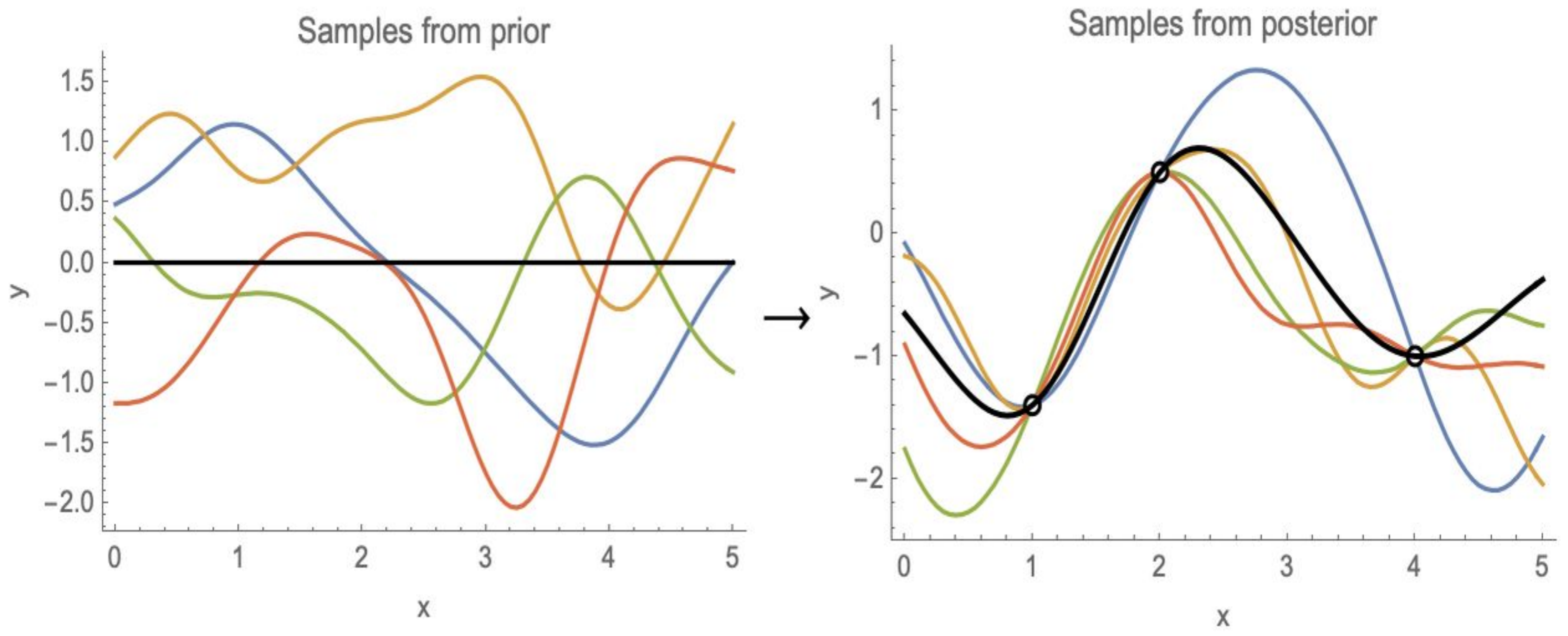
Source: https://www.wolfram.com/language/introduction-machine-learning/bayesian-inference/
Source: https://www.researchgate.net/figure/Illustration-of-the-SMOTE-oversampling-approach_fig3_347937180

Tabular Data Synthesis Methods

- **Traditional Generation**



Oversampling Methods (Ex. SMOTE)

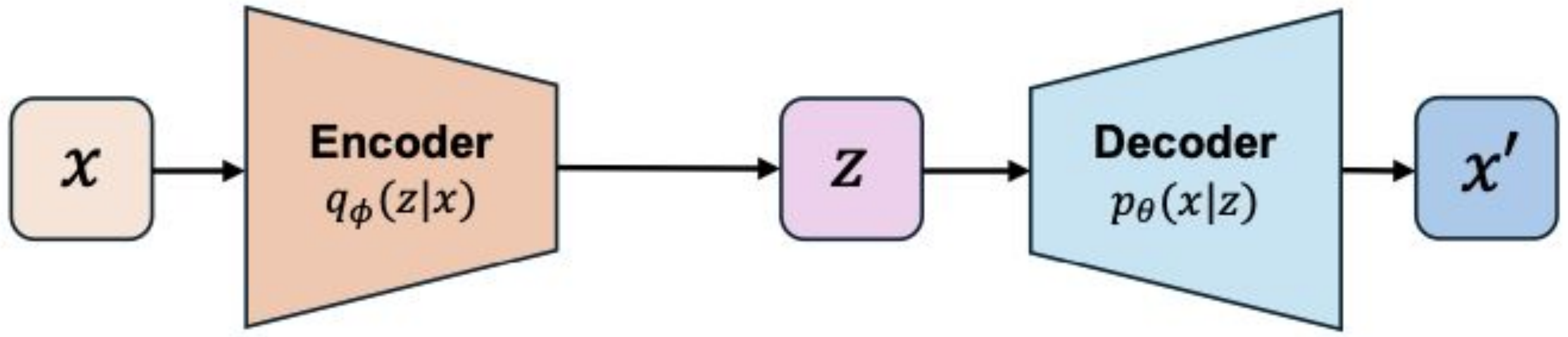


Multivariate Statistical Methods (Ex. Bayesian Networks)

Source: https://www.wolfram.com/language/introduction-machine-learning/bayesian-inference/
Source: https://www.researchgate.net/figure/Illustration-of-the-SMOTE-oversampling-approach_fig3_347937180

Tabular Data Synthesis Methods

- **Deep Learning-Based Generation**

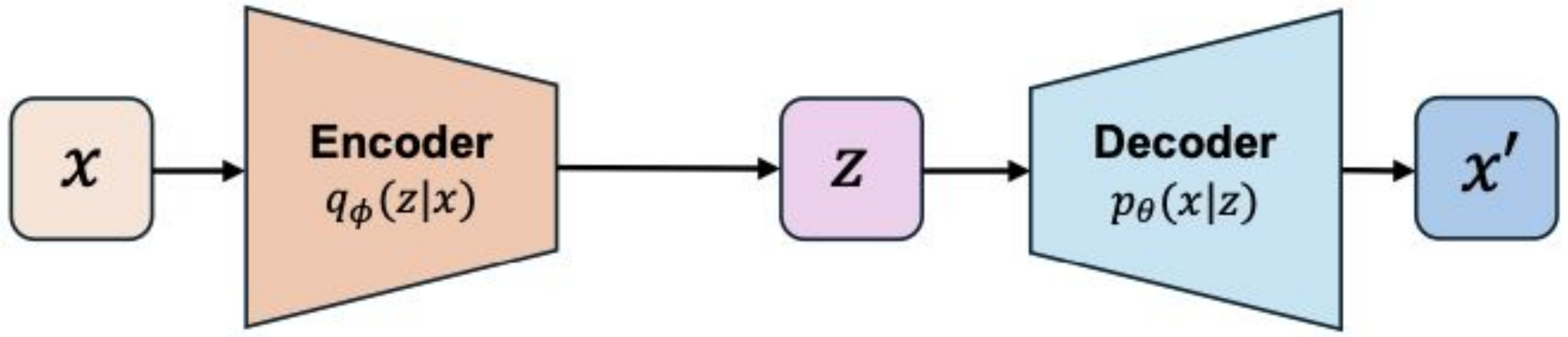


VAEs (Ex. TVAE, GOGGLE)

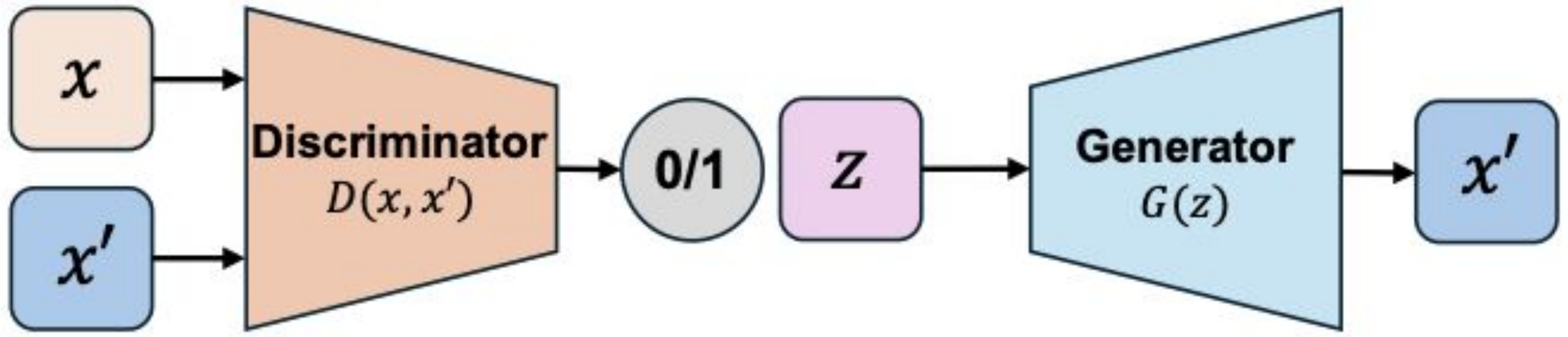
Source: https://github.com/SanDiegoMachineLearning/talks

Tabular Data Synthesis Methods

- **Deep Learning-Based Generation**



VAEs (Ex. TVAE, GOGGLE)



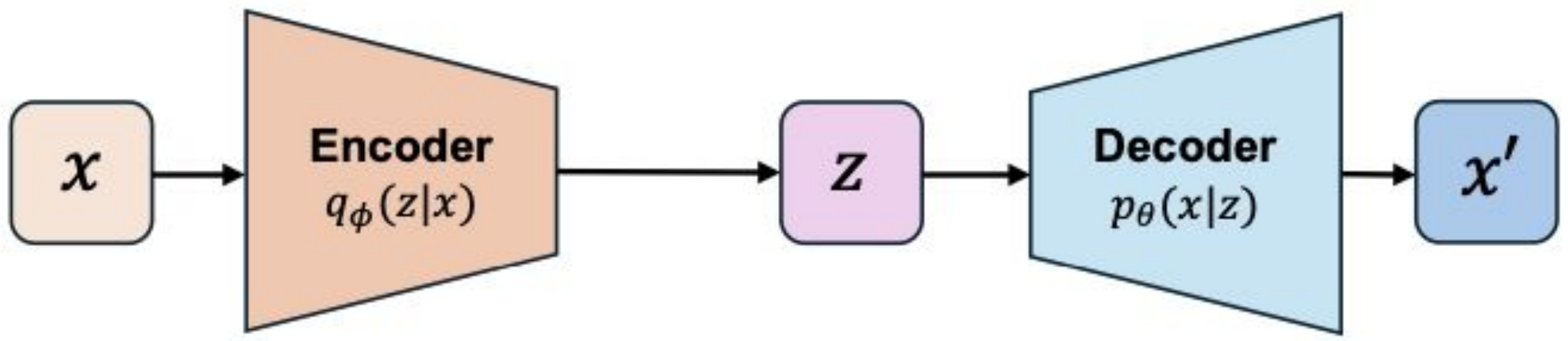
GANs (Ex. CTGAN)

Source: https://github.com/SanDiegoMachineLearning/talks

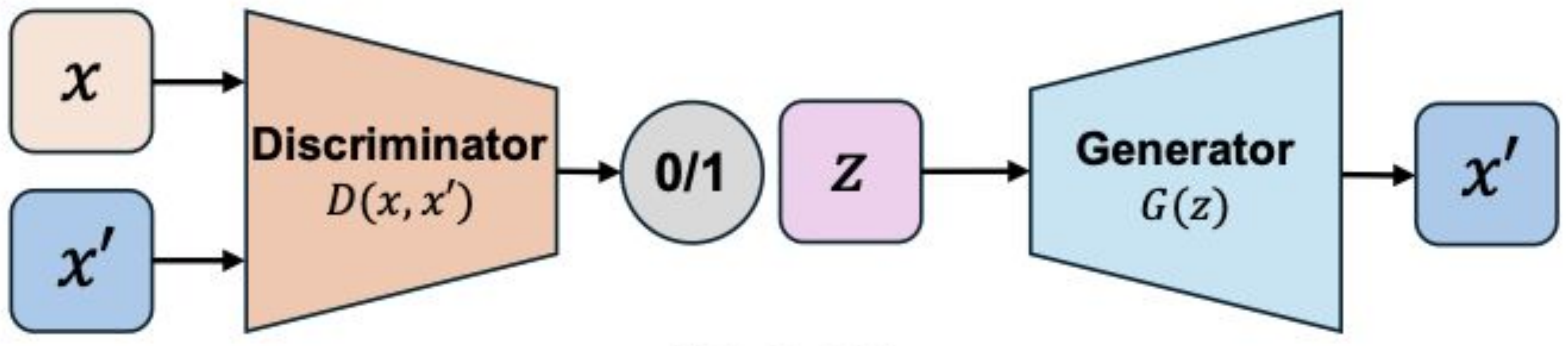
Tabular Data Synthesis Methods

VECTOR INSTITUTE

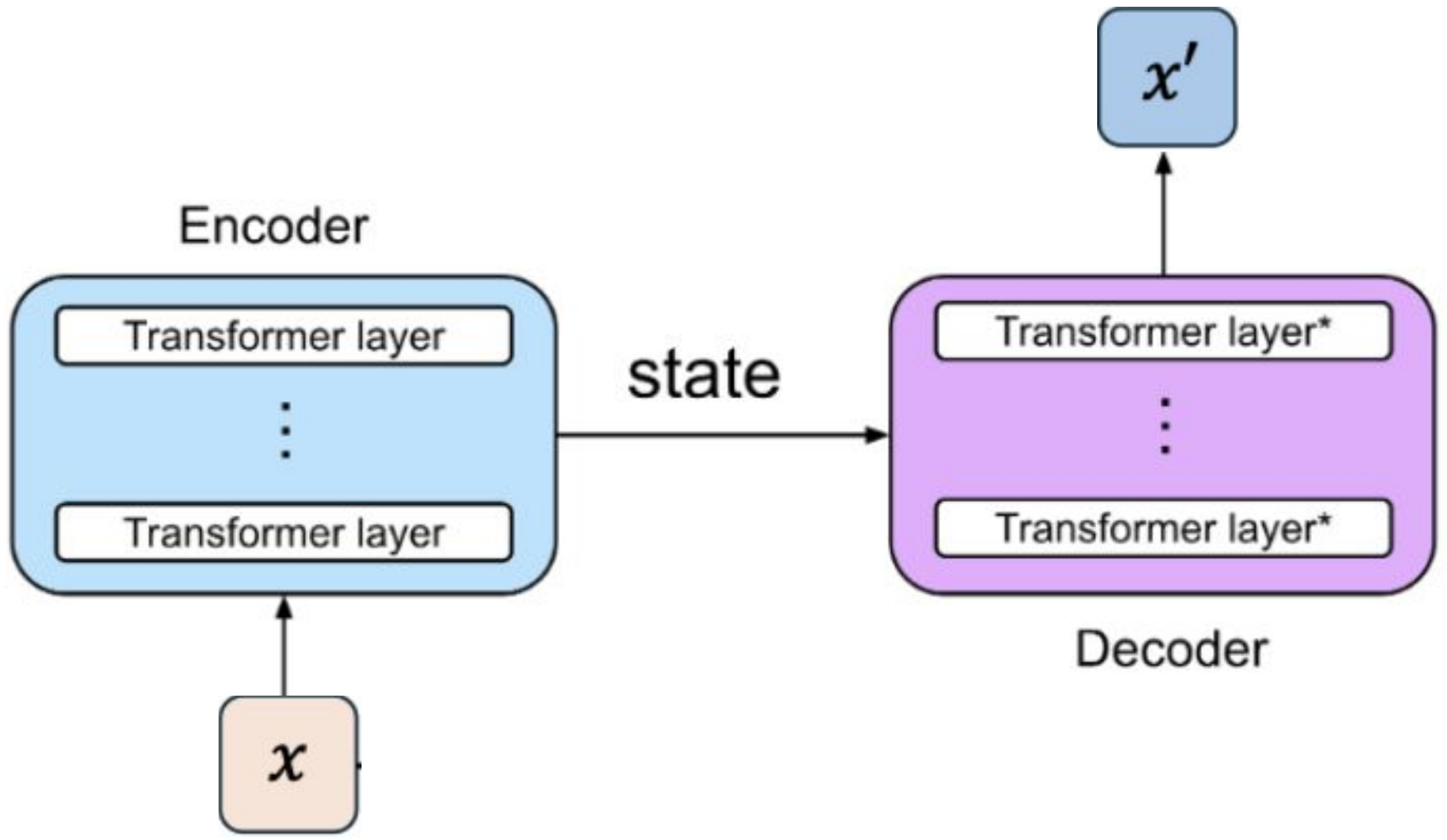
- **Deep Learning-Based Generation**



VAEs (Ex. TVAE, GOGGLE)



GANs (Ex. CTGAN)

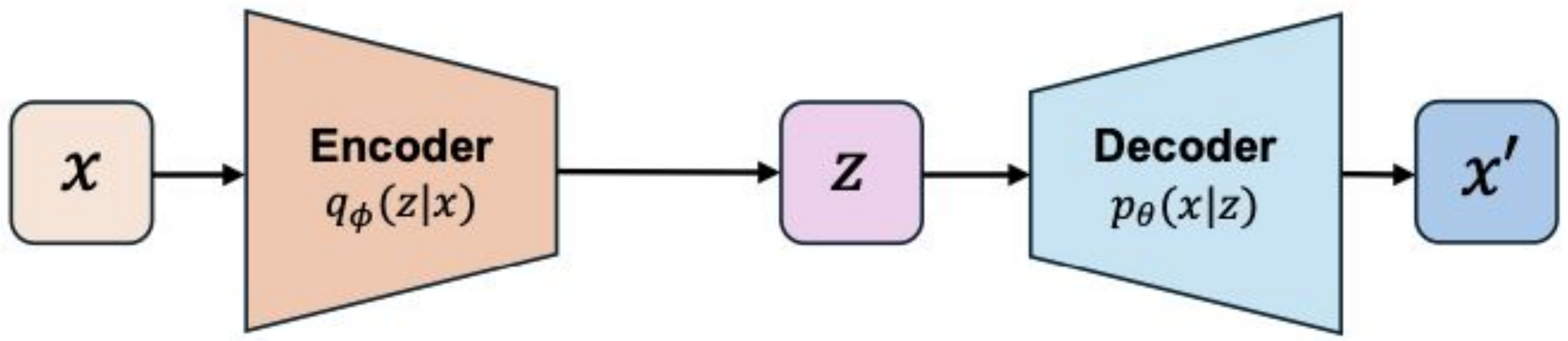


Transformers (Ex.GReaT)

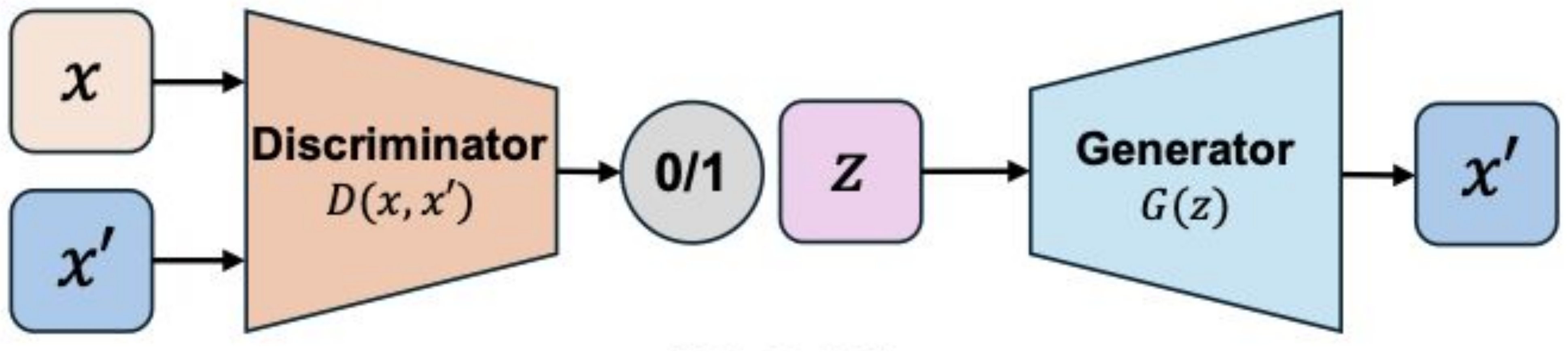
Source: https://github.com/SanDiegoMachineLearning/talks

Tabular Data Synthesis Methods

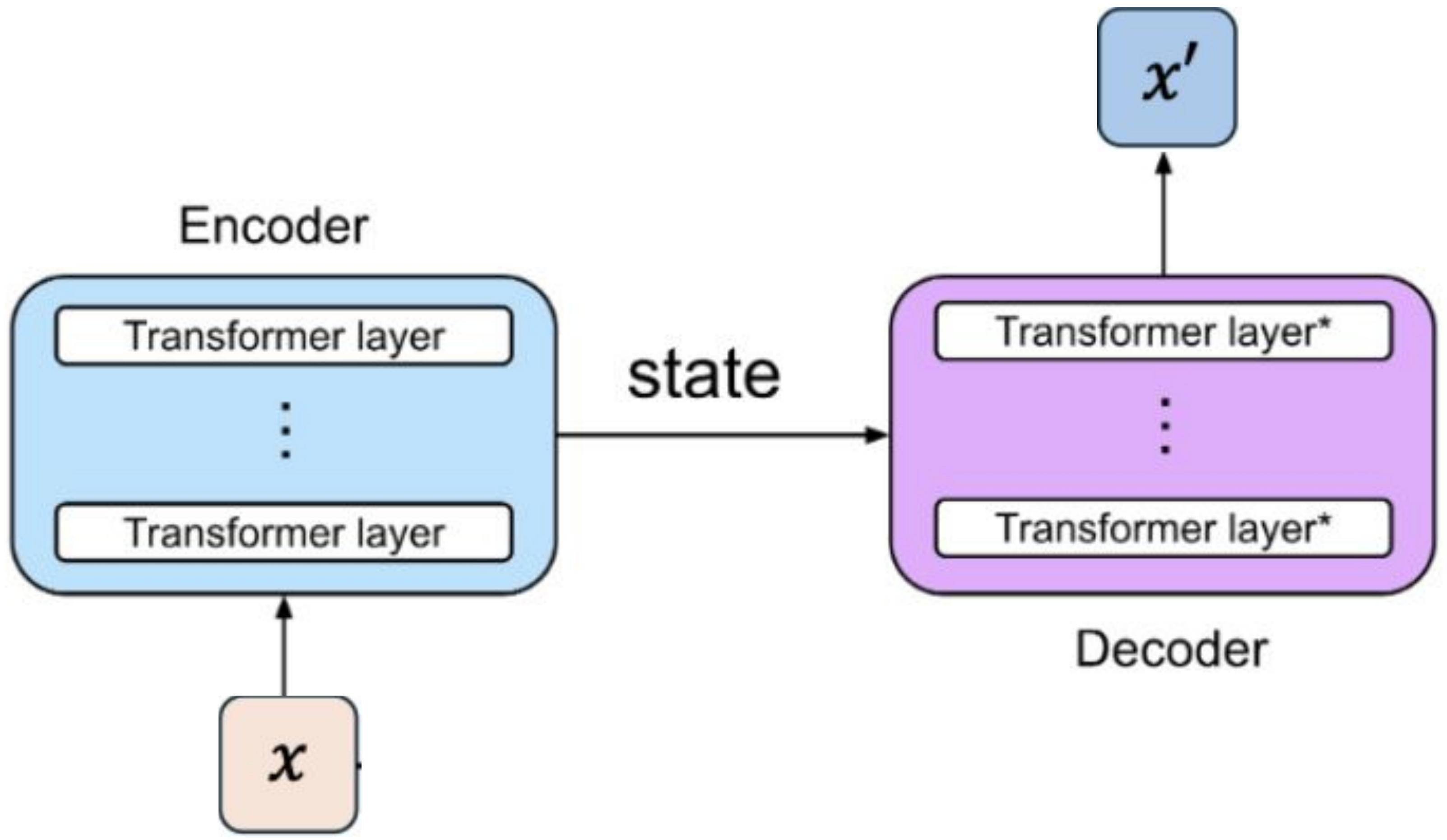
- **Deep Learning-Based Generation**



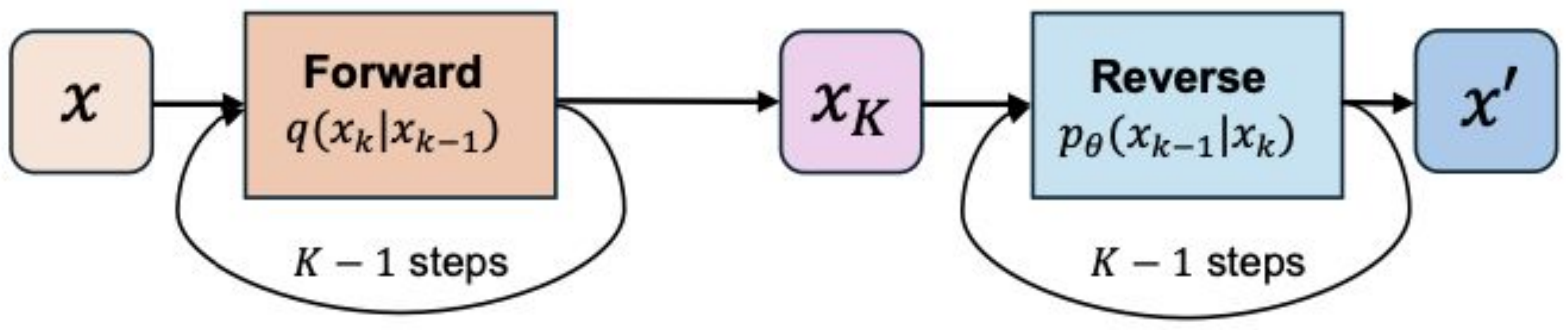
VAEs (Ex. TVAE, GOGGLE)



GANs (Ex. CTGAN)



Transformers (Ex.GReaT)



Diffusion Models

Source: https://github.com/SanDiegoMachineLearning/talks

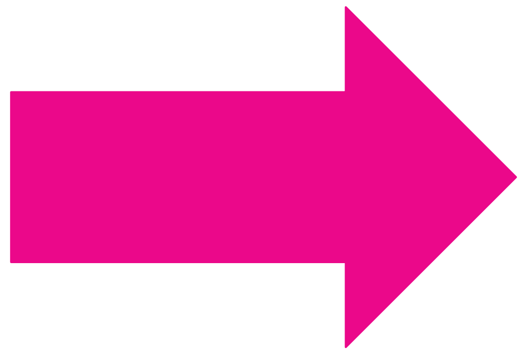
Diffusion Models for Tabular Data Synthesis

VECTOR INSTITUTE

- **Diffusion Models** originally designed for 1) pure continuous pixels of image data 2) with only local correlation.

- **Challenges:**

1) Tabular data contain mixed type of data → Hard to learn **discrete categorical feature.**
2) Tabular data have complex and varied distribution → Hard to learn **joint probabilities across columns.**



	age (n)	job (c)	marital (c)	education (c)	balance (n)	housing (c)
0	30	unemployed	married	primary	1787	no
1	33	services	married	secondary	4789	yes
2	35	management	single	tertiary	1350	yes
3	30	management	married	tertiary	1476	yes
4	59	blue-collar	married	secondary	0	yes
5	35	management	single	tertiary	747	no

Source: https://github.com/SanDiegoMachineLearning/talks

Diffusion Models for Tabular Data Synthesis

VECTOR INSTITUTE

- **Diffusion Models** originally designed for 1) pure continuous pixels of image data 2) with only local correlation.

- **Challenges:**

1) **Tabular data contain mixed type of data → Hard to learn categorical feature.**

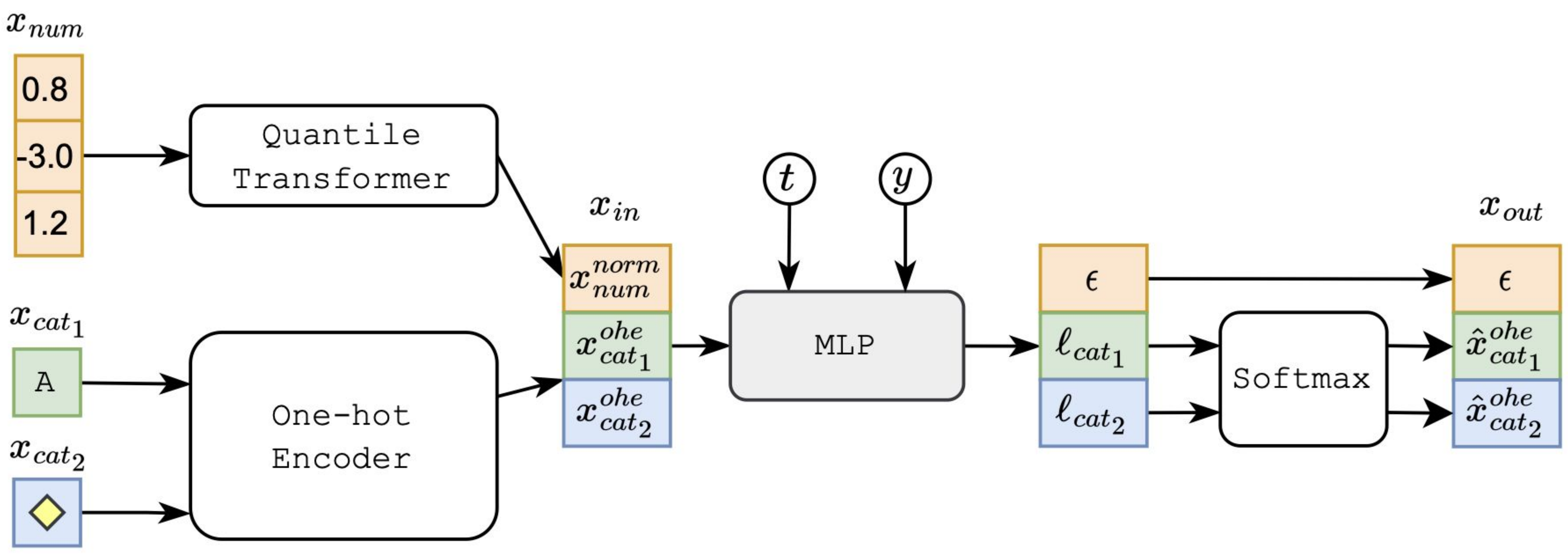
   - Transform categorical features to numerical one
     - One-hot Encoding (Ex. StaSy)
     - Analog Bit Encoding (Ex. TabCSDI)

   - Use diffusion model tailored for discrete categorical (ex. **TabDDPM**)

Overview of TabDDPM

VECTOR INSTITUTE

- Utilize two different diffusion process:
  - **Gaussian diffusion models** for numerical features
  - **Multinomial diffusion models** for categorical variables



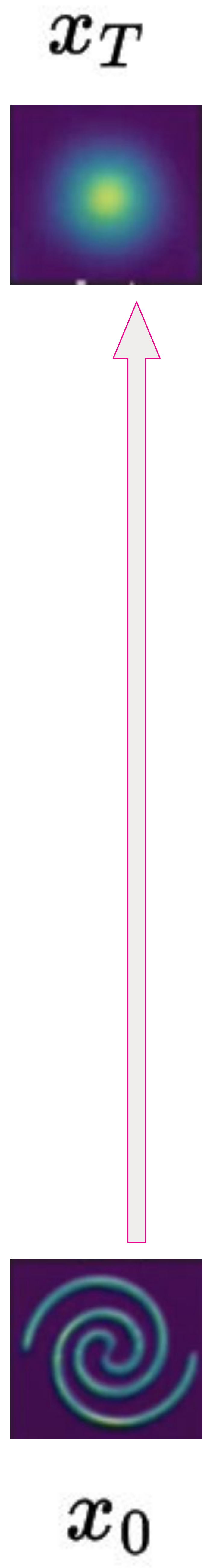
Source: [TABDDPM: MODELLING TABULAR DATA WITH DIFFUSION MODELS]

Gaussian Diffusion Model (Recap)

x_0

Source: Argmax Flows and Multinomial Diffusion

Gaussian Diffusion Model (Recap)

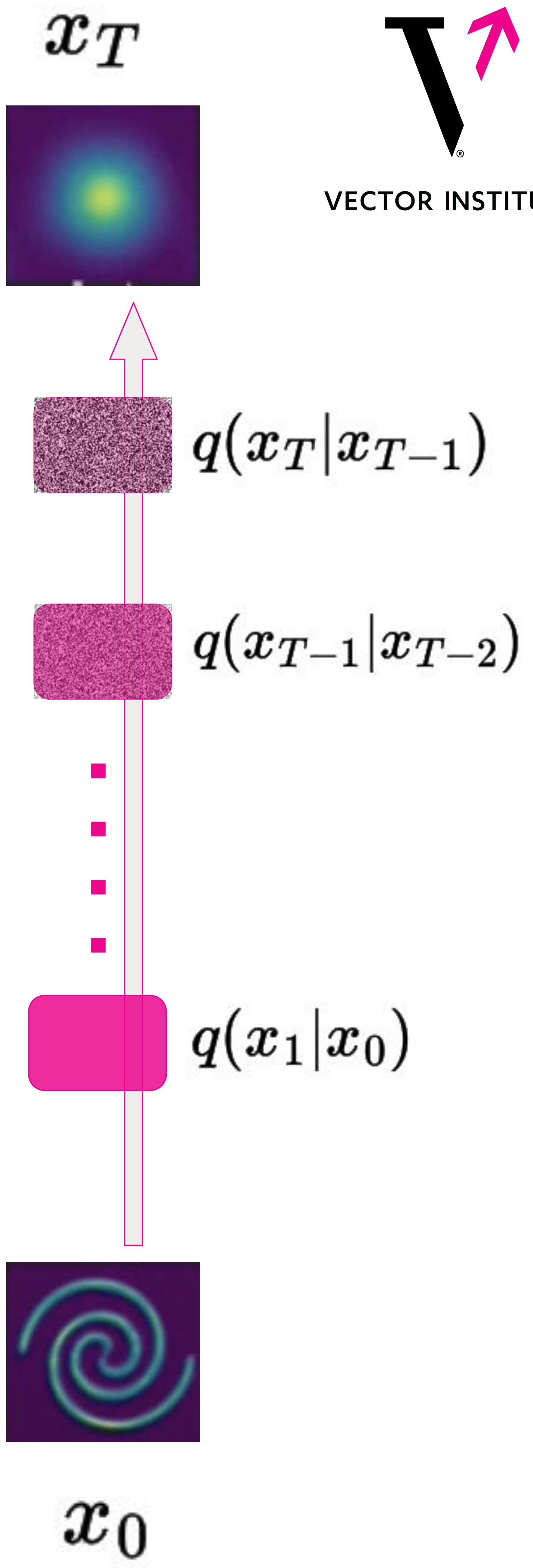


Source: Argmax Flows and Multinomial Diffusion

Gaussian Diffusion Model (Recap)

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-1}) = N(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$



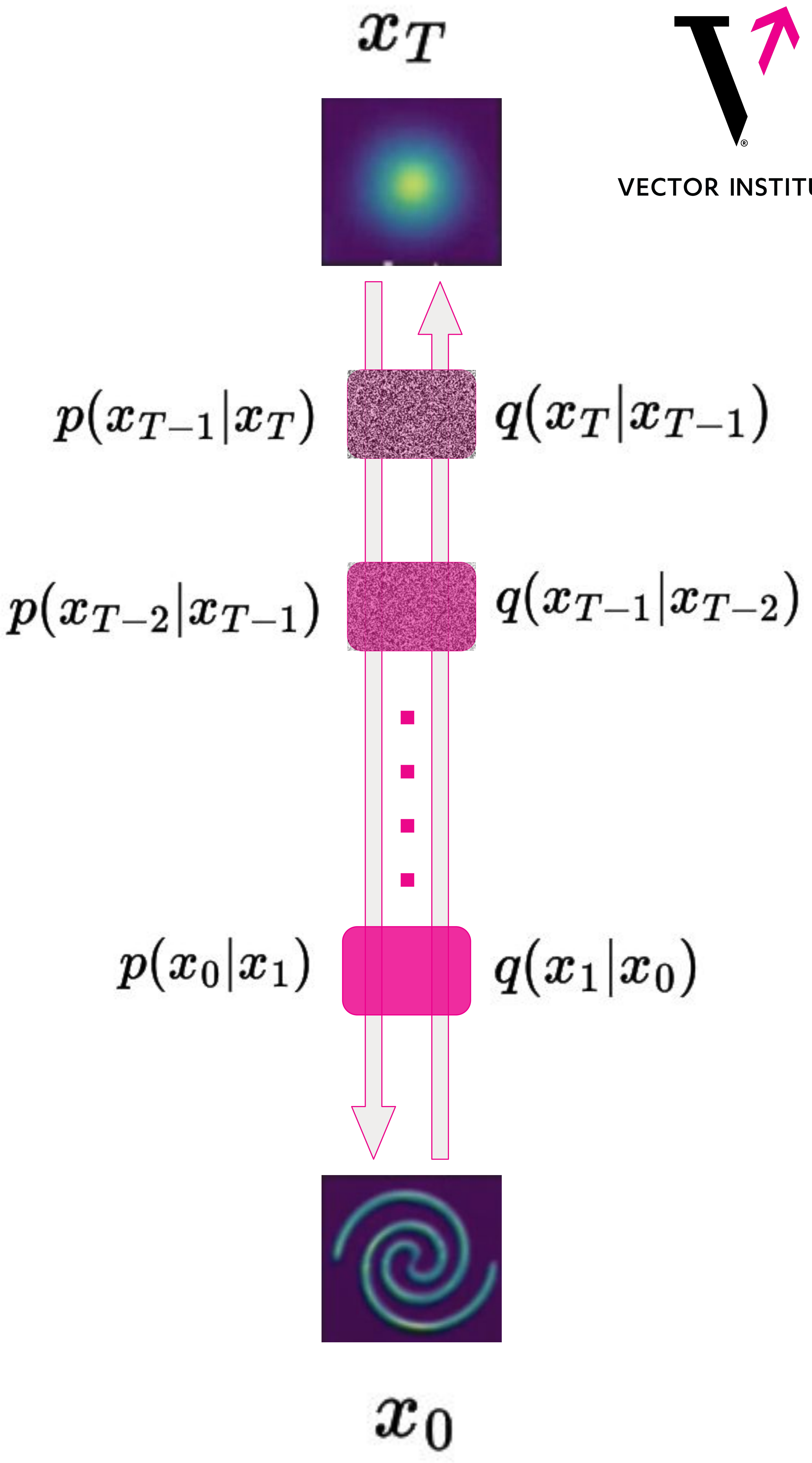
VECTOR INSTITUTE

Source: Argmax Flows and Multinomial Diffusion

Gaussian Diffusion Model (Recap)

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-1}) = N(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$



VECTOR INSTITUTE

Source: Argmax Flows and Multinomial Diffusion

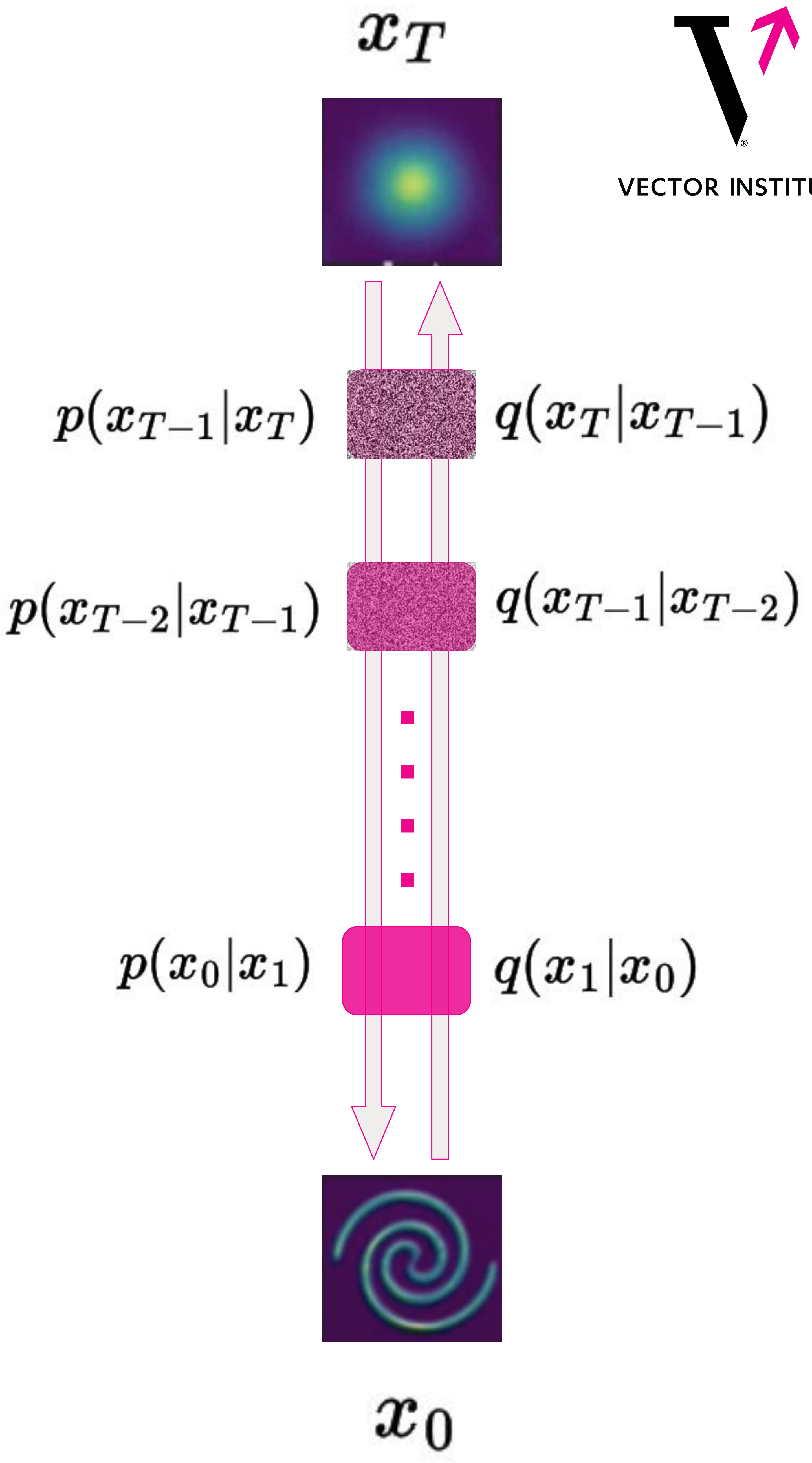
Gaussian Diffusion Model (Recap)

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-1}) = N(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

- Learnable denoising process:

$$p(x_{t-1}|x_t) = N(x_{t-1}|\mu_\theta(x_t), \Sigma_\theta(x_t))$$



Source: Argmax Flows and Multinomial Diffusion

Gaussian Diffusion Model (Recap)

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-1}) = N(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

- Learnable denoising process:

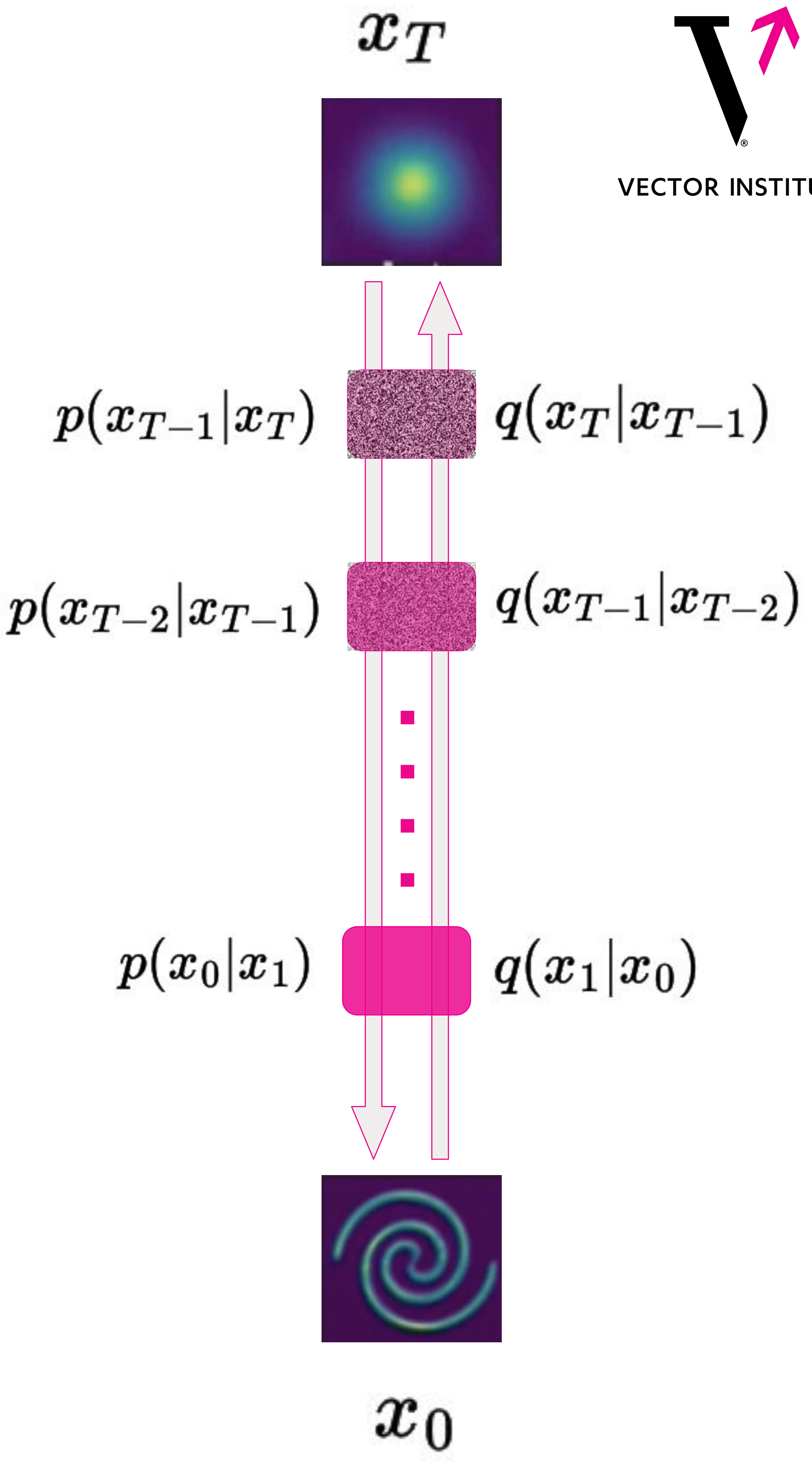
$$p(x_{t-1}|x_t) = N(x_{t-1}|\mu_\theta(x_t), \Sigma_\theta(x_t))$$

- We can decompose ELBO: $\mathcal{L} = \Sigma_{t=0}^{T} L_t$

$$L_0 = \mathbb{E}_{q(x_1|x_0)}[\log p(x_0|x_1)]$$

$$L_t = \mathbb{E}_{q(x_t|x_0)}[\mathbb{D}_{KL}[q(x_{t-1}|x_t, x_0)||p(x_{t-1}|x_t)]]$$

$$L_T = \mathbb{D}_{KL}[q(x_T|x_0)||p(x_t)]$$



VECTOR INSTITUTE

Source: Argmax Flows and Multinomial Diffusion

Gaussian Diffusion Model (Recap)

VECTOR INSTITUTE

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-1}) = N(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

- Learnable denoising process:

$$p(x_{t-1}|x_t) = N(x_{t-1}|\mu_\theta(x_t), \Sigma_\theta(x_t))$$

- We can decompose ELBO: $\mathcal{L} = \Sigma_{t=0}^{T} L_t$

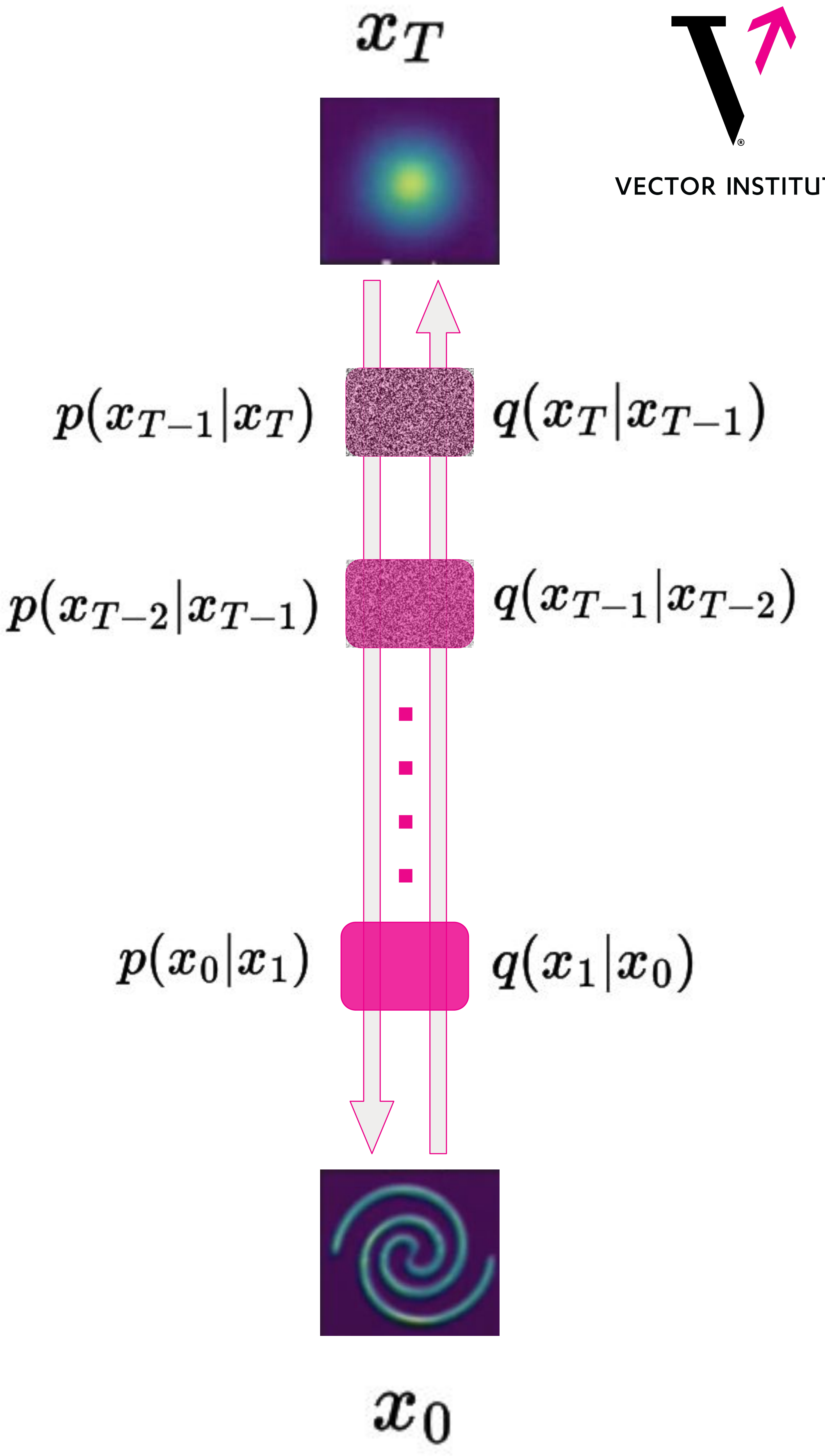
$$L_0 = \mathbb{E}_{q(x_1|x_0)}[\log p(x_0|x_1)]$$

$$L_t = \mathbb{E}_{q(x_t|x_0)}[\mathbb{D}_{KL}[q(x_{t-1}|x_t, x_0)||p(x_{t-1}|x_t)]]$$

$$L_T = \mathbb{D}_{KL}[q(x_T|x_0)||p(x_t)]$$

- Which allows efficient training by sampling, using that:

$$\left.\begin{array}{r} q(x_t|x_0) \\ q(x_{t-1}|x_t, x_0) \end{array}\right\} \text{Closed-form Gaussians}$$



Source: [Argmax Flows and Multinomial Diffusion]

Gaussian Diffusion Model (Recap)

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-1}) = \boxed{N(x_t|\sqrt{1-\beta_t}x_{t-1}, \beta_t I)}$$

- Learnable denoising process:

$$p(x_{t-1}|x_t) = \boxed{N(x_{t-1}|\mu_\theta(x_t), \Sigma_\theta(x_t))}$$

- We can decompose ELBO: $\mathcal{L} = \Sigma_{t=0}^{T} L_t$

$$L_0 = \mathbb{E}_{q(x_1|x_0)}[\log p(x_0|x_1)]$$

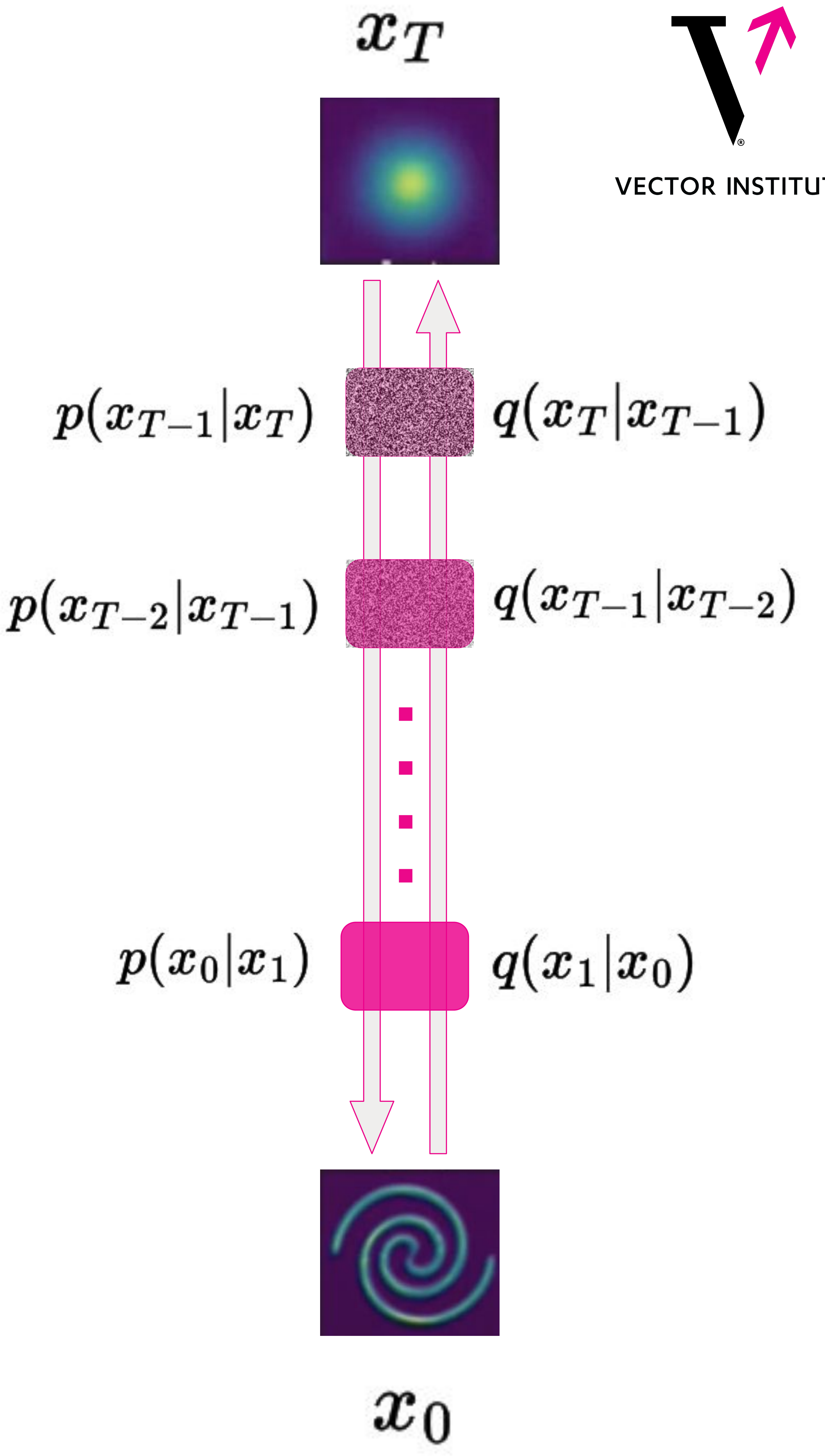
$$L_t = \mathbb{E}_{q(x_t|x_0)}[\mathbb{D}_{KL}[q(x_{t-1}|x_t, x_0)||p(x_{t-1}|x_t)]]$$

$$L_T = \mathbb{D}_{KL}[q(x_T|x_0)||p(x_t)]$$

- Which allows efficient training by sampling, using that:

$$\left.\begin{array}{r} q(x_t|x_0) \\ q(x_{t-1}|x_t, x_0) \end{array}\right\} \boxed{\text{Closed-form Gaussians}}$$

Source: [Argmax Flows and Multinomial Diffusion]



VECTOR INSTITUTE

Multinomial Diffusion Model

- Fixed noising process (no learnable parameters):

$$q(x_t|x_{t-q}) = \boxed{Cat(x_t|(1-\beta_t)x_{t-1} + \beta_t \frac{1}{K})}$$

- Learnable denoising process:

$$p(x_{t-1}|x_t) = \boxed{Cat(x_{t-1}|\pi_\theta(x_t))}$$

- We can decompose ELBO: $\mathcal{L} = \Sigma_{t=0}^{T} L_t$

$$L_0 = \mathbb{E}_{q(x_1|x_0)}[\log p(x_0|x_1)]$$

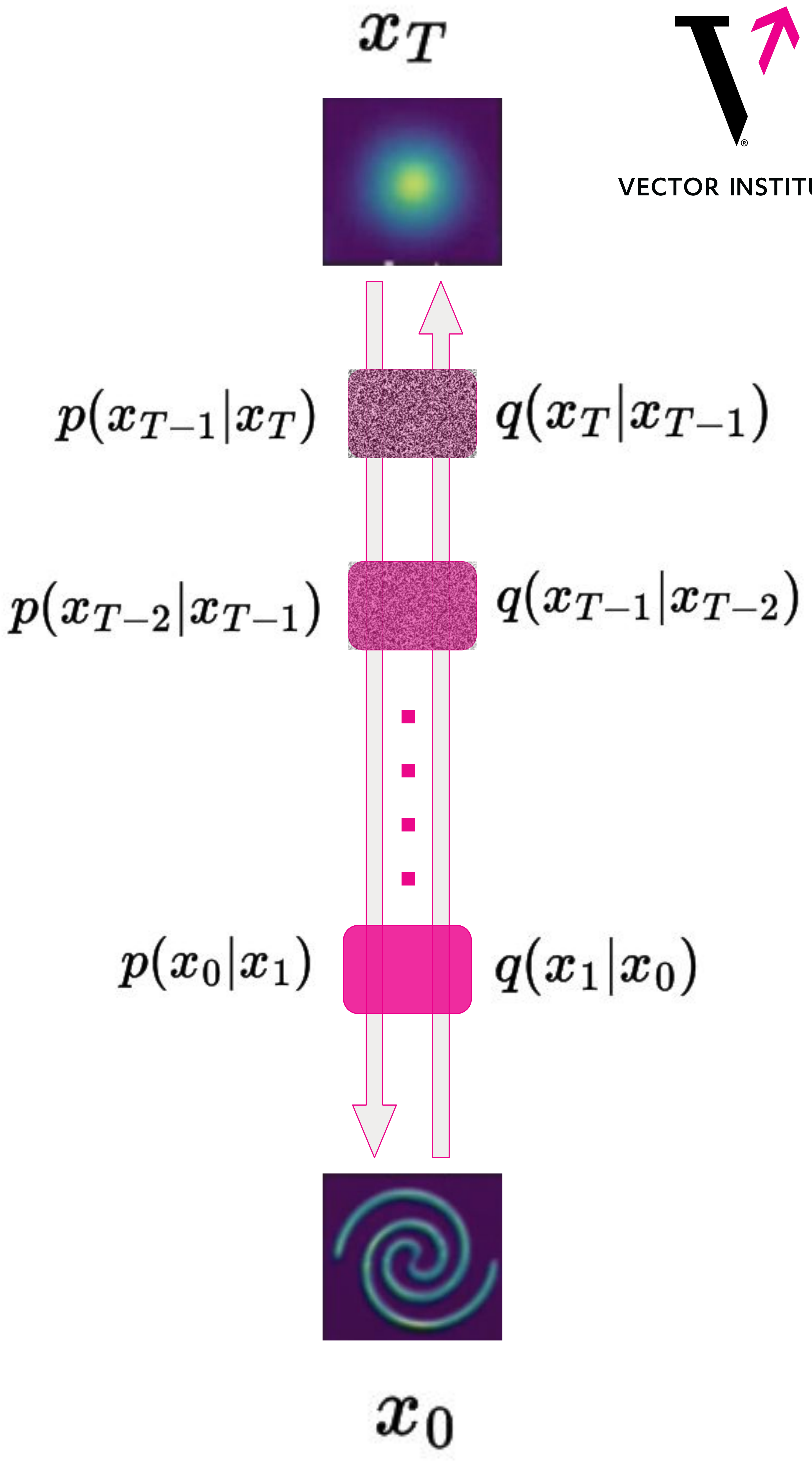
$$L_t = \mathbb{E}_{q(x_t|x_0)}[\mathbb{D}_{KL}[q(x_{t-1}|x_t, x_0)||p(x_{t-1}|x_t)]]$$

$$L_T = \mathbb{D}_{KL}[q(x_T|x_0)||p(x_t)]$$

- Which allows efficient training by sampling, using that:

$$\left.\begin{array}{r} q(x_t|x_0) \\ q(x_{t-1}|x_t, x_0) \end{array}\right\} \boxed{\text{Closed-form Categoricals}}$$

Source: Argmax Flows and Multinomial Diffusion



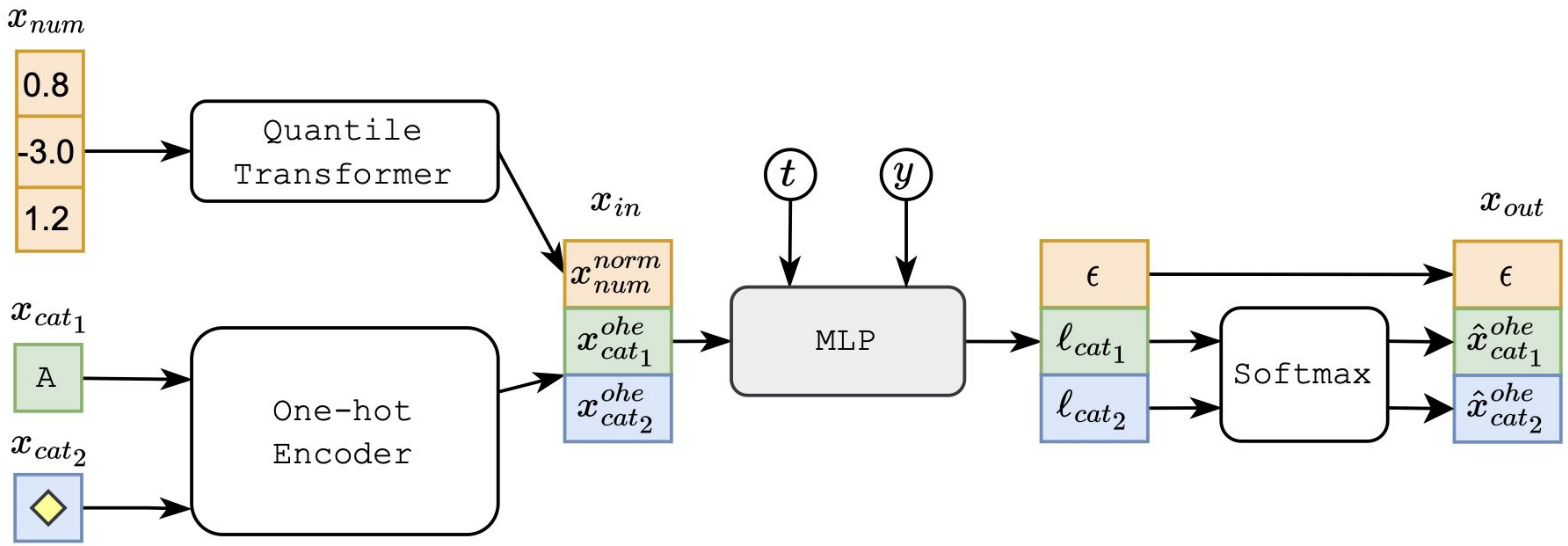
VECTOR INSTITUTE

Train Diffusion in Data Space

- We apply proper transformations on each type of data and apply forward process.
- Define loss for **Gaussian & Multinomial diffusion models** as:

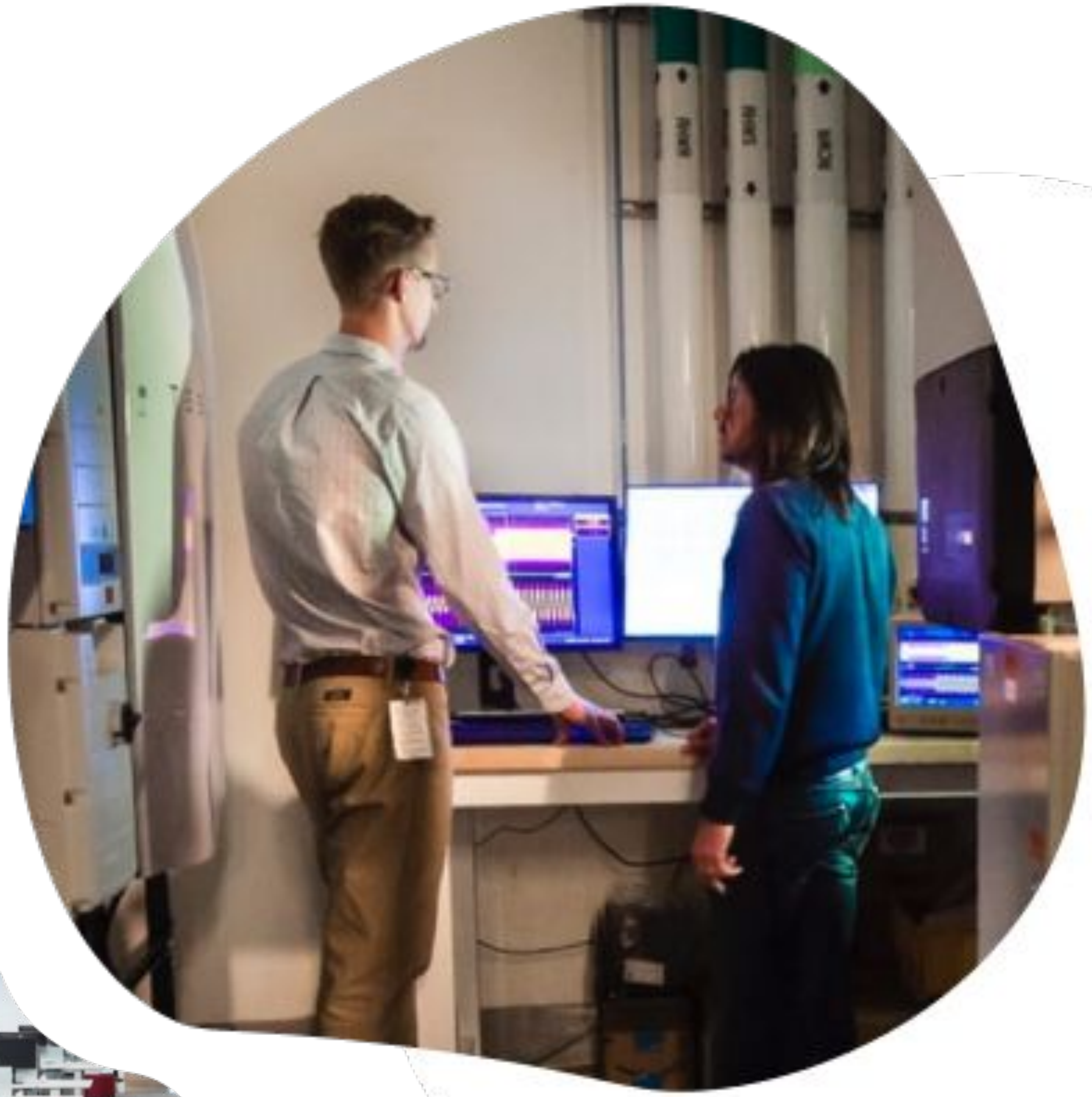
$$L_t^{TabDDPM} = L_t^{simple} + \frac{\sum_{i \le C} L_t^i}{C}$$

- Train Multilayer Perceptron (MLP) model **conditional on label**:
  - For classification tasks → Use class-conditioned model
  - For regression tasks → Add target value as numerical feature



Source: [TABDDPM: MODELLING TABULAR DATA WITH DIFFUSION MODELS]

Next Steps and Q&A



VECTOR INSTITUTE

Diffusion Models Bootcamp

For Single-Table Tabular Datasets

(Part 2)

Applied AI Projects

August 7, 2024

Diffusion Models for Tabular Data Synthesis

VECTOR INSTITUTE

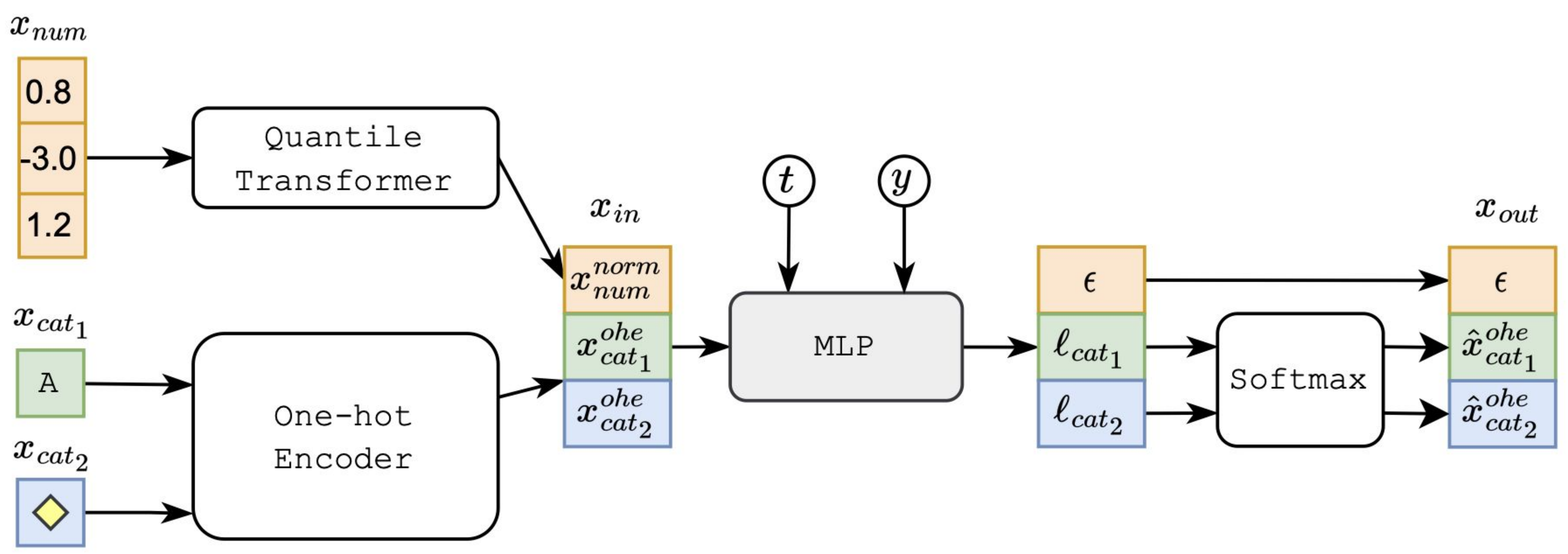
- **Diffusion Models** originally designed for 1) pure continuous pixels of image data 2) with only local correlation.

- **Challenges:**

1) **Tabular data contain mixed type of data → Hard to learn categorical feature.**

   - Transform categorical features to numerical one
     - One-hot Encoding (Ex. StaSy)
     - Analog Bit Encoding (Ex. TabCSDI)

   - Train two different diffusion process for numerical and categorical data (Ex. **TabDDPM**)



Diffusion Models for Tabular Data Synthesis

VECTOR INSTITUTE

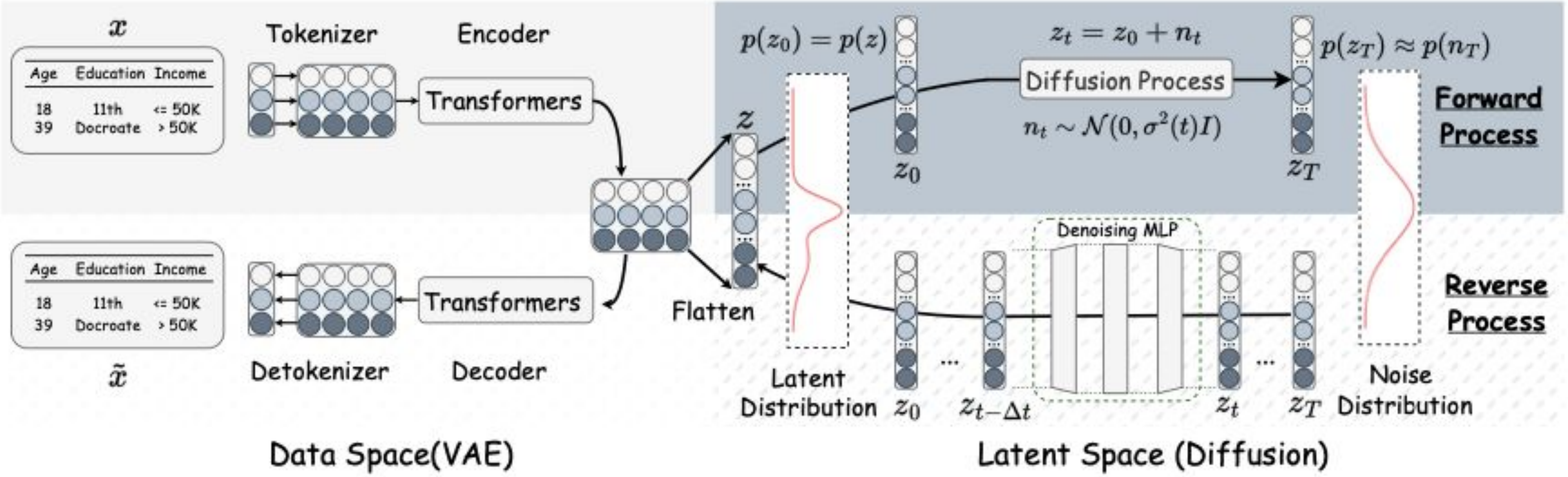
- **Diffusion Models** originally designed for 1) pure continuous pixels of image data 2) with only local correlation.
- **Challenges:**

1) **Tabular data contain mixed type of data → Hard to learn categorical feature.**
   - Transform categorical features to numerical one
     - One-hot Encoding (Ex. StaSy)
     - Analog Bit Encoding (Ex. TabCSDI)
   - Train two different diffusion process for numerical and categorical data (Ex. **TabDDPM**)
1) **Tabular data have complex and varied distribution → Hard to learn joint probabilities across columns.**
   - Develop diffusion model in latent space for both numerical and categorical data (Ex. **TabSyn**)

TabSyn aims to improve TabDDPM in generality, quality and speed !!!

Overview of TabSyn

- Utilize two step process:
  - train a **VAE model** to map the mixed data space to continuous latent space.
  - train a **latent diffusion model** over latent space.



Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Train VAE in Data Space

VECTOR INSTITUTE

1. Feature Tokenizer

Learnable

$$e_i^{\mathrm{num}} = x_i^{\mathrm{num}} \cdot \boxed{\boldsymbol{w}_i^{\mathrm{num}}} + \boxed{\boldsymbol{b}_i^{\mathrm{num}}}, \quad \boldsymbol{e}_i^{\mathrm{cat}} = \boldsymbol{x}_i^{\mathrm{oh}} \cdot \boxed{\boldsymbol{W}_i^{\mathrm{cat}}} + \boxed{\boldsymbol{b}_i^{\mathrm{cat}}}$$

$$\boldsymbol{E} = [\boldsymbol{e}_1^{\mathrm{num}}, \cdots, \boldsymbol{e}_{M_{\mathrm{num}}}^{\mathrm{num}}, \boldsymbol{e}_1^{\mathrm{cat}}, \cdots, \boldsymbol{e}_{M_{\mathrm{cat}}}^{\mathrm{cat}}] \in \mathbb{R}^{M \times d}$$

Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Train VAE in Data Space

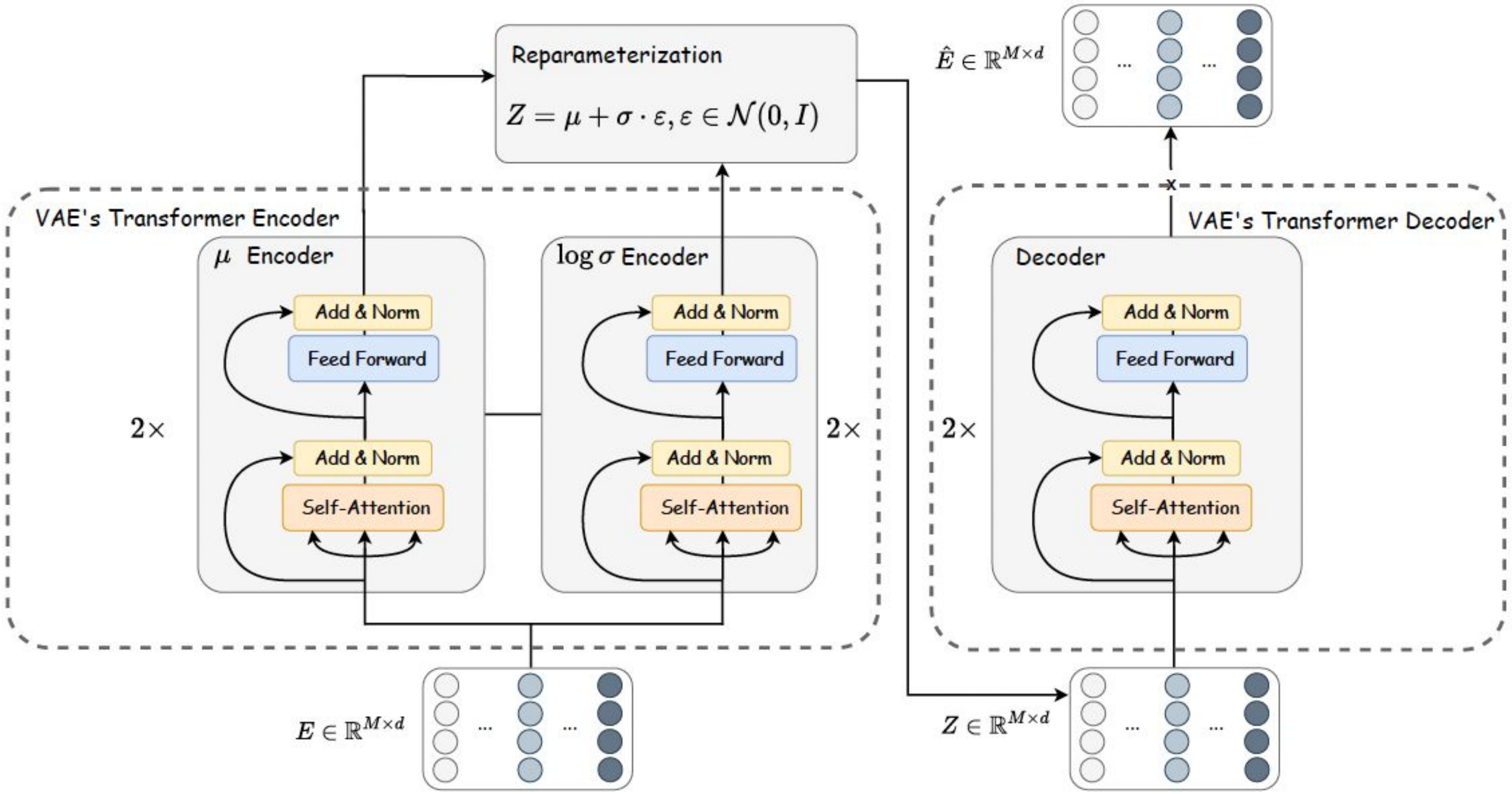
1. Feature Tokenizer

Learnable

$$e_i^{\text{num}} = x_i^{\text{num}} \cdot \boxed{w_i^{\text{num}}} + \boxed{b_i^{\text{num}}}, \quad e_i^{\text{cat}} = x_i^{\text{oh}} \cdot \boxed{W_i^{\text{cat}}} + \boxed{b_i^{\text{cat}}}$$

$$E = [e_1^{\text{num}}, \cdots, e_{M_{\text{num}}}^{\text{num}}, e_1^{\text{cat}}, \cdots, e_{M_{\text{cat}}}^{\text{cat}}] \in \mathbb{R}^{M \times d}$$

1. Transformer Encoding and Decoding



Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Train VAE in Data Space

Learnable

$$e_i^{\mathrm{num}} = x_i^{\mathrm{num}} \cdot \boxed{\boldsymbol{w}_i^{\mathrm{num}}} + \boxed{\boldsymbol{b}_i^{\mathrm{num}}}, \quad \boldsymbol{e}_i^{\mathrm{cat}} = \boldsymbol{x}_i^{\mathrm{oh}} \cdot \boxed{\boldsymbol{W}_i^{\mathrm{cat}}} + \boxed{\boldsymbol{b}_i^{\mathrm{cat}}}$$

$$\boldsymbol{E} = [\boldsymbol{e}_1^{\mathrm{num}}, \cdots, \boldsymbol{e}_{M_{\mathrm{num}}}^{\mathrm{num}}, \boldsymbol{e}_1^{\mathrm{cat}}, \cdots, \boldsymbol{e}_{M_{\mathrm{cat}}}^{\mathrm{cat}}] \in \mathbb{R}^{M \times d}$$

1. Feature Tokenizer

1. Transformer Encoding and Decoding

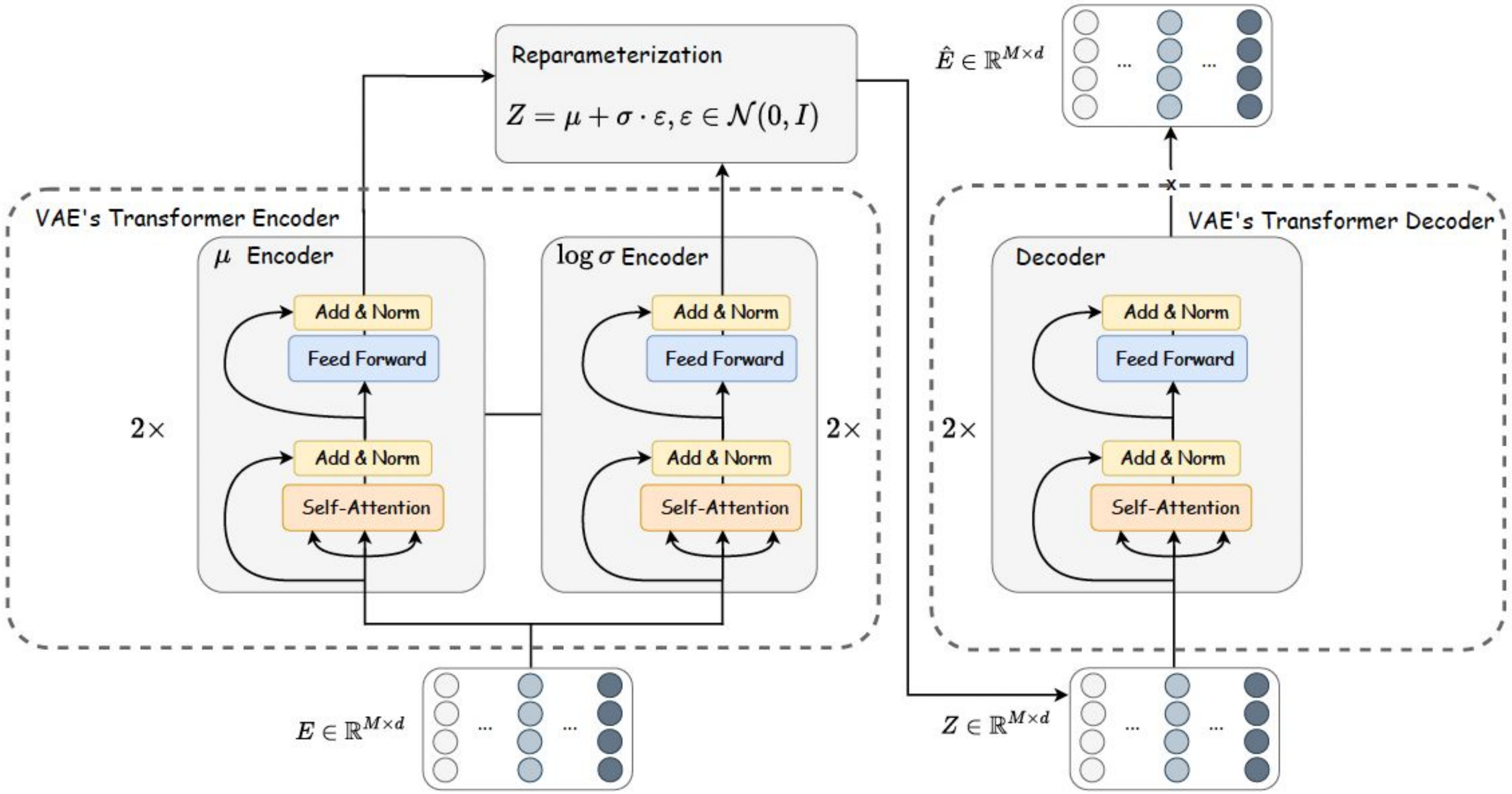
1. Feature Detokenizer

Learnable

$$\hat{x}_i^{\mathrm{num}} = \hat{\boldsymbol{e}}_i^{\mathrm{num}} \cdot \boxed{\hat{\boldsymbol{w}}_i^{\mathrm{num}}} + \boxed{\hat{b}_i^{\mathrm{num}}}$$

$$\hat{\boldsymbol{x}}_i^{\mathrm{oh}} = \mathrm{Softmax}(\hat{\boldsymbol{e}}_i^{\mathrm{cat}} \cdot \boxed{\hat{\boldsymbol{W}}_i^{\mathrm{cat}}} + \boxed{\hat{\boldsymbol{b}}_i^{\mathrm{cat}}}),$$

$$\hat{\boldsymbol{x}} = [\hat{x}_1^{\mathrm{num}}, \cdots, \hat{x}_{M_{\mathrm{num}}}^{\mathrm{num}}, \hat{\boldsymbol{x}}_1^{\mathrm{oh}}, \cdots, \hat{\boldsymbol{x}}_{M_{\mathrm{cat}}}^{\mathrm{oh}}]$$



Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Train VAE in Data Space

Learnable

$$e_i^{\text{num}} = x_i^{\text{num}} \cdot \boxed{w_i^{\text{num}}} + \boxed{b_i^{\text{num}}}, \quad e_i^{\text{cat}} = x_i^{\text{oh}} \cdot \boxed{W_i^{\text{cat}}} + \boxed{b_i^{\text{cat}}}$$

$$E = [e_1^{\text{num}}, \cdots, e_{M_{\text{num}}}^{\text{num}}, e_1^{\text{cat}}, \cdots, e_{M_{\text{cat}}}^{\text{cat}}] \in \mathbb{R}^{M \times d}$$

1. Feature Tokenizer

1. Transformer Encoding and Decoding

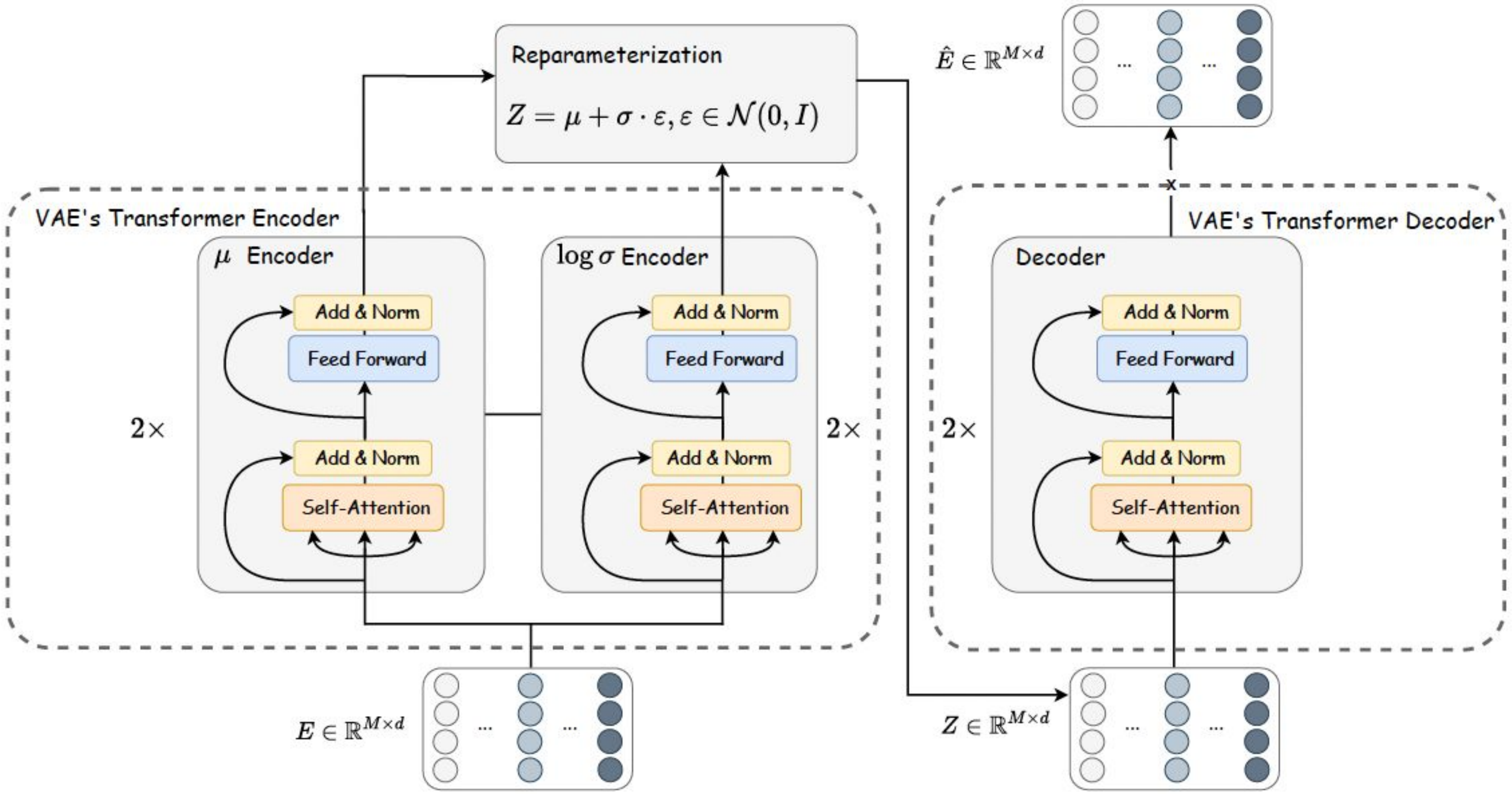
1. Feature Detokenizer

Learnable

$$\hat{x}_i^{\text{num}} = \hat{e}_i^{\text{num}} \cdot \boxed{\hat{w}_i^{\text{num}}} + \boxed{\hat{b}_i^{\text{num}}}$$

$$\hat{x}_i^{\text{oh}} = \text{Softmax}(\hat{e}_i^{\text{cat}} \cdot \boxed{\hat{W}_i^{\text{cat}}} + \boxed{\hat{b}_i^{\text{cat}}}),$$

$$\hat{x} = [\hat{x}_1^{\text{num}}, \cdots, \hat{x}_{M_{\text{num}}}^{\text{num}}, \hat{x}_1^{\text{oh}}, \cdots, \hat{x}_{M_{\text{cat}}}^{\text{oh}}]$$

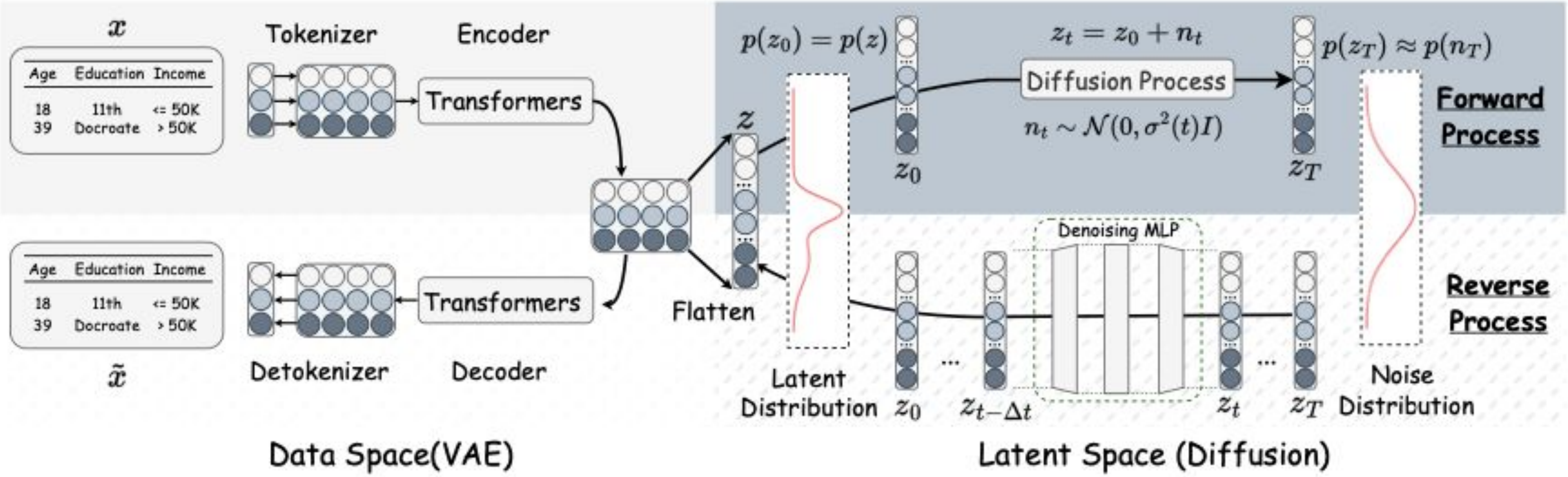


Loss function: $\mathcal{L} = \ell_{\text{recon}}(x, \hat{x}) + \beta \ell_{\text{kl}}$

Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Train Diffusion in Latent Space

- As all of data modality maps to one latent space in has more **generality** in handling broader spectrum of data.
- The generative model in latent space improves robustness and flexibility in controlling generated styles, resulting in higher **quality** output.
- Noise is added through linear scheduler helps to skip steps inference leading to increase **speed**.
- As they use unconditional diffusion models they can easily use the same trained model for both **imputation and synthesis**.



Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

VECTOR INSTITUTE

Inference Data Synthesis

- Initialize latent space with random gaussian noise

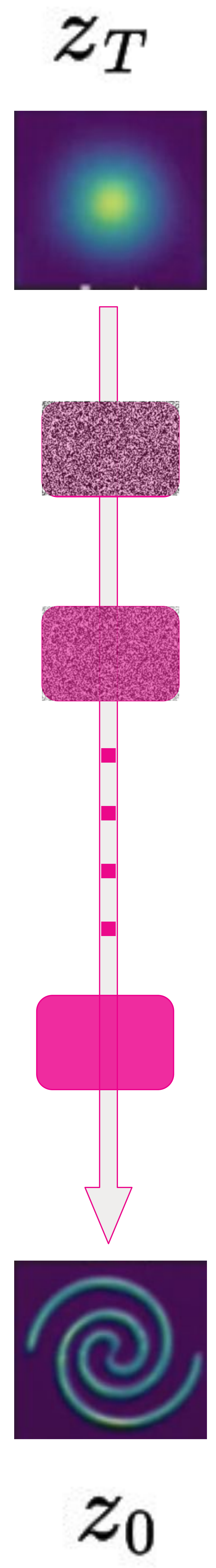
$$z \in R^{Md}$$

- Apply the denoise step as backward diffusion process

$$z_{t-1} \sim N(\mu_\theta(z_t), \sigma_\theta(z_t))$$

- Feed the output into VAE decoder

$$\hat{z} \in R^{Md} \rightarrow \hat{x} \in R^{M}$$



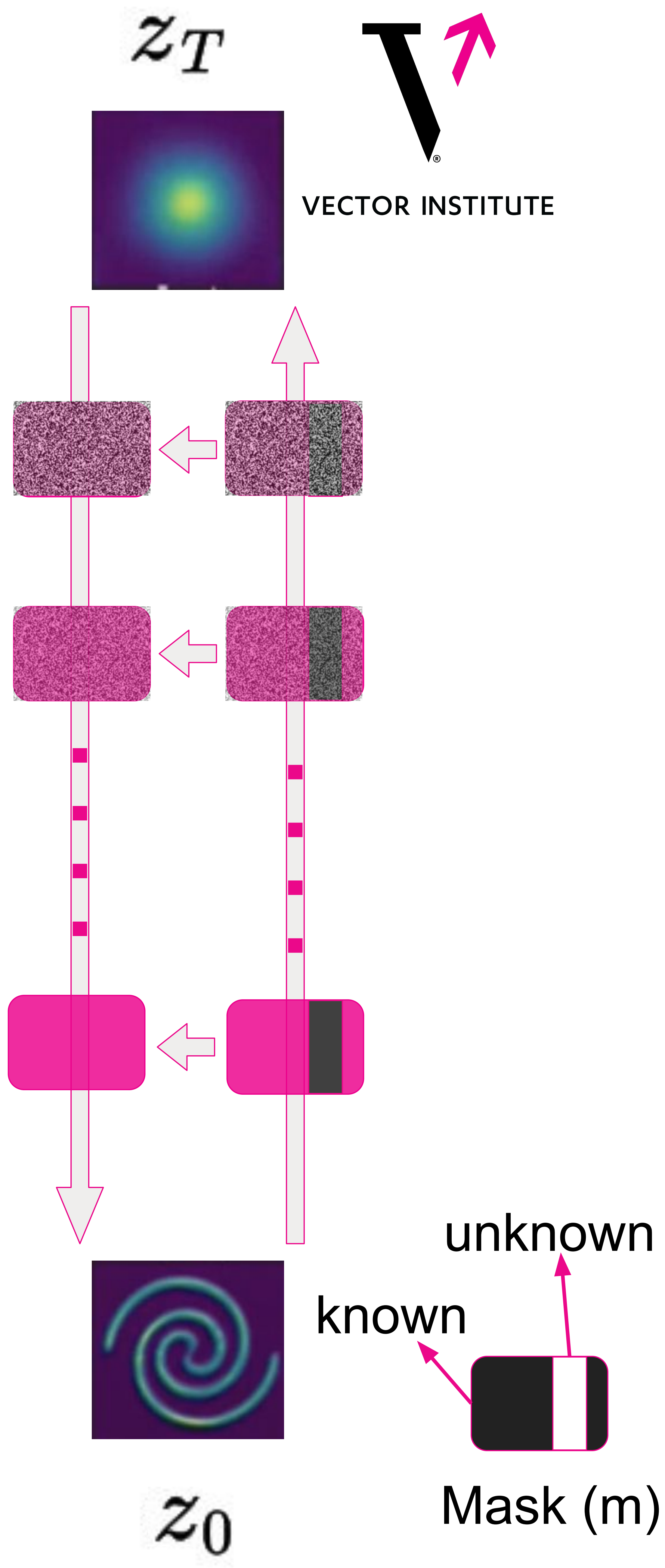
Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Inference Missing Value Imputation

- Preprocess missing column by
  - Numerical column: $x_{i,j}^{\text{num}} \Leftarrow \text{mean}(x_{:,j}^{\text{train}})$
  - Categorical column: $x_{i,j}^{\text{oh}} \Leftarrow [\frac{1}{C_j}, \cdots, \frac{1}{C_j}, \cdots \frac{1}{C_j}] \in \mathbb{R}^{1 \times C_j}$
- Feed the masked data into VAE encoder $x \in R^M \longrightarrow z \in R^{Md}$
- Obtain masking vector on latent space
  - As it is a deterministic mapping, we can create a masking vector

$$z_{t-1} = m \odot z_{t-1}^{\text{known}} + (1-m) \odot z_{t-1}^{\text{unknown}}$$

- Apply the denoise step as the mixture of known parts forwards process and unknown parts backward process $z_{t-1}^{known} \sim N(\sqrt{1-\beta_t} z_0, \beta_t I)$ $\quad z_{t-1}^{unknown} \sim N(\mu_\theta(z_t), \Sigma_\theta(z_t))$
- Feed the output into VAE decoder $\hat{z} \in R^{Md} \rightarrow \hat{x} \in R^M$
- Since this process is stochastic, we run imputation multiple times and get average.

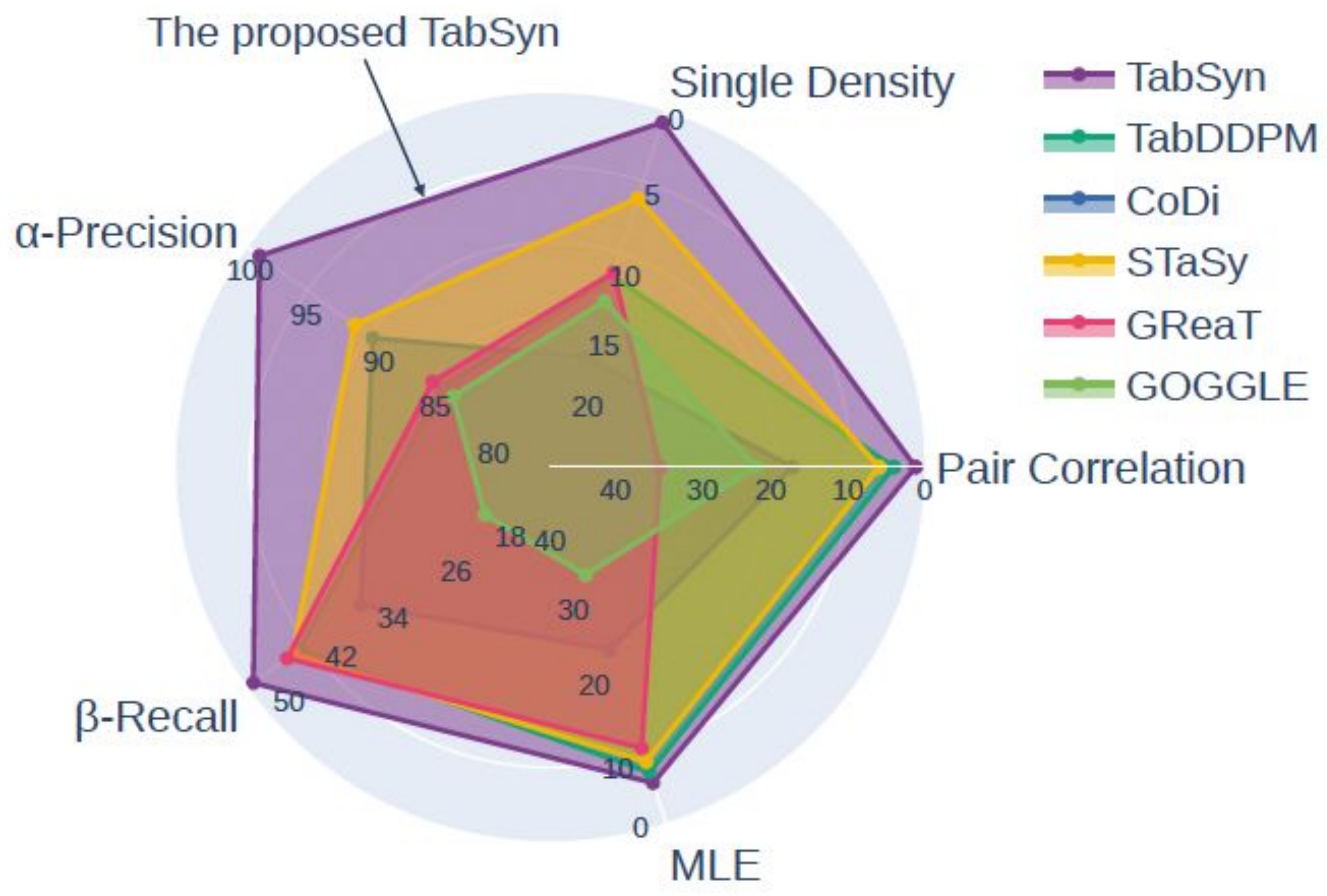


Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Comprehensive Evaluation

VECTOR INSTITUTE

- We compare synthesize data from following aspects:
  - Utility: application to downstream task
    - Machine learning efficiency
    - Missing value imputation
  - Fidelity: how realistic is synthetic data
    - Low-order statistics
    - High-order statistics
    - Real vs synthetic detection
  - Privacy protection
    - Distance to closest record



Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Single Table Dataset

VECTOR INSTITUTE

- They study 6 tabular dataset from UCI Machine Learning repository

Dataset	# Rows	# Num	# Cat	# Train	# Validation	# Test	Task
Adult	48,842	6	9	28,943	3,618	16,281	Classification
Default	30,000	14	11	24,000	3,000	3,000	Classification
Shoppers	12,330	10	8	9,864	1,233	1,233	Classification
Magic	19,019	10	1	15,215	1,902	1,902	Classification
Beijing	43,824	7	5	35,058	4,383	4,383	Regression
News	39,644	46	2	31,714	3,965	3,965	Regression

Source: [MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE]

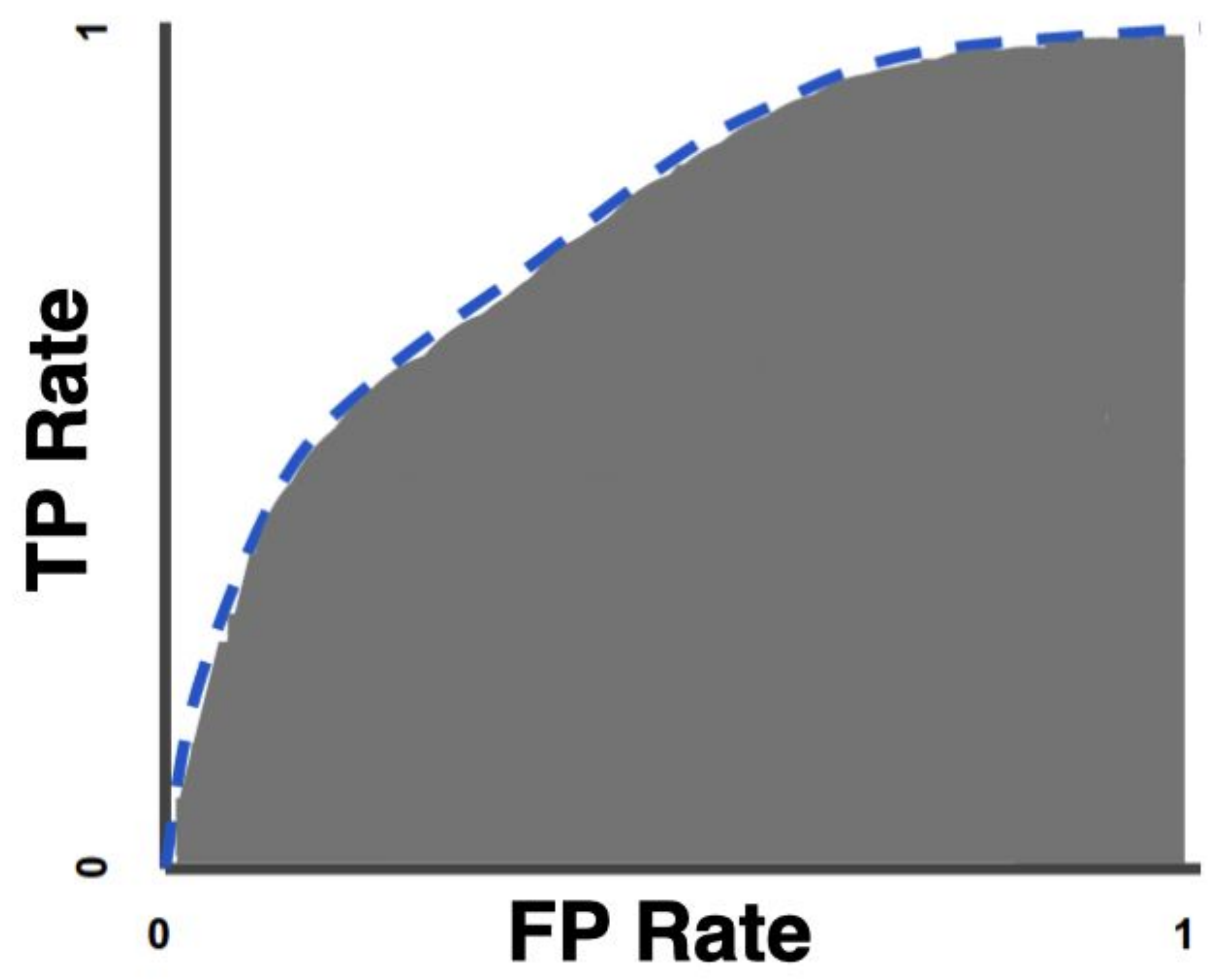
Single Table Dataset

VECTOR INSTITUTE

- They study 6 tabular dataset from UCI Machine Learning repository

Dataset	# Rows	# Num	# Cat	# Train	# Validation	# Test	Task
Adult	48, 842	6	9	28, 943	3, 618	16, 281	Classification
Default	30, 000	14	11	24, 000	3, 000	3, 000	Classification
Shoppers	12, 330	10	8	9, 864	1, 233	1, 233	Classification
Magic	19, 019	10	1	15, 215	1, 902	1, 902	Classification
Beijing	43, 824	7	5	35, 058	4, 383	4, 383	Regression
News	39, 644	46	2	31, 714	3, 965	3, 965	Regression

- Metrics used for tasks:
  - Classification → Area Under ROC Curve (ROC-AUC)



Source: [MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE]()

Single Table Dataset

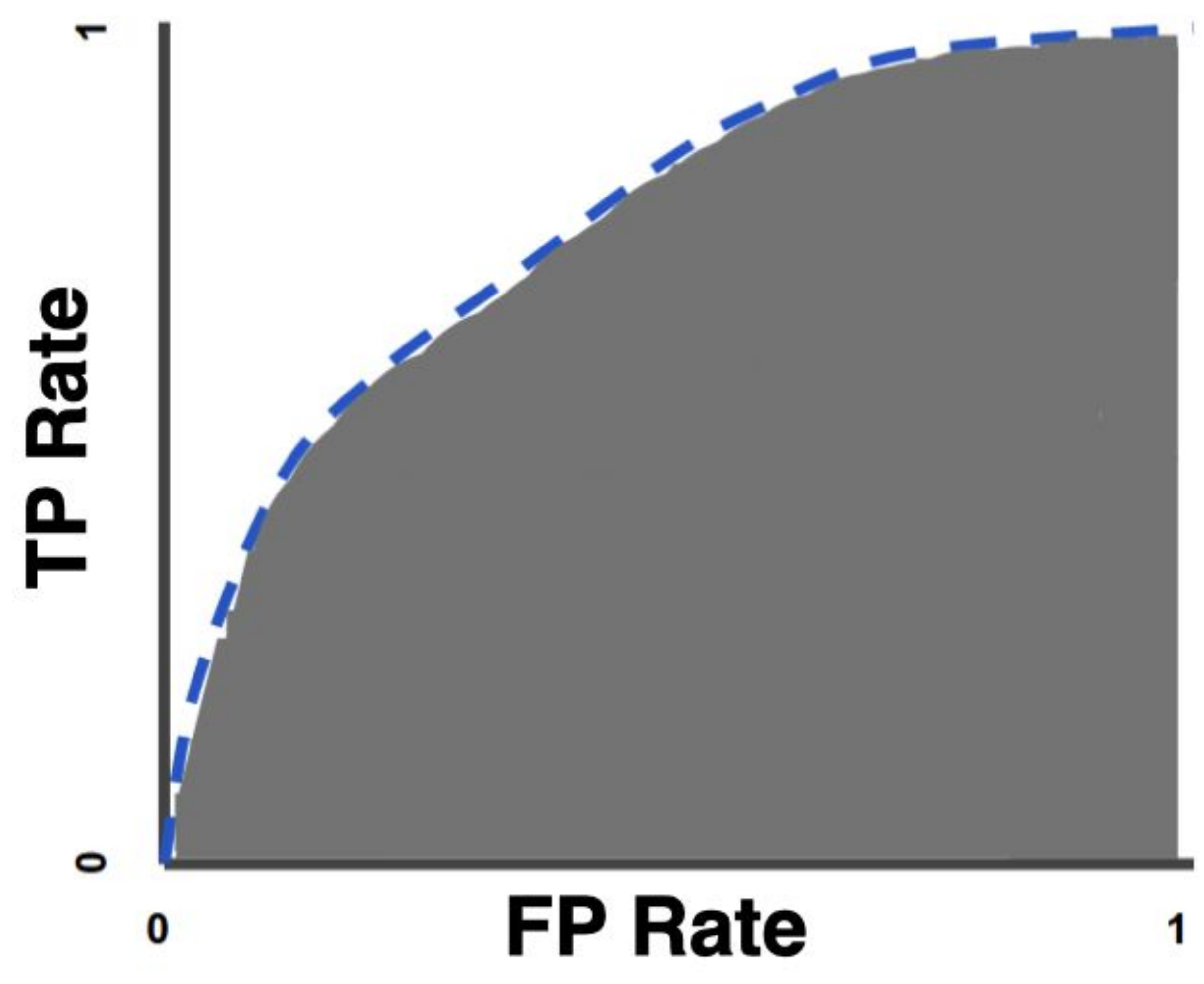
VECTOR INSTITUTE

- They study 6 tabular dataset from UCI Machine Learning repository

Dataset	# Rows	# Num	# Cat	# Train	# Validation	# Test	Task
Adult	48, 842	6	9	28, 943	3, 618	16, 281	Classification
Default	30, 000	14	11	24, 000	3, 000	3, 000	Classification
Shoppers	12, 330	10	8	9, 864	1, 233	1, 233	Classification
Magic	19, 019	10	1	15, 215	1, 902	1, 902	Classification
Beijing	43, 824	7	5	35, 058	4, 383	4, 383	Regression
News	39, 644	46	2	31, 714	3, 965	3, 965	Regression

- Metrics used for tasks:
  - Classification → Area Under ROC Curve (ROC-AUC)
  - Regression → Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \|y(i) - \hat{y}(i)\|^2}{N}},$$



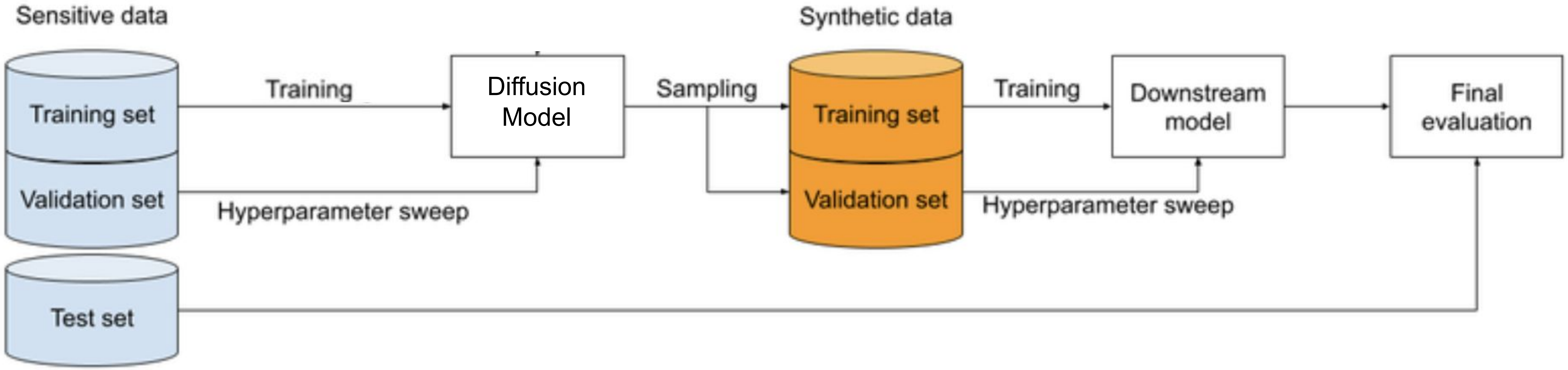
Source: MIXED-TYPE TABULAR DATA SYNTHESIS WITH SCORE-BASED DIFFUSION IN LATENT SPACE

Machine Learning Efficiency

VECTOR INSTITUTE

- Compare the performance of a model trained on synthetically generated data and evaluate on real test data for a downstream task
- Reported ROC-AUC for classification and RMSE for regression tasks

Methods	Adult	Default	Shoppers	Magic	Beijing	News[1]
	AUC ↑	AUC ↑	AUC ↑	AUC ↑	RMSE ↓	RMSE ↓
Real	$.927_{\pm .000}$	$.770_{\pm .005}$	$.926_{\pm .001}$	$.946_{\pm .001}$	$.423_{\pm .003}$	$.842_{\pm .002}$
SMOTE	$.899_{\pm .007}$	$.741_{\pm .009}$	$.911_{\pm .012}$	$.934_{\pm .008}$	$.593_{\pm .011}$	$.897_{\pm .036}$
CTGAN	$.886_{\pm .002}$	$.696_{\pm .005}$	$.875_{\pm .009}$	$.855_{\pm .006}$	$.902_{\pm .019}$	$.880_{\pm .016}$
TVAE	$.878_{\pm .004}$	$.724_{\pm .005}$	$.871_{\pm .006}$	$.887_{\pm .003}$	$.770_{\pm .011}$	$1.01_{\pm .016}$
GOGGLE	$.778_{\pm .012}$	$.584_{\pm .005}$	$.658_{\pm .052}$	$.654_{\pm .024}$	$1.09_{\pm .025}$	$.877_{\pm .002}$
GReaT	$.913_{\pm .003}$	$.755_{\pm .006}$	$.902_{\pm .005}$	$.888_{\pm .008}$	$.653_{\pm .013}$	OOM
STaSy	$.906_{\pm .001}$	$.752_{\pm .006}$	$.914_{\pm .005}$	$.934_{\pm .003}$	$.656_{\pm .014}$	$.871_{\pm .002}$
CoDi	$.871_{\pm .006}$	$.525_{\pm .006}$	$.865_{\pm .006}$	$.932_{\pm .003}$	$.818_{\pm .021}$	$1.21_{\pm .005}$
TabDDPM[2]	$.907_{\pm .001}$	$.758_{\pm .004}$	$.918_{\pm .005}$	$.935_{\pm .003}$	$.592_{\pm .011}$	$4.86_{\pm 3.04}$
TabSyn	$.915_{\pm .002}$	$.764_{\pm .004}$	$.920_{\pm .005}$	$.938_{\pm .002}$	$.582_{\pm .008}$	$.861_{\pm .027}$



Classification/Regression as Missing Value Imputation

VECTOR INSTITUTE

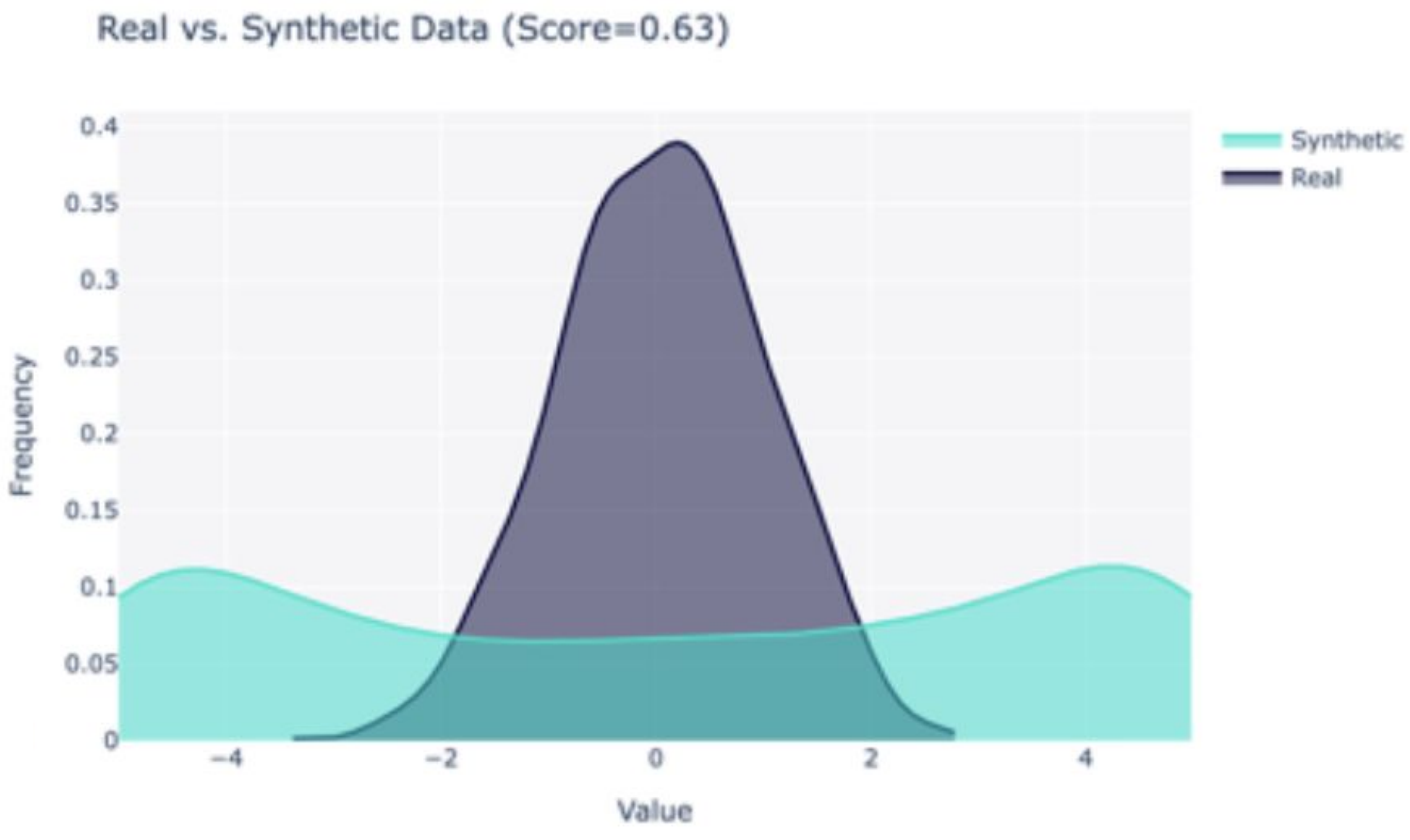
- Mask target column by replacing them with average value of respective column in train data
- Apply trained TabSyn to impute the masked values
- Compare the performance with directly training a discriminative model (classifier or regressor)
- Generative models tend to less face the overfitting phenomena compared to discriminative models

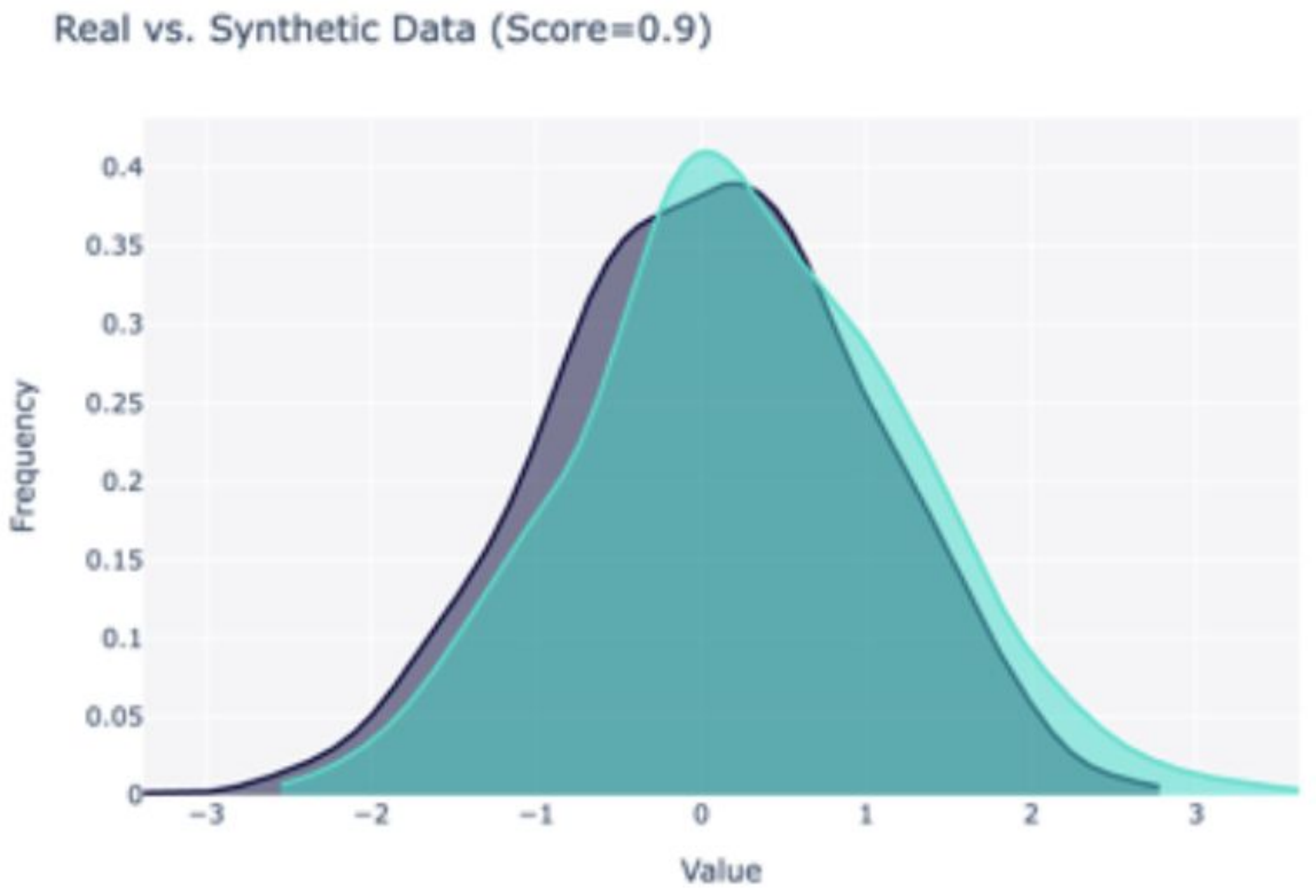
Methods	Adult	Default	Shoppers	Magic	Beijing	News
	AUC ↑	AUC ↑	AUC ↑	AUC ↑	RMSE ↓	RMSE ↓
Real with XGBoost	92.7	77.0	92.6	**94.6**	0.423	**0.842**
Impute with TabSyn	**93.2**	**87.2**	**96.6**	88.8	**0.258**	1.253

Low-order Statistics

- **Single column similarity score** measures if each column of synthetic data captures the same density distribution of each column of real data
- The average error percentage for **single column similarity score**

Method	Adult	Default	Shoppers	Magic	Beijing	News
SMOTE	1.60±0.23	1.48±0.15	2.68±0.19	0.91±0.05	1.85±0.21	5.31±0.46
CTGAN	16.84± 0.03	16.83±0.04	21.15±0.10	9.81±0.08	21.39±0.05	16.09±0.02
TVAE	14.22±0.08	10.17±0.05	24.51±0.06	8.25±0.06	19.16±0.06	16.62±0.03
GOGGLE[1]	16.97	17.02	22.33	1.90	16.93	25.32
GReaT[2]	12.12±0.04	19.94±0.06	14.51±0.12	16.16±0.09	8.25±0.12	OOM
STaSy	11.29±0.06	5.77±0.06	9.37±0.09	6.29±0.13	6.71±0.03	6.89±0.03
CoDi	21.38±0.06	15.77± 0.07	31.84±0.05	11.56±0.26	16.94±0.02	32.27±0.04
TabDDPM[3]	1.75±0.03	1.57± 0.08	2.72±0.13	1.01±0.09	1.30±0.03	78.75±0.01
TabSyn	0.58±0.06	0.85±0.04	1.43±0.24	0.88±0.09	1.12±0.05	1.64±0.04
Improv.	66.9% ↓	45.9% ↓	47.4% ↓	12.9% ↓	13.8% ↓	76.2% ↓



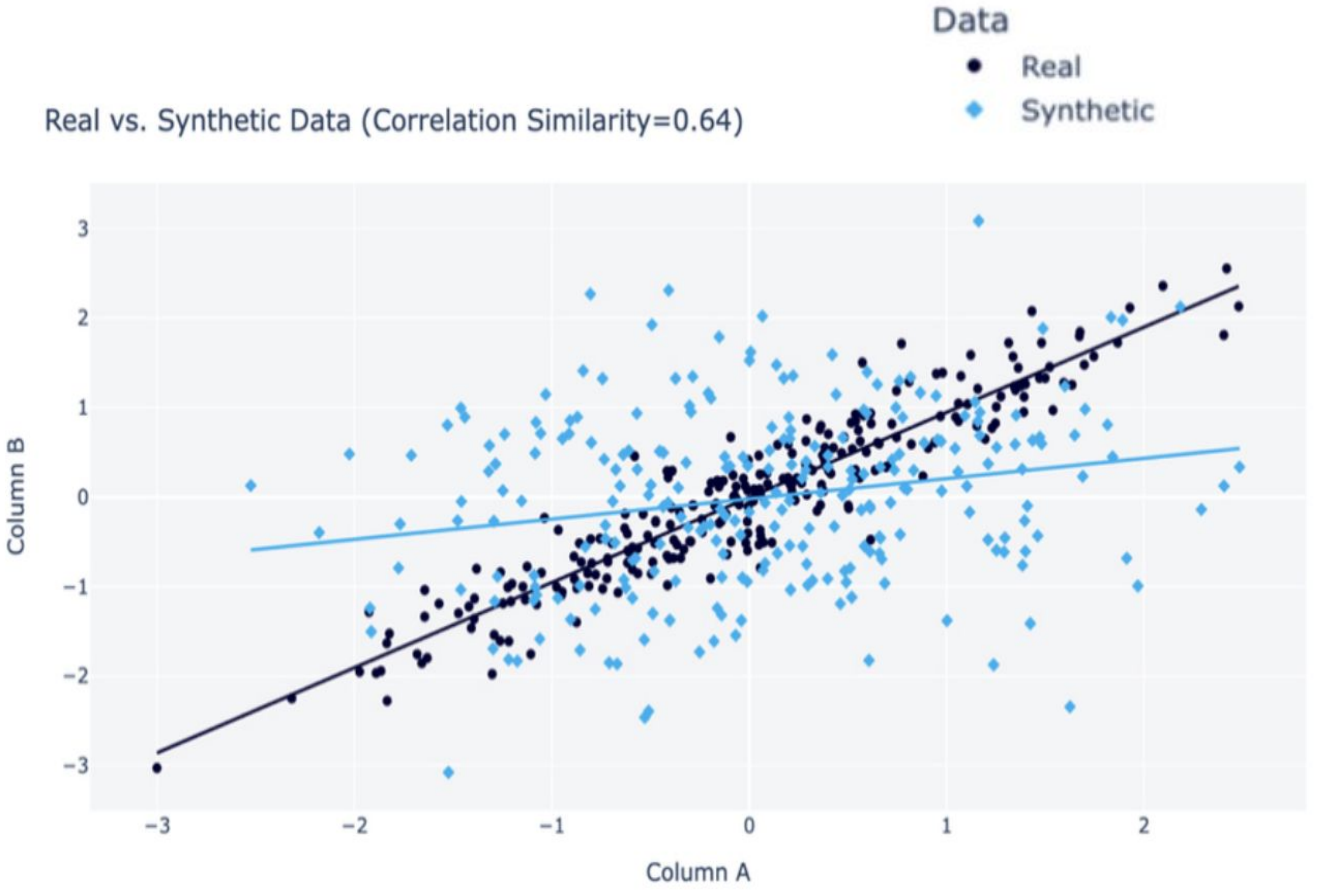


Low-order Statistics

VECTOR INSTITUTE

- **Pair-wise correlation score** measure the correlation between pairs of columns of real and synthetic data
- The average error percentage for **pair-wise correlation score**

Method	Adult	Default	Shoppers	Magic	Beijing	News
SMOTE	3.28±0.29	8.41±0.38	3.56±0.22	3.16±0.41	2.39±0.35	5.38±0.76
CTGAN	20.23±1.20	26.95±0.93	13.08±0.16	7.00±0.19	22.95±0.08	5.37±0.05
TVAE	14.15±0.88	19.50±0.95	18.67±0.38	5.82±0.49	18.01±0.08	6.17±0.09
GOGGLE	45.29	21.94	23.90	9.47	45.94	23.19
GReaT	17.59±0.22	70.02±0.12	45.16±0.18	10.23±0.40	59.60±0.55	OOM
STaSy	14.51±0.25	5.96±0.26	8.49±0.15	6.61±0.53	8.00±0.10	3.07±0.04
CoDi	22.49±0.08	68.41±0.05	17.78±0.11	6.53±0.25	7.07±0.15	11.10±0.01
TabDDPM	3.01±0.25	4.89±0.10	6.61±0.16	1.70±0.22	2.71±0.09	13.16±0.11
TabSyn	1.54±0.27	2.05±0.12	2.07±0.21	1.06±0.31	2.24±0.28	1.44±0.03
Improve.	48.8% ↓	58.1% ↓	68.7% ↓	37.6% ↓	17.3% ↓	53.1% ↓

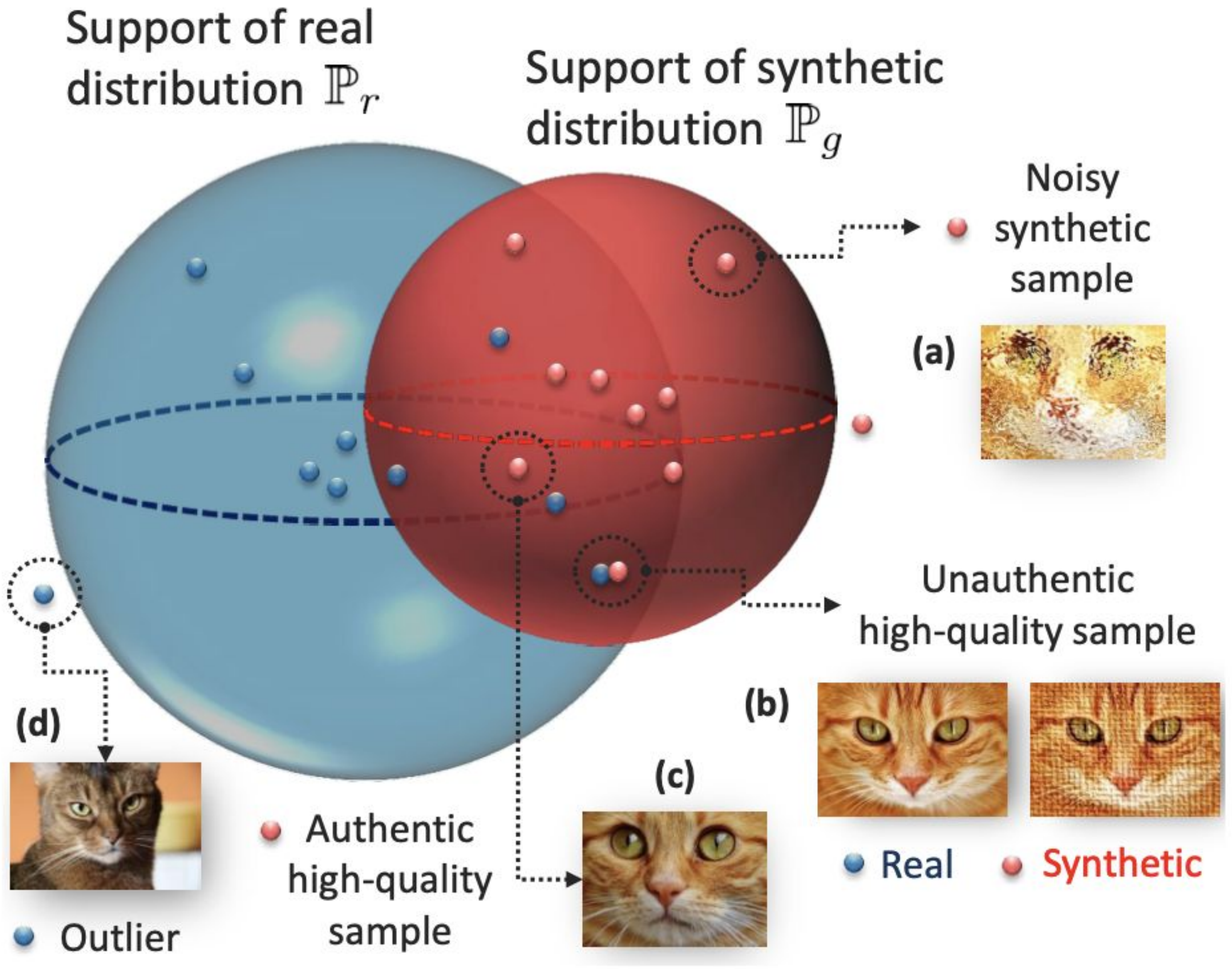


High-order Statistics

VECTOR INSTITUTE

- α-precision to measure fidelity of synthetic data
  - The fraction of synthetic data that resemble the most "typical" real data (Red circle)

Methods	Adult	Default	Shoppers	Magic	Beijing	News	Average
CTGAN	77.74±0.15	62.08±0.08	76.97±0.39	86.90±0.22	96.27±0.14	96.96±0.17	82.82
TVAE	98.17±0.17	85.57±0.34	58.19±0.26	86.19±0.48	97.20±0.10	86.41±0.17	85.29
GOGGLE	50.68	68.89	86.95	90.88	88.81	86.41	78.77
GReaT	55.79±0.03	85.90±0.17	78.88±0.13	85.46±0.54	98.32±0.22	-	80.87
STaSy	82.87±0.26	90.48±0.11	89.65±0.25	86.56±0.19	89.16±0.12	94.76±0.33	88.91
CoDi	77.58±0.45	82.38±0.15	94.95±0.35	85.01±0.36	98.13±0.38	87.15±0.12	87.03
TabDDPM	96.36±0.20	97.59±0.36	88.55±0.68	98.59±0.17	97.93±0.30	0.00±0.00	79.83
TabSyn	99.52±0.10	99.26±0.27	99.16±0.22	99.38±0.27	98.47±0.10	96.80±0.25	98.67

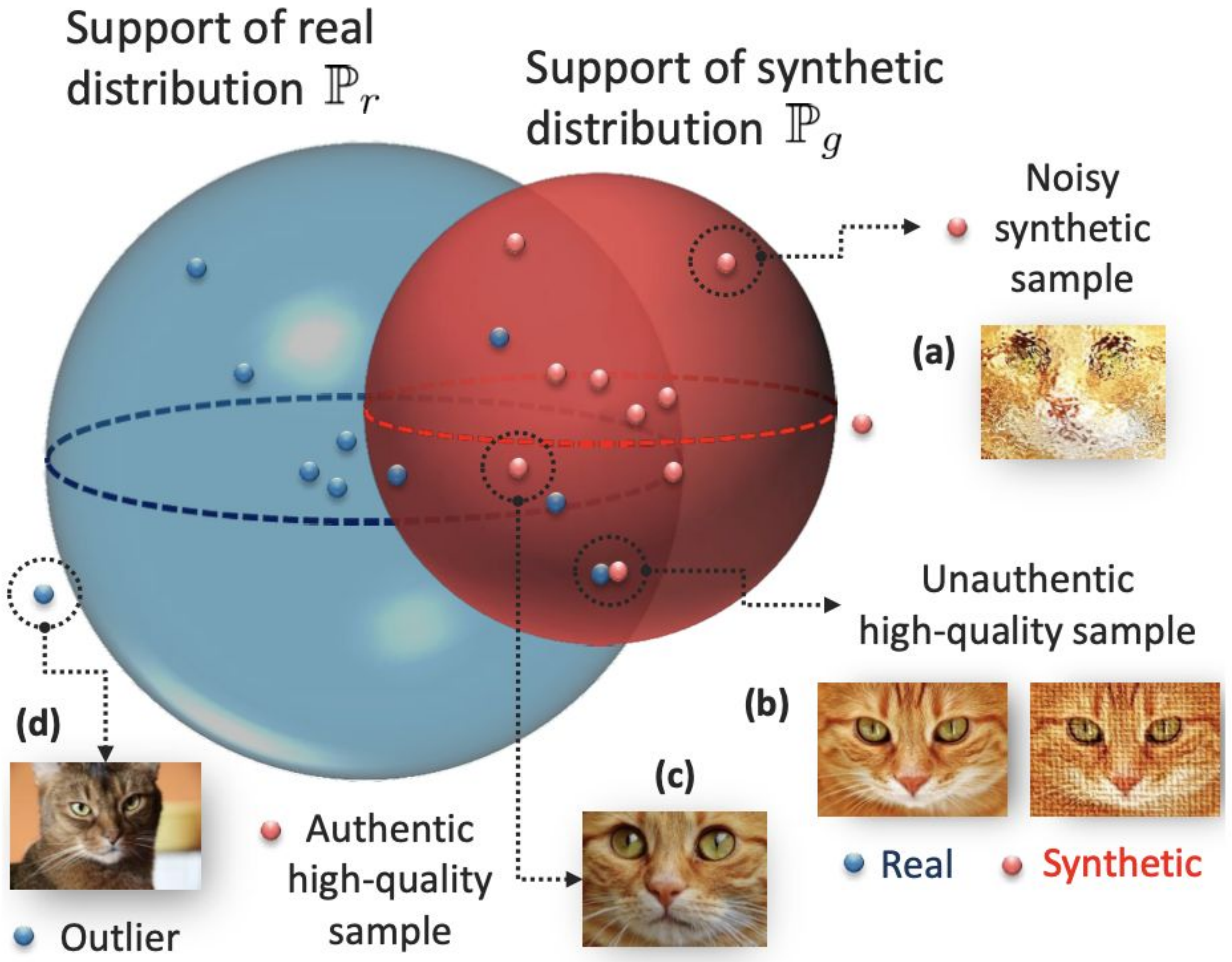


High-order Statistics

VECTOR INSTITUTE

- α-precision to measure fidelity of synthetic data
  - The fraction of synthetic data that resemble the most typical real data (Red circle)
- β-recall scores that measure the diversity of synthetic data
  - The fraction of real data covered by the most typical synthetic data (Blue circle)

Methods	Adult	Default	Shoppers	Magic	Beijing	News	Average
CTGAN	30.80±0.20	18.22±0.17	31.80±0.350	11.75±0.20	34.80±0.10	24.97±0.29	25.39
TVAE	38.87±0.31	23.13±0.11	19.78±0.10	32.44±0.35	28.45±0.08	29.66±0.21	28.72
GOGGLE	8.80	14.38	9.79	9.88	19.87	2.03	10.79
GReaT	49.12±0.18	42.04±0.19	44.90±0.17	34.91±0.28	43.34±0.31	-	43.34
STaSy	29.21±0.34	39.31±0.39	37.24±0.45	53.97±0.57	54.79±0.18	39.42±0.32	42.32
CoDi	9.20±0.15	19.94±0.22	20.82±0.23	50.56±0.31	52.19±0.12	34.40±0.31	31.19
TabDDPM	47.05±0.25	47.83±0.35	47.79±0.25	48.46±0.42	56.92±0.13	0.00±0.00	41.34
TabSyn	47.56±0.22	48.00±0.35	48.95±0.28	48.03±0.23	55.84±0.19	45.04±0.34	48.90



Detection Score

VECTOR INSTITUTE

- Classifier two sample Test (C2ST) to detect whether synthetic data can be detected from real data:
  1. Create a single, augmented table that has all the rows of real data and all the rows of synthetic data eith an extra column to keep track of whether each original row is real or synthetic.
  2. Split the augmented data to create a training and validation sets.
  3. Train the model on the training split to predict whether each row is real or synthetic
  4. Compute ROC-AUC score on validation set
  5. Repeat steps #2-4 multiple times and average ROC-AUC score.
  6. Compute final score → $score = 1 - (max(\overline{\text{ROC-AUC}}, 0.5) \times 2 - 1)$.

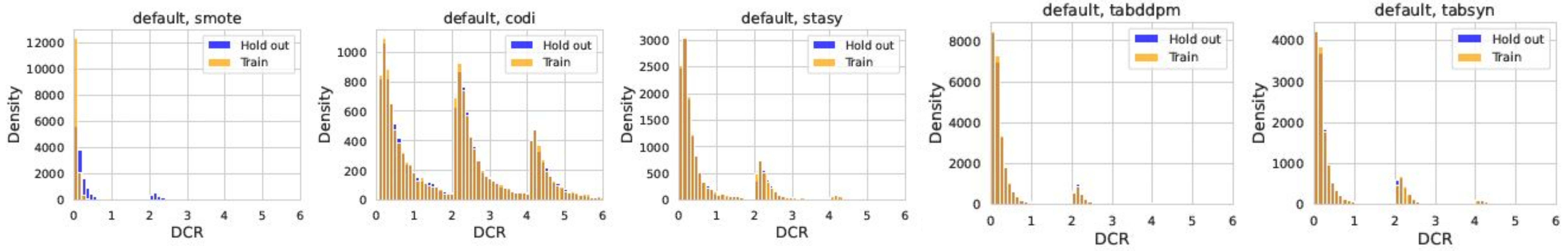
Method	Adult	Default	Shoppers	Magic	Beijing	News
SMOTE	0.9710	0.9274	0.9086	0.9961	0.9888	0.9344
CTGAN	0.5949	0.4875	0.7488	0.6728	0.7531	0.6947
TVAE	0.6315	0.6547	0.2962	0.7706	0.8659	0.4076
GOGGLE	0.1114	0.5163	0.1418	0.9526	0.4779	0.0745
GReaT	0.5376	0.4710	0.4285	0.4326	0.6893	-
STaSy	0.4054	0.6814	0.5482	0.6939	0.7922	0.5287
CoDi	0.2077	0.4595	0.2784	0.7206	0.7177	0.0201
TabDDPM	0.9755	0.9712	0.8349	**0.9998**	0.9513	0.0002
TabSyn	**0.9986**	**0.9870**	**0.9740**	0.9732	**0.9603**	**0.9749**

Privacy Protection

VECTOR INSTITUTE

- Evaluate if the synthetic data is randomly sampled according to the distribution density rather than copied from the training data via Compute Distance to Closest Records (DCR)
- DCR is the Euclidean distance between synthetic data point and nearest real data point
- This score reported as portion of synthetic data point that are closer to training data rather than test data
- For an equal size of train and test this score should be close to 0.5

Method	Default
SMOTE	$91.41\%_{\pm 3.42}$
STaSy	$50.23\%_{\pm 0.09}$
CoDi	$51.82\%_{\pm 0.26}$
TabDDPM	$52.15\%_{\pm 0.20}$
TabSyn	$51.20\%_{\pm 0.18}$



Next Steps and Q&A

VECTOR INSTITUTE