Evaluating Deep Generative Models

A $p(reality)$ check

Joey Bose

Generative Modelling

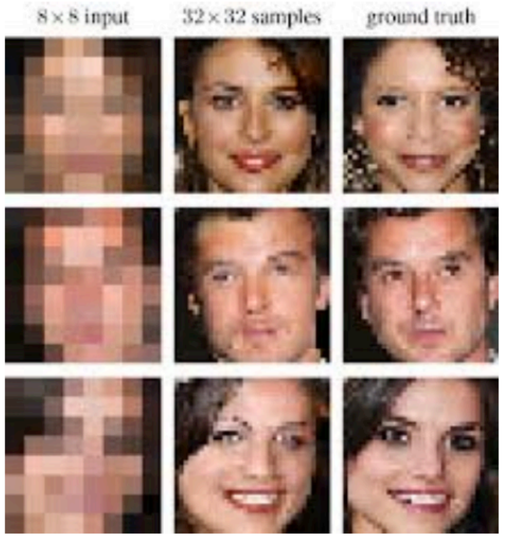
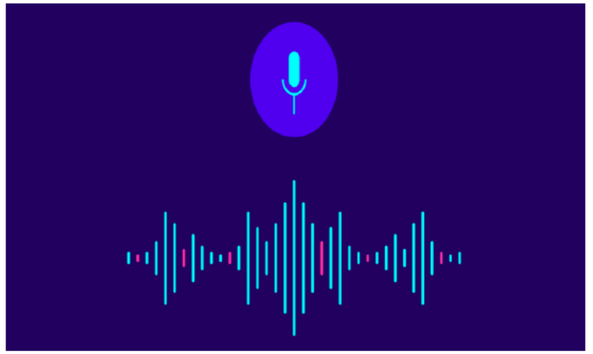
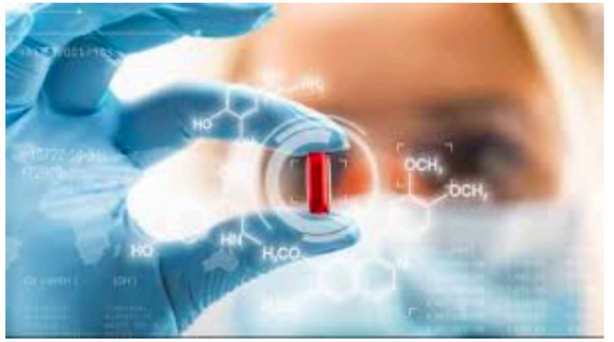


Image Super Resolution



Text to Speech

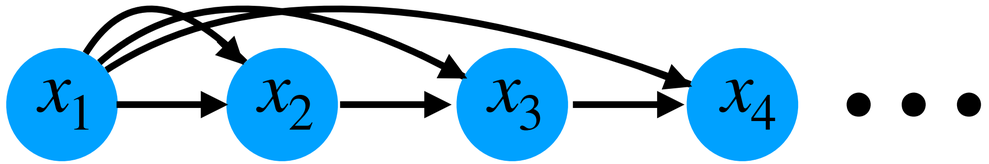


Drug Discovery

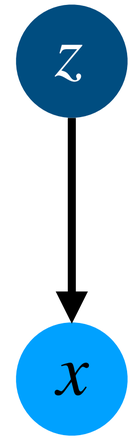
“Creating noise from data is easy; creating data from noise is generative modeling.”
Song et. Al 2020

Types of Generative Models

- Fully-observed models



- Latent Variable Models
  - Prescribed Models: Likelihood + noise
  - Implicit Models (Likelihood-free)



- Score Matching

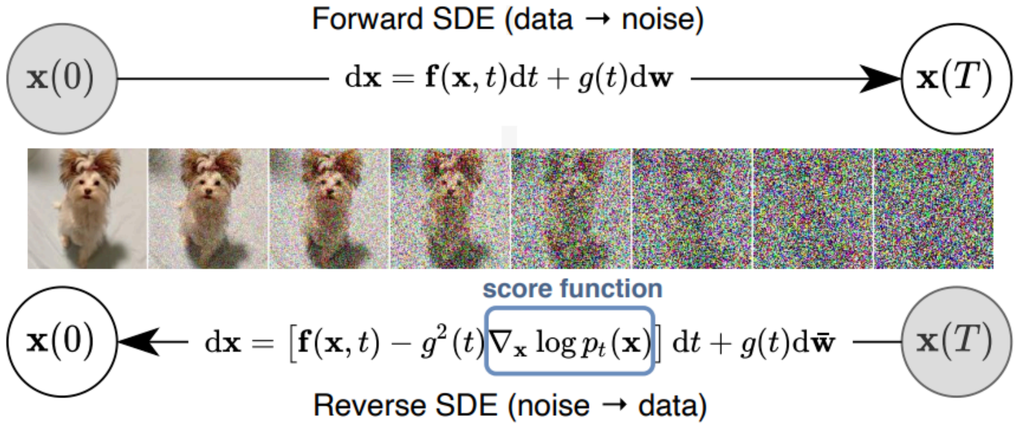


Fig Credit: Song et. Al 2021

Evaluating Generative Models

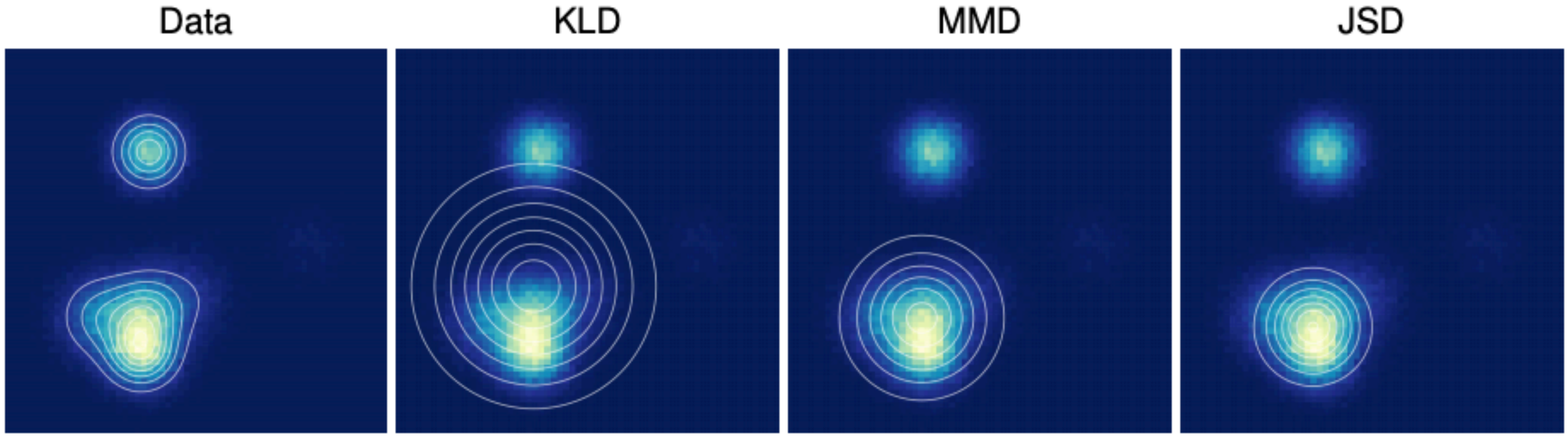
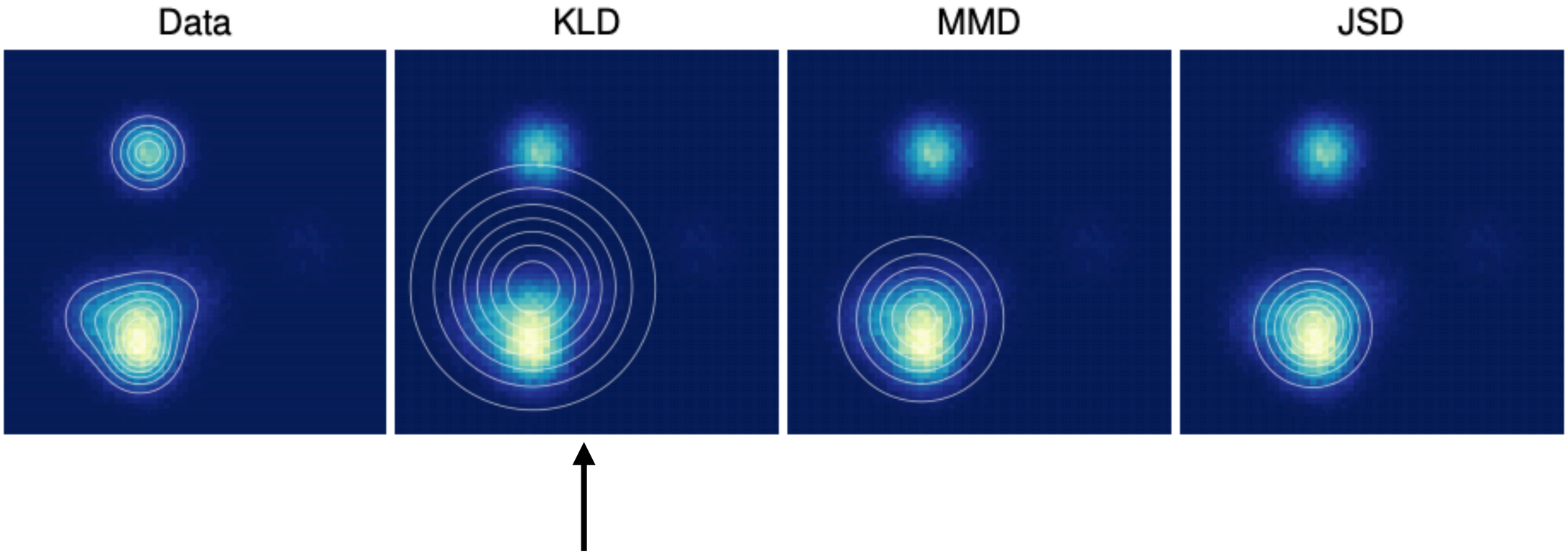


Fig Credit: Theis et. al 2016

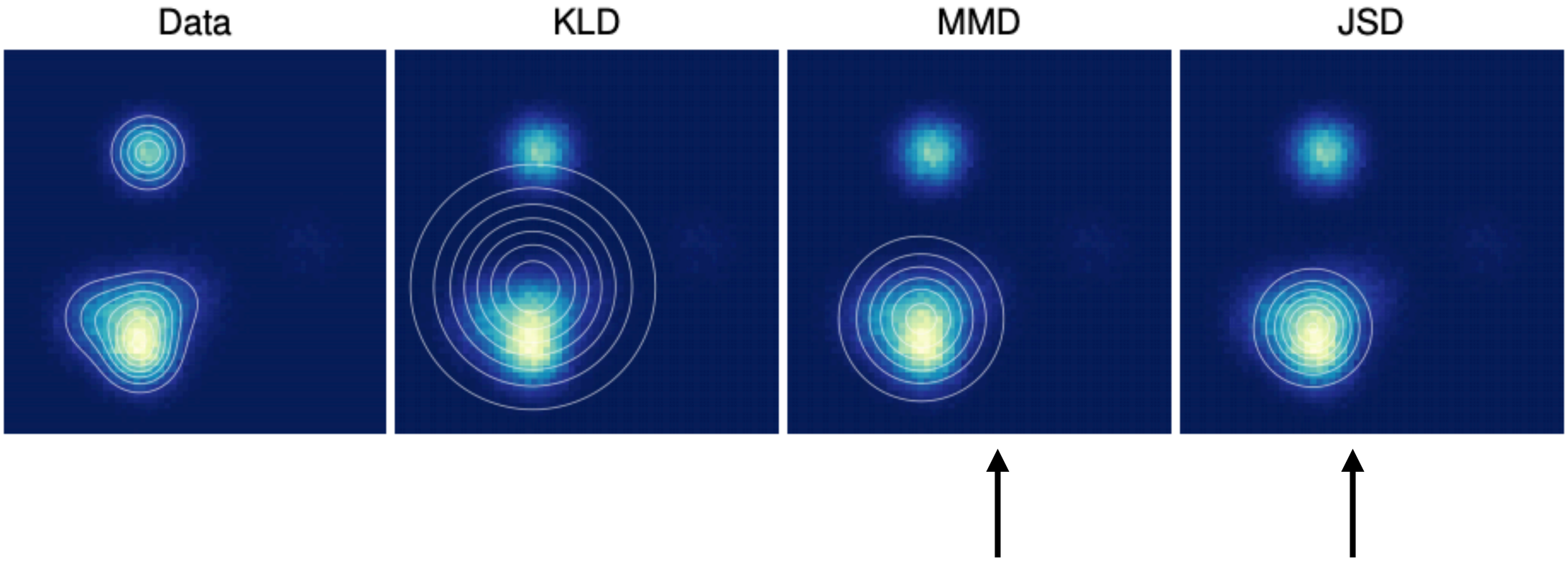
Evaluating Generative Models



Avoids assigning extremely small probability to any data point but assigns a lot of probability mass to non-data regions.

Fig Credit: Theis et. al 2016

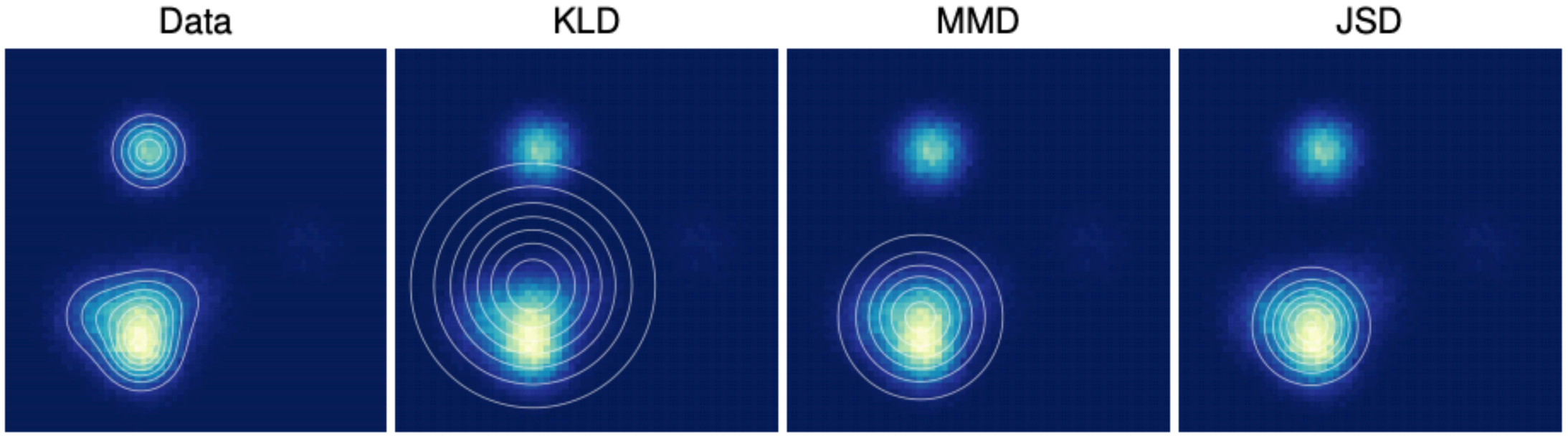
Evaluating Generative Models



Fits one mode really well but ignores other parts of the data

Fig Credit: Theis et. al 2016

Evaluating Generative Models



Different Evaluation Mesures have Different Tradeoffs!

Fig Credit: Theis et. al 2016

Evaluating Generative Models

Key question: What is the task you care about?

- Density Estimation
- Sampling/Generation
- Semi-supervised Learning
- Learning useful representations
- Anomaly Detection
- Hybrid Modeling
- Calibration

Evaluating Generative Models

Key question: What is the task you care about?

- Density Estimation
- Sampling/Generation
- Semi-supervised Learning
- Learning useful representations
- Anomaly Detection
- Hybrid Modeling
- Calibration

Most research papers do this

Evaluating Generative Models

Key question: What is the task you care about?

- Density Estimation
- Sampling/Generation

→ Most research papers do this

- Semi-supervised Learning
- Learning useful representations

→ Others claim their models do a good job of this

- Anomaly Detection
- Hybrid Modeling
- Calibration

Evaluating Generative Models

Key question: What is the task you care about?

- Density Estimation
- Sampling/Generation

→ Most research papers do this

- Semi-supervised Learning
- Learning useful representations

→ Others claim their models do a good job of this

- Anomaly Detection
- Hybrid Modeling
- Calibration

→ Hope we will magically do well on these.

Evaluating Generative Models

Key question: What is the task you care about?

- Density Estimation
- Sampling/Generation

→ Most research papers do this

- Semi-supervised Learning
- Learning useful representations

→ Others claim their models do a good job of this

- Anomaly Detection
- Hybrid Modeling
- Calibration

→ Hope we will magically do well on these.

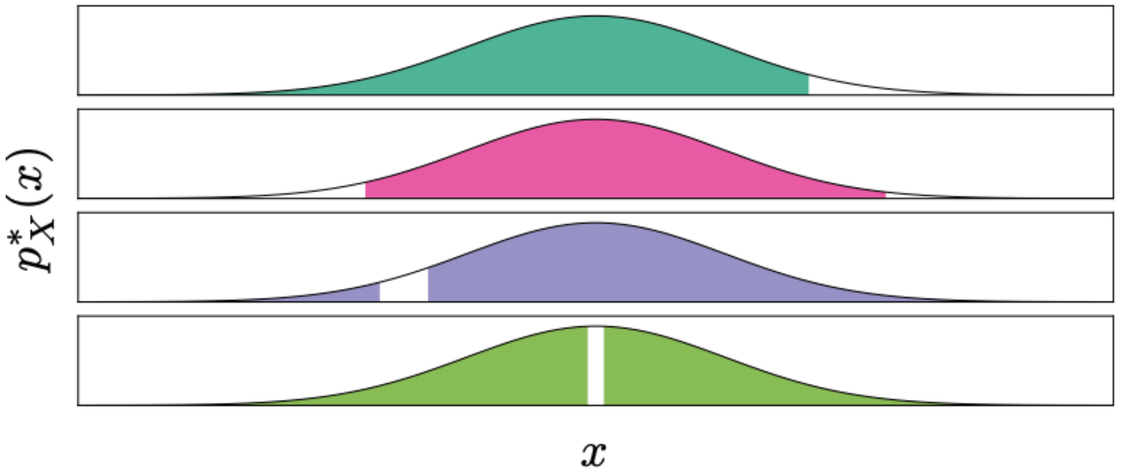
How much progress have we actually made on these tasks?

Explicit Likelihood Models for Anomaly Detection

So you have an explicit likelihood model and you want to do Anomaly Detection? Should be simple right?

Problem Definition

- Input space $\mathcal{X} \subseteq \mathbb{R}^D$
- Partitioning $\mathcal{X}_{in} \cup \mathcal{X}_{out} = \mathcal{X}$
- $\mathcal{X}_{in} \cap \mathcal{X}_{out} = \emptyset$



Pick $\mathcal{X}_{in}$ such that it contains “the majority” of the mass

e.g. $P_X^*(\mathcal{X}_{in}) = 1 - \alpha \in (0.5,1)$

Fig Credit: Le Lan and Dinh 2021

Approaches for Anomaly Detection

Density Scoring

Unlikely samples should have “low” likelihood —i.e. low $p^*_X(x)$

A heuristic is to define inliers as points whose density is greater than a threshold $\lambda > 0$

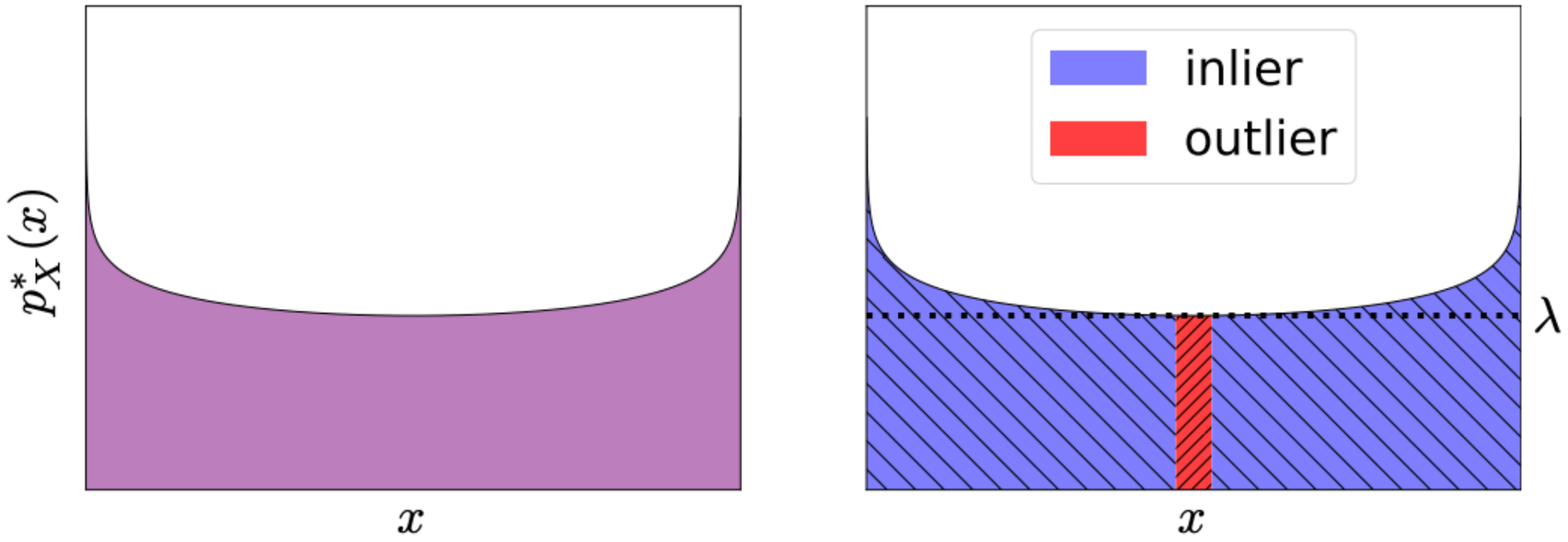


Fig Credit: Le Lan and Dinh 2021

Approaches for Anomaly Detection

Typicality Test

Inliers are part of typical set —-i.e. points whose average log density is close to the average log-density of the distribution $\lambda > 0$. Loose definition, the typical set $A_\epsilon^{(N)}(p_X^*) \subset \mathcal{X}^N$ of points satisfy:

$$\left| H(p^*x) + \frac{1}{N}\sum_{n=1}^{N} \log p_X^*(x^{(n)}) \right| \leq \epsilon$$

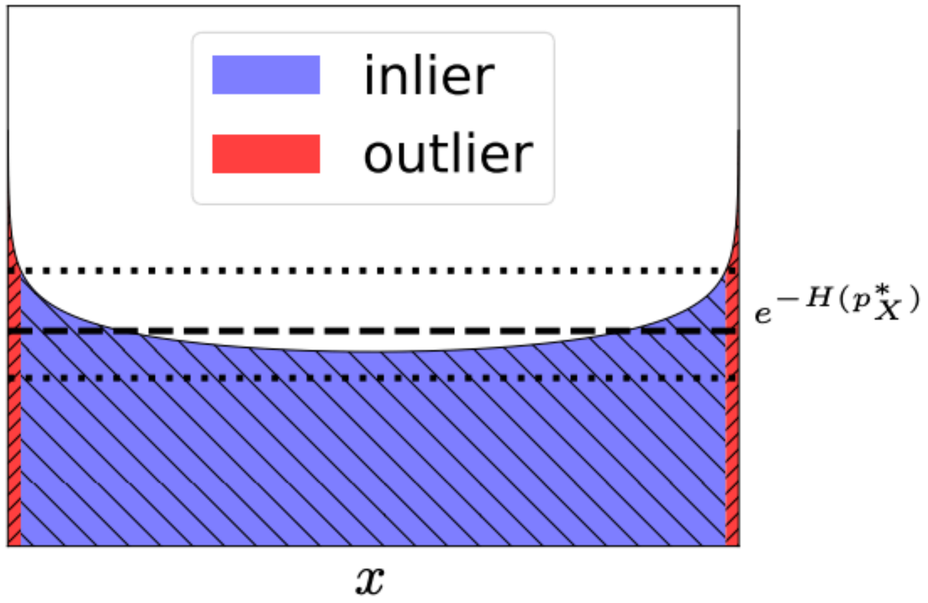


Fig Credit: Le Lan and Dinh 2021

Approaches for Anomaly Detection

Assume we have trained the perfect density model under infinite data and capacity: $p_X^{(\theta)} = p_X^*$

The observed data is a chosen representation of the real data and should remain invariant under an invertible map f (There is no loss of information).

$$c^*(x)$$

Perfect Classifier on X

$$(c^* \circ f^{-1})(f(x))$$

Perfect Classifier on $f(X)$

Approaches for Anomaly Detection

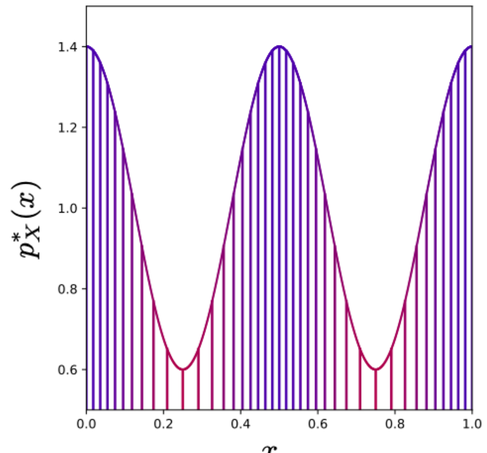
Principle. *In an infinite data and capacity setting, the result of an anomaly detection should be invariant to continuous reparametrization.*

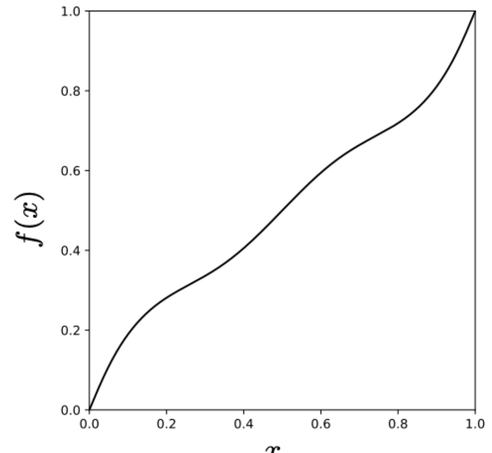
$\mathcal{X}_{out}$ remains a low probability subset as

$$P_X(\mathcal{X}_{out}) = P_{f(X)}(f(\mathcal{X}_{out})) \quad \text{and} \quad \forall x \in \mathcal{X}, x \in \mathcal{X}_{out} \iff f(x) \in f(\mathcal{X}_{out})$$

$$p^*_{f(X)}(f(x)) = p^*_X(x) \left| \frac{\partial f}{\partial x^T}(x) \right|^{-1}$$

→ Change density under an invertible map





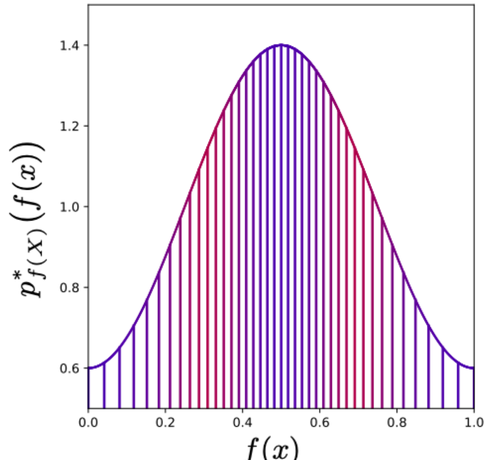
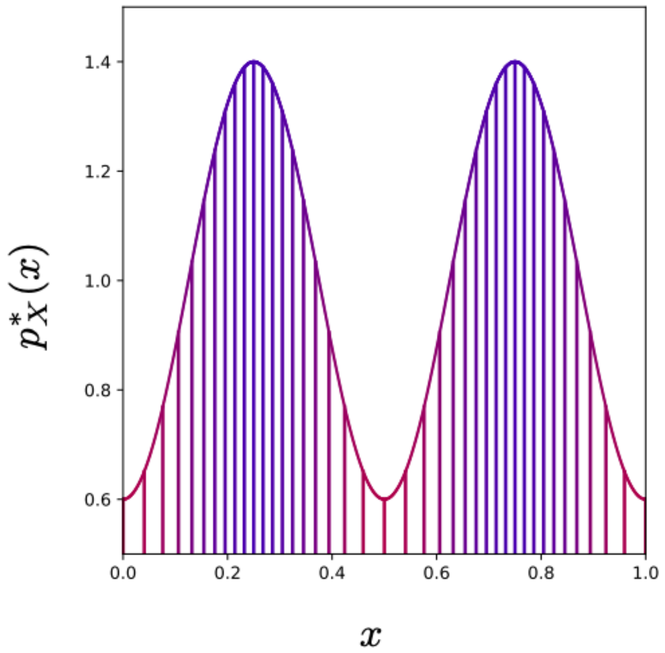


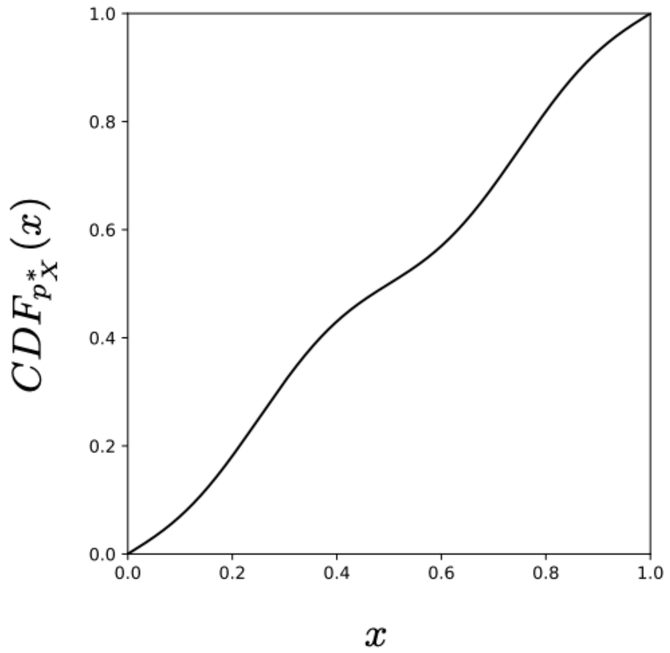
Fig Credit: Le Lan and Dinh 2021

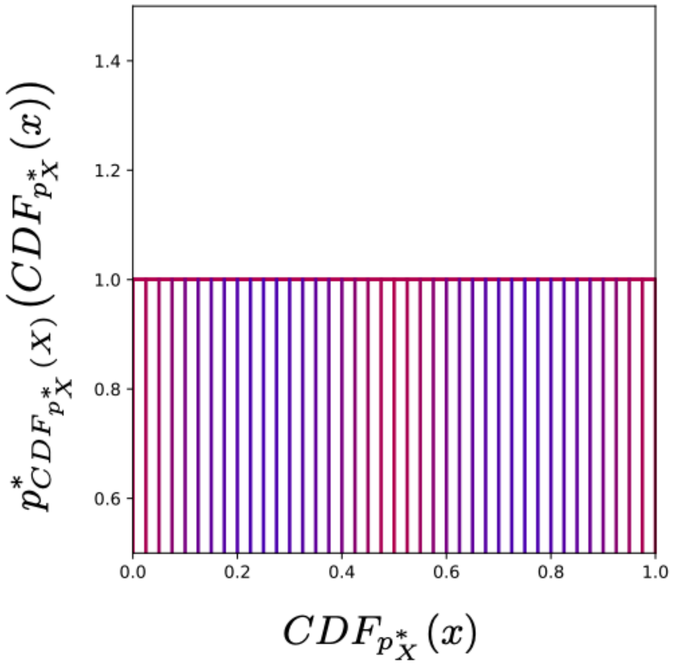
Anomaly Detection: Uniformization

Under weak assumptions one can map any distribution to an uniform one with an invertible map f^{KR}. $p^*_{f^(KR)(X)} = 1$ is constant everywhere.

$$\forall d \in \{1,\cdots,D\}, f^{(KR)}(x) = CDF_{p^* x_{d|X_{<d}}}(x_d \,|\, x_{<d})$$







Anomaly Detection: Arbitrary Score

Proposition 1(Le Lan and Dinh 2021). For any R.V. $X \sim p_X^*$ with p_X^* continuous strictly positive (with $\mathcal{X}$ convex) and any measurable continuous map $s : \mathcal{X} \to \mathbb{R}_+^*$ bounded below by a strictly positive number, there exists a continuous bijection $f^{(s)}$ s.t. $x \in \mathcal{X}, p_{f^{(s)}(X)}(f^{(s)}(x)) = s(x)$

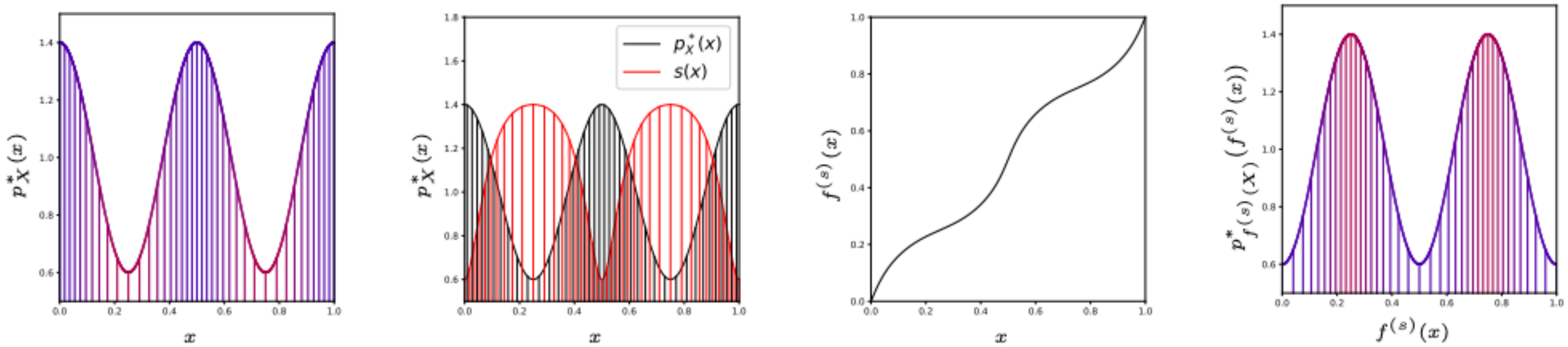


Fig Credit: Le Lan and Dinh 2021

Hybrid Modeling

What if we want to use generative models for discriminative tasks?

Generative Part

$$\log p(x, y) = \log p(y \mid x) + \log p(x)$$

Discriminative part

The discriminative part will assign a score to any input even if the point is actually not part of the data distribution. Can we balance this with the generative part? Can we use any pre-trained generative model to help here?

Hybrid Modeling: Naive Approach

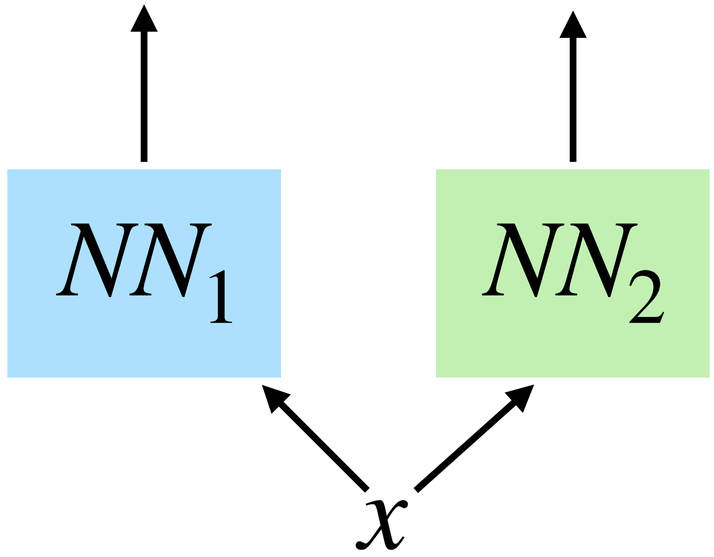
If the discriminative and generative parts were trained separately we have:

Generative Part

$$\log p(x, y) = \log p_{\theta_1}(y \mid x) + \log p_{\theta_2}(x)$$

Discriminative part

Operationally we can use separate networks



Hybrid Modeling: Naive Approach

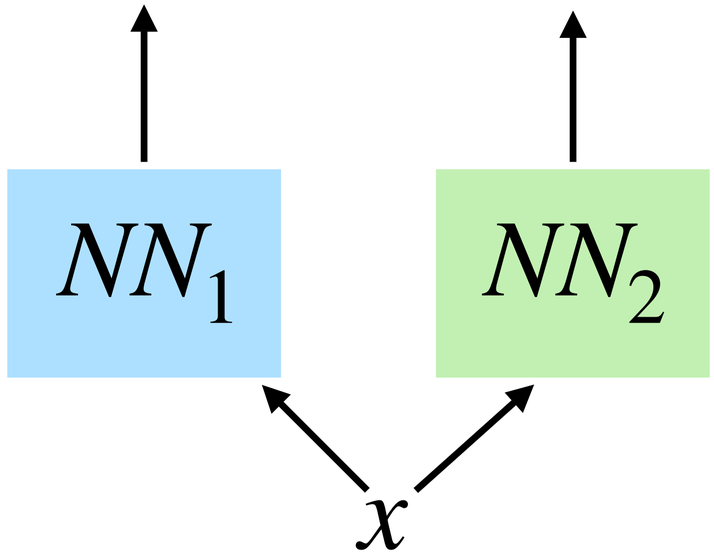
If the discriminative and generative parts were trained separately we have:

Generative Part

$$\log p(x, y) = \log p_{\theta_1}(y \mid x) + \log p_{\theta_2}(x)$$

Discriminative part

There is no information flow between x and y. There is no reason each NN treats x in the same way!



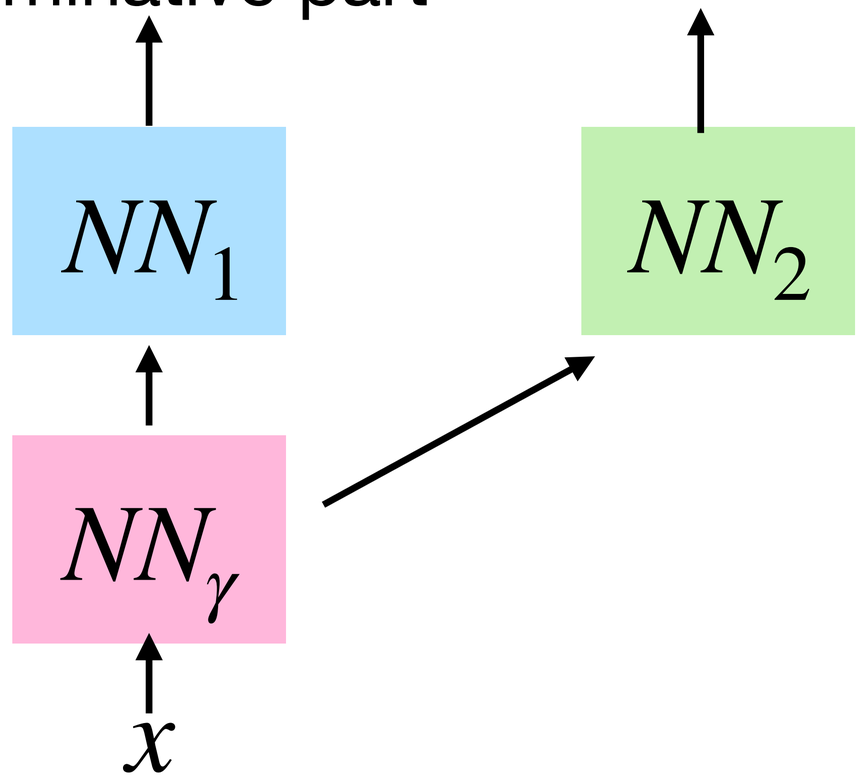
Hybrid Modeling

What if we share parameters

Generative Part

$$\log p(x, y) = \log p_{\theta_1,\gamma}(y \mid x) + \log p_{\theta_2,\gamma}(x)$$

Discriminative part



Hybrid Modeling: Parameter Sharing

If y is a binary label it needs only 1 bit while for a D-dim x vector we need at least as many bits.

$$\nabla_\gamma logp(x, y) = \nabla_\gamma \log p_{\theta_1,\gamma}(y \mid x) + \nabla_\gamma \log p_{\theta_2,\gamma}(x)$$

Let's also assume that the output of both discriminative and generative nets is 0.5. What is the

$$\log p_{\theta_1,\gamma}(y \mid 0.5) = y \log 0.5 + (1 - y) \log 0.5$$

Hybrid Modeling: Parameter Sharing

$$\log Bern(y \mid 0.5) = y \log 0.5 + (1 - y) \log 0.5$$
$$= -\log 2$$

$$\log \prod_{i}^{D} Bern(x_d \mid 0.5) = \sum_{i}^{D} \log Bern(x_d \mid 0.5)$$
$$= -D \log 2$$

Hybrid Modeling: How do we fix this?

- Convex combination of the two objectives (Bouchard & Triggs, 2004)
- Up weight the generative part by a positive factor $\lambda \geq 0$ (Nalisnick et al., 2019)
- Use an invertible network and a learned flow prior (Chen et al., 2019)

Take home message: If this is your end downstream task it can't be treated as two separate problems. The generative modelling part must be tightly coupled with the end task!

Generative Models as Lossy Compression

Shannon's fundamental compression theorem states that we can compress a random variable $x \sim p(x)$ losslessly at $\mathscr{H}(x)$. That is a good generative model should be able to "transmit" a coded input with a rate close the entropy for optimal reconstruction.

What about lossy compression? Suppose we allow a compression rate $R \leq \mathscr{H}(x)$ using a code z and have a lossy reconstruction $\hat{x} = f(z)$. What is the tradeoff now?

Generative Models as Lossy Compression

Given a distortion threshold D Shannon's rate distortion theorem states that the rate distortion function $\mathscr{R}(D)$ equals the minimum of the following:

$$\min_{q(z|x)} \mathscr{I}(x,z) \quad s.t. \mathbb{E}_{q(x,z)}[d(x,f(z))] \leq D$$

The encoder distribution or "approximate posterior"

Joint distribution

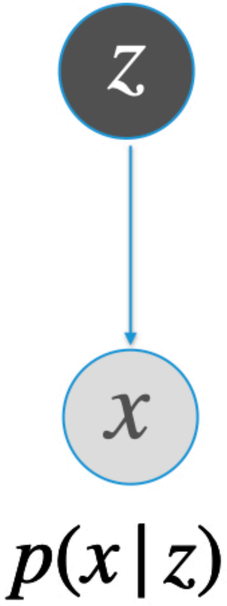
Latent Variable modeling & Variational Inference

$$\log p(\mathscr{D}) = \sum_{i=1}^{N} \log \mathbb{E}_{p(z)}[p(x_i \mid z)]$$

$$\log p(\mathscr{D}) \geq \sum_{i=1}^{N} \mathbb{E}_{q_i(z)}\left[\log \frac{p(x \mid z)p(z)}{q_i(z)}\right]$$

$$\mathbb{E}_{q_i(z)}\left[\log \frac{p(x \mid z)p(z)}{q_i(z)}\right] = \mathbb{E}_{q_i(z)}[\log p(x_i \mid z)] - D_{KL}(q_i \mid\mid p)$$

Reconstruction — Regularizer



Lossy Compression in VAE's

We can modify the formalism of rate distortion theory to match the generative model formalism

$$\mathcal{I}(x,z) \leq \mathcal{I}(x,z) + KL(q(z)\,||\,p(z))$$
$$= \mathbb{E}_{p_d(x)}[KL(q(z\,|\,x)\,||\,p(z))]$$

↑

"approximate posterior"

$$\min_{q(z|x)} \mathbb{E}_{p_d(x)}[KL(q(z\,|\,x)\,||\,p(z))] \quad s.t. \; \mathbb{E}_{q(x,z)}[d(x,f(z))] \leq D$$

Variational Rate Distortion function: $\mathcal{R}_p(D)$

Lossy Compression in VAE's

We can change the constrained optimization using the method of long-range multipliers:

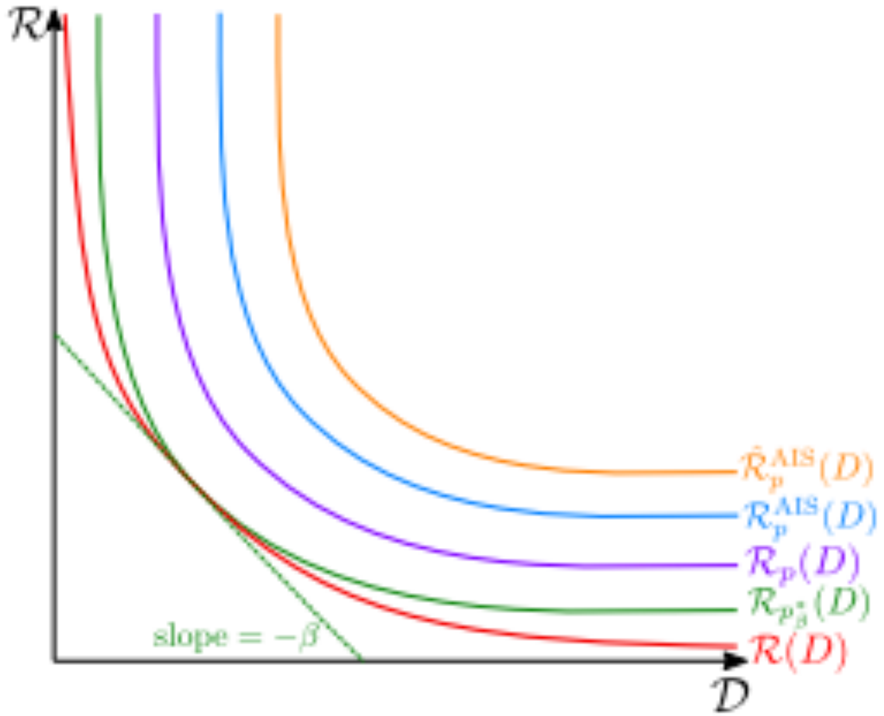
$$\min_{q(z|x)} \mathbb{E}_{p_d(x)}[KL(q(z\,|\,x)\,||\,p(z))] \quad s.t. \; \mathbb{E}_{q(x,z)}[d(x, f(z))] \leq D$$

$$\downarrow$$

$$\min_{q(z|x)} \mathbb{E}_{p_d(x)}[KL(q(z\,|\,x)\,||\,p(z))] + \beta \mathbb{E}_{q(x,z)}[d(x, f(z))] \leq D$$

Independent optimization problems for each x

Rate Distortion Curves



- $\mathscr{R}_p(D)$ is an upper bound for any prior on $\mathscr{R}(D)$
- $\mathscr{R}(D) = \min_{p(z)} \mathscr{R}(D)$ which means for any β there's an optimal prior

Fig Credit: Huang et. al 2020

Rate Distortion Curves

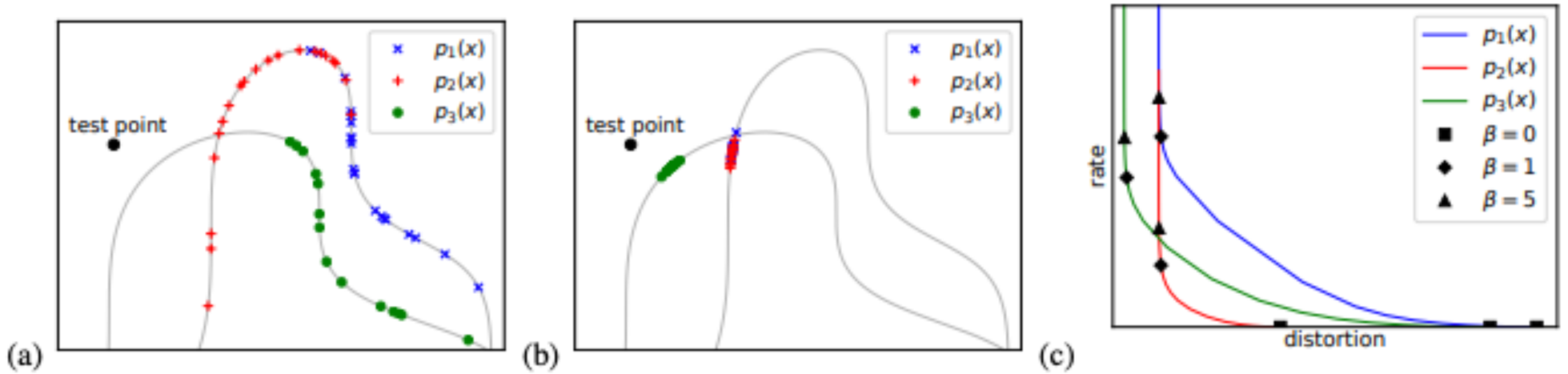
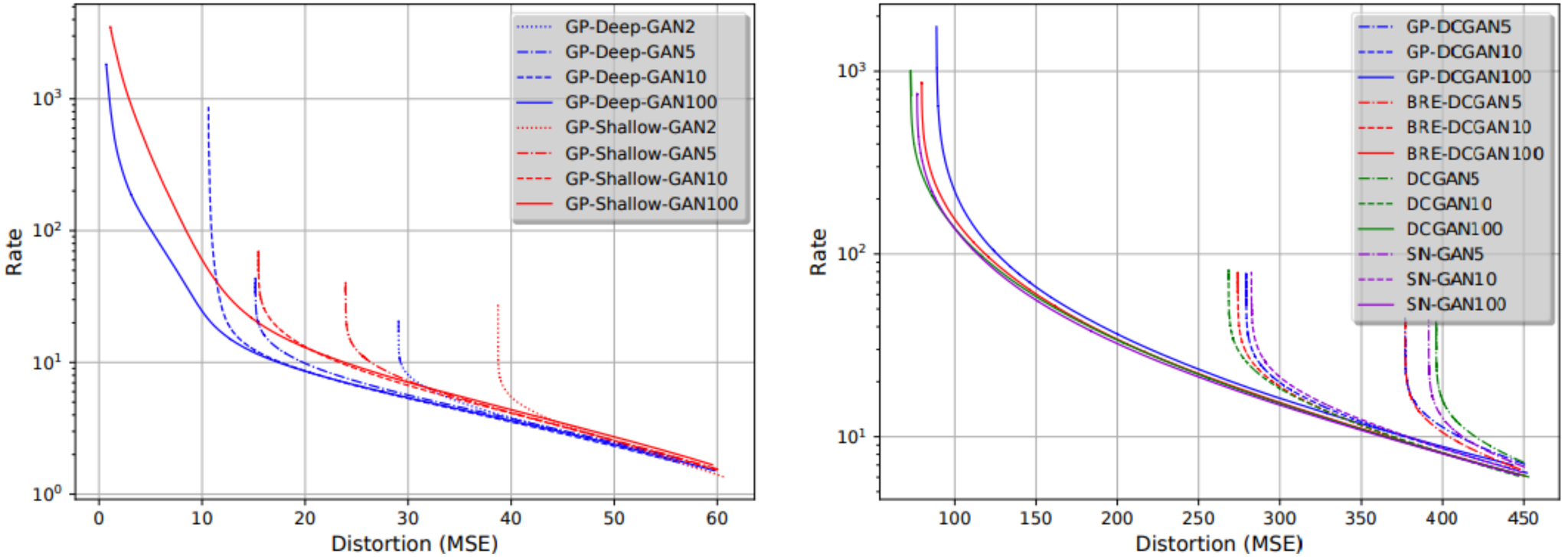


Figure 2. **(a)** Prior samples, or equivalently, reconstructions with $\beta = 0$. **(b)** High-rate reconstructions ($\beta = 5$). **(c)** Variational rate distortion curves for each of these three models.

- $p_1(x), p_2(x)$ *and* $p_3(x)$ are 2-d gen models with 1-d latent code.
- Conditional likelihoods are the grey curves and coloured dots are prior samples
- $p_1(x)$ *and* $p_2(x)$ have the same prior but different decoder

Fig Credit: Huang et. al 2020

Rate Distortion Curves: GANs

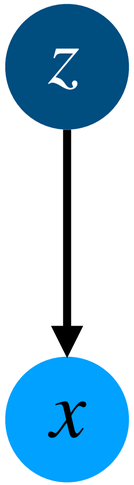


- Increasing the code size has the effect of extending the curve leftward. High-rate regime is effectively measuring reconstruction ability and a larger code size helps with this.
- Increasing the depth pushes the curves down and to the left. Capacity helps the network make better use of the information in code space

Fig Credit: Huang et. al 2020

What about Disentanglement?

Principle. *A disentangled representation should separate the distinct, informative factors of variations in the data (Bengio et. Al 2013)*



2-step Generative Process: First, a multivariate latent random variable z is sampled from a distribution $P(z)$. z corresponds to semantically meaningful factors of variation of the observations. Then, in a second step, the observation x is sampled from the conditional distribution $P(x \mid z)$.

What about Disentanglement?

Theorem 1. *For $d > 1$, let $\mathbf{z} \sim P$ denote any distribution which admits a density $p(\mathbf{z}) = \prod_{i=1}^{d} p(\mathrm{z}_i)$. Then, there exists an infinite family of bijective functions $f : \mathrm{supp}(\mathbf{z}) \to \mathrm{supp}(\mathbf{z})$ such that $\frac{\partial f_i(\boldsymbol{u})}{\partial u_j} \neq 0$ almost everywhere for all i and j (i.e., $\mathbf{z}$ and $f(\mathbf{z})$ are completely entangled) and $P(\mathbf{z} \leq \boldsymbol{u}) = P(f(\mathbf{z}) \leq \boldsymbol{u})$ for all $\boldsymbol{u} \in \mathrm{supp}(\mathbf{z})$ (i.e., they have the same marginal distribution).*

Unsupervised Disentanglement is impossible (Locatello et. Al 2019)**!**

Strong statements but there are caveats.

Interpreting the Impossibility result

Theorem 1. *For $d > 1$, let $\mathbf{z} \sim P$ denote any distribution which admits a density $p(\mathbf{z}) = \prod_{i=1}^{d} p(\mathrm{z}_i)$. Then, there exists an infinite family of bijective functions $f : \mathrm{supp}(\mathbf{z}) \to \mathrm{supp}(\mathbf{z})$ such that $\frac{\partial f_i(\boldsymbol{u})}{\partial u_j} \neq 0$ almost everywhere for all i and j (i.e., $\mathbf{z}$ and $f(\mathbf{z})$ are completely entangled) and $P(\mathbf{z} \leq \boldsymbol{u}) = P(f(\mathbf{z}) \leq \boldsymbol{u})$ for all $\boldsymbol{u} \in \mathrm{supp}(\mathbf{z})$ (i.e., they have the same marginal distribution).*

Suppose you have disentanglement method that gives a representation $r(x)$ that is perfectly disentangled w.r.t to z. Then there exists an equivalent generative model with latents $\hat{z} = f(z)$ which is completely entangled and $p(z) = p(\hat{z})$ almost everywhere and thus the same martial distributions $P(x)$. Therefore, the disentanglement method cannot distinguish between the two generative models.

Implications of the Impossibility result

Unsupervised disentanglement is impossible. But if you have inductive biases structure can be exploited for better disentanglement.

Visual Proof: Which way is the baseball going? From this evidence alone there are multiple paths the baseball could take and each corresponds to a different generative model. The path of the ball cannot be disentangled without inductiv biases —i.e. flow of time.

What if we had a different definition of Disentanglement?

Definition (Informal Higgins et. Al 2018): A vector representation is called a disentangled representation to a particular decomposition of a symmetry group into subgroups, if it decomposes into independent subspaces, where each subspace is affected by the action of a single subgroup and the action of all other subgroups leave the subspace unaffected.

Groups

1. $e \in G$

   Identity

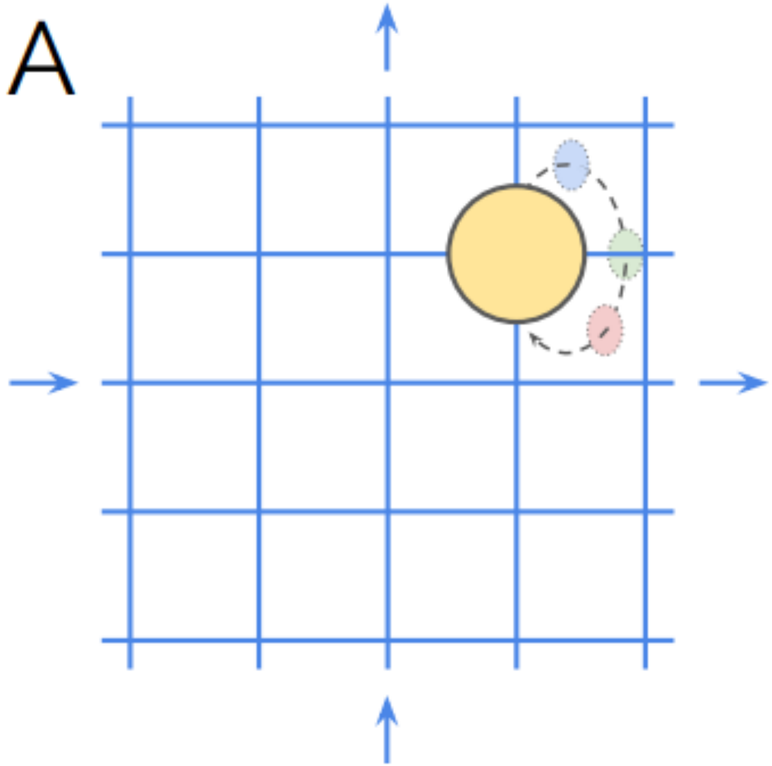
2. $(a \circ b) \circ c = a \circ (b \circ c)$

   Associativity

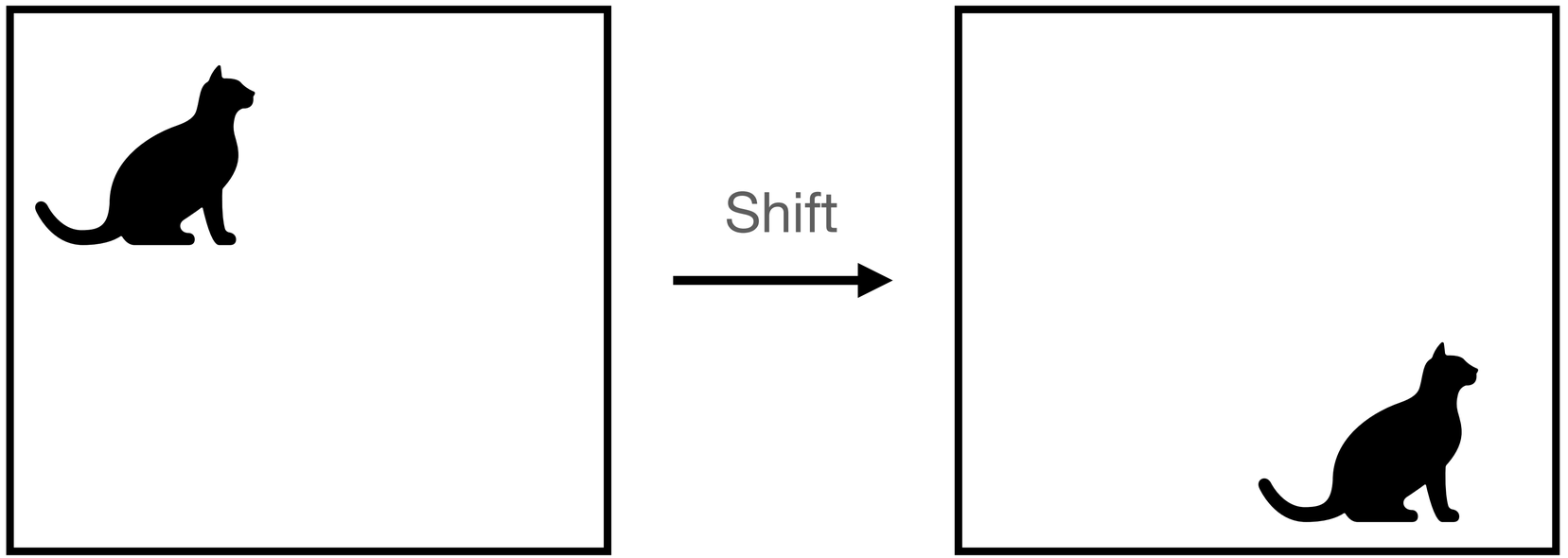
3. $\forall a \in G \quad \exists b \in G$

   $a \circ b = e$

   Unique Inverses

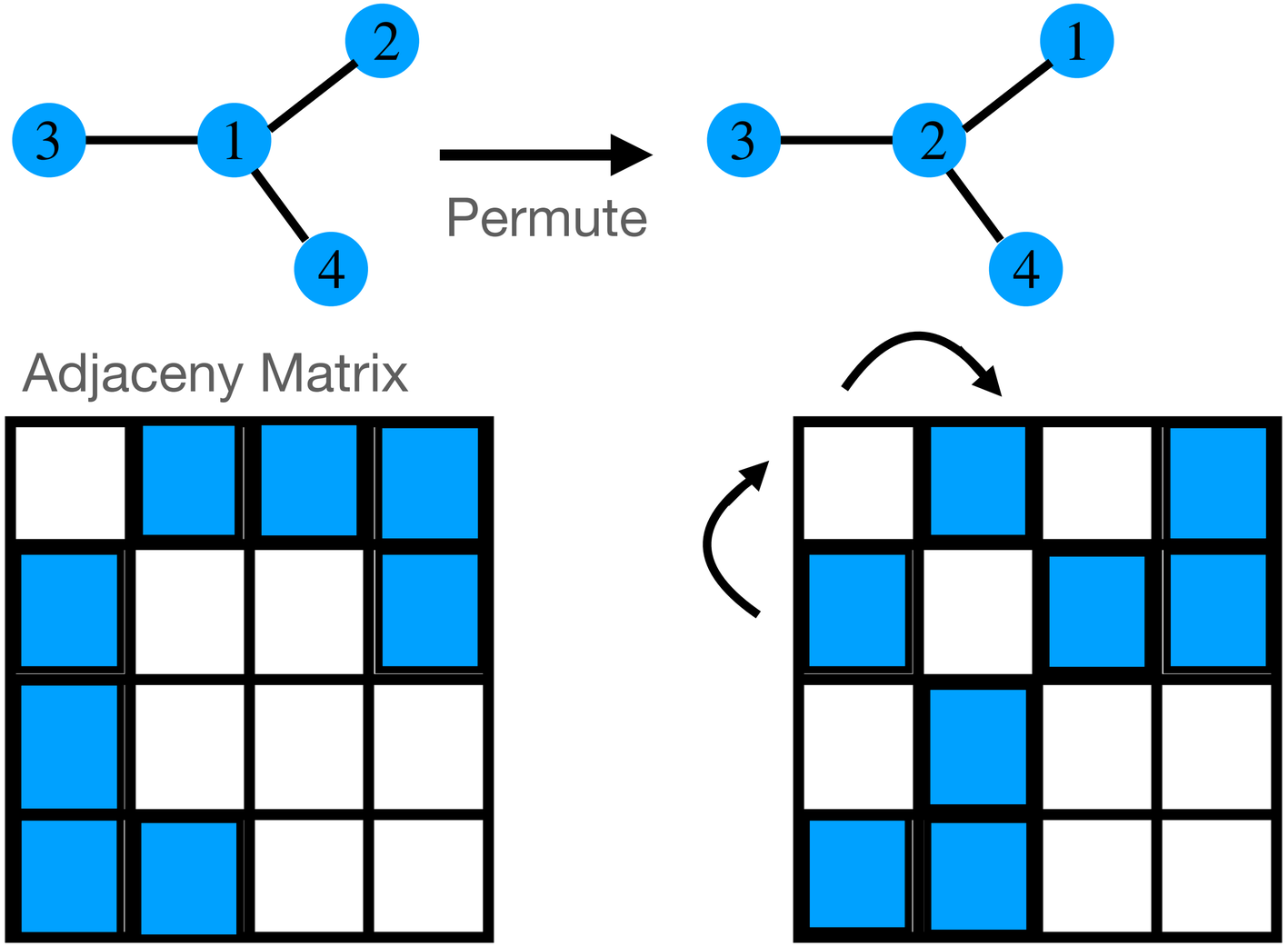


Symmetries in ML



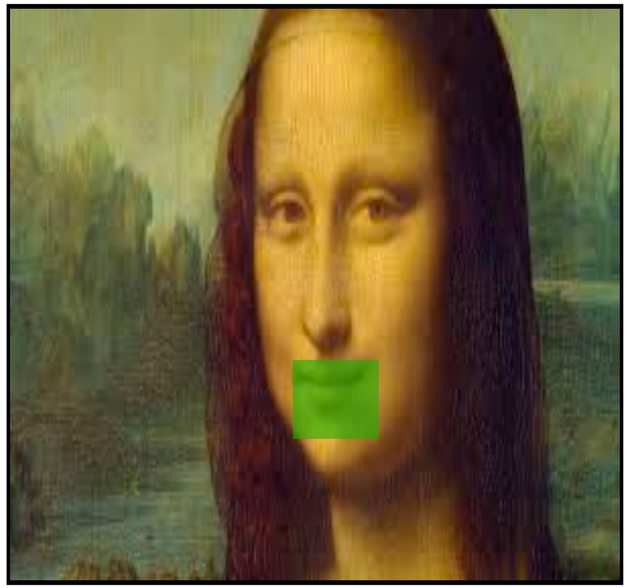
Translation Invariance
in image labels

Symmetries in ML

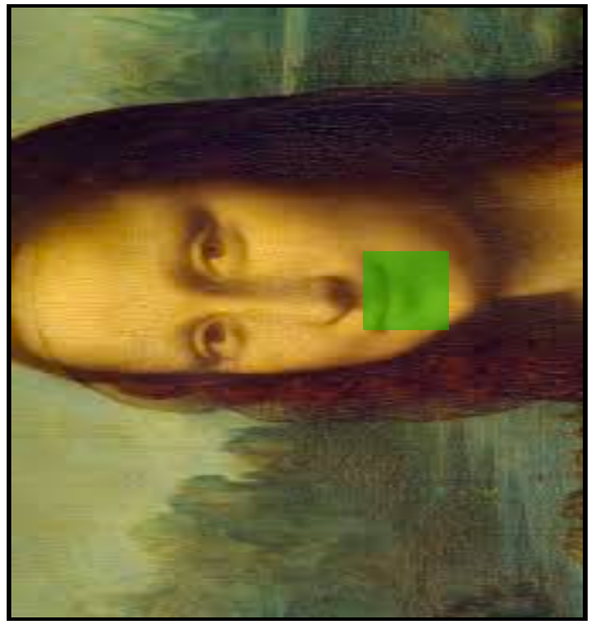


Permutation Invariance in Node Labels in a Graph

Symmetries in ML

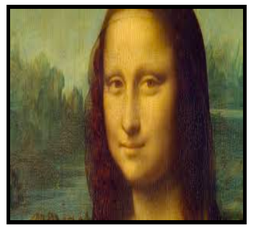


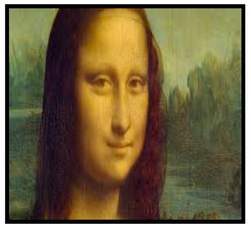
Rotation



Rotation Equivariance
in image features

Example: The dihedral group D_4

e 

s 

$r^4 = e$

r 

rs 

$s^2 = e$

r^2 

r^2s 

$srs = r^{-1}$

r^3 

r^3s 

Basic facts on Representations

1. Two representation R and R' are said to be equivalent if

$$R'(g) = UR(g)U^{\dagger} \text{ for some Unitary Matrix } U$$

2. A representation R is said to be (completely) reducible if

$$R(g) = U \left(\begin{array}{c|c} R_1(g) & \\ \hline & R_2(g) \end{array} \right) U^{\dagger}$$

Complete Reducibility

Theorem. Let R be a representation of a compact group G on a vector space V. If R fixes the subspace W, then it also fixes $W^{\perp}$.

$$R(g) = U \left(\begin{array}{c|c} R_1(g) & B(g) \\ \hline & R_2(g) \end{array} \right) U^{\dagger} \qquad \Longrightarrow \qquad B(g) = 0$$

Corollary. Any representation of a compact group is reducible into a direct sum of irreducible representations. This is Maschke's Theorem in the group is finite, and Peter-Weyl (part 2) for continuous.

		R_0	R_1	R_2	R_3	R_4
e		(1)	(1)	(1)	(1)	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
r		(1)	(1)	(-1)	(-1)	$\begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}$
r^2		(1)	(1)	(1)	(1)	$\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$
r^3		(1)	(1)	(-1)	(-1)	$\begin{pmatrix} -i & 0 \\ 0 & i \end{pmatrix}$
s		(1)	(-1)	(1)	(-1)	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
rs		(1)	(-1)	(-1)	(1)	$\begin{pmatrix} 0 & i \\ -i & 0 \end{pmatrix}$
r^2s		(1)	(-1)	(1)	(-1)	$\begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}$
r^3s		(1)	(-1)	(-1)	(1)	$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$

Conjecture

Joey's Principle: Generative Models should focus on learning symmetries in data for disentangled representations.

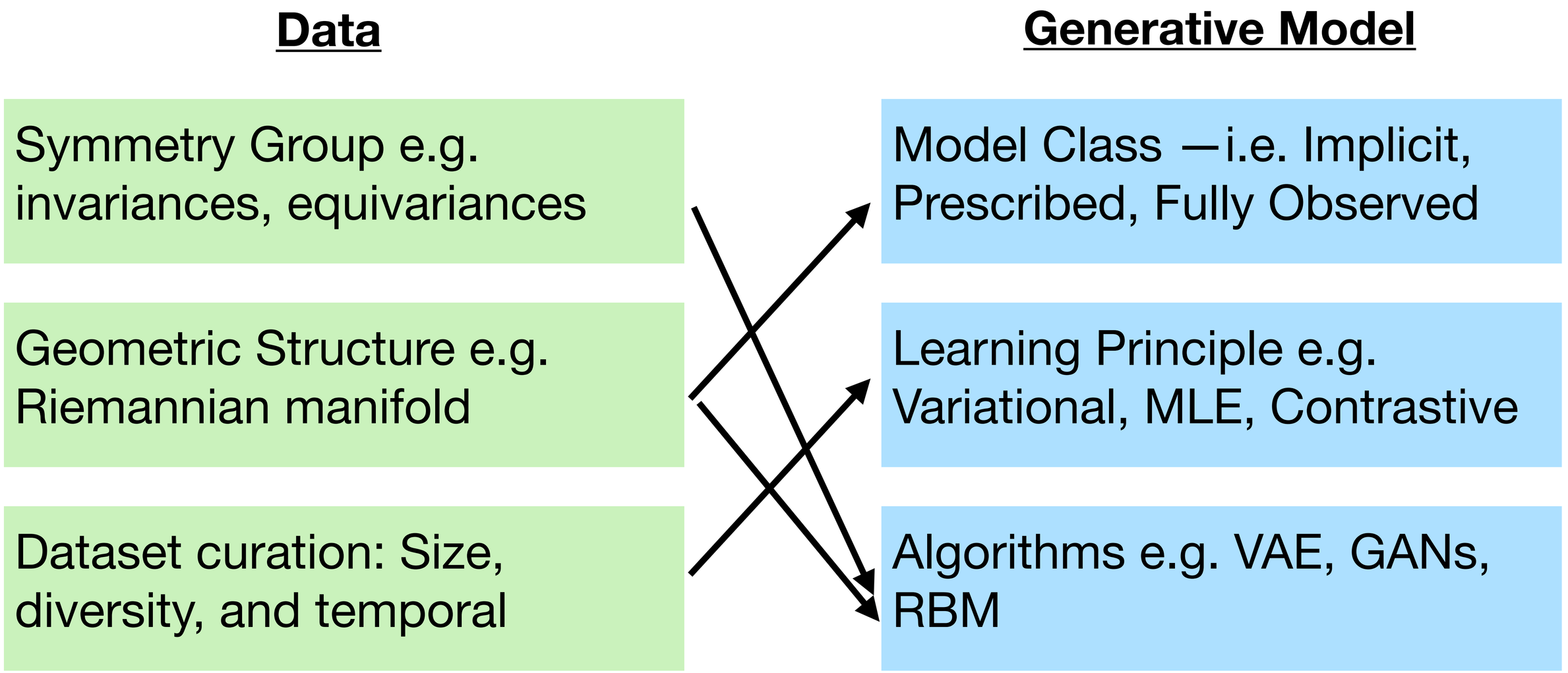
Free Research Idea: We can now solve the Anomaly Detection problem if we use symmetry based disentangled representations. If a data point is not represented as irreducible representations we simply decompose such that it is. We can then build any density model on these representations.

Theorem 1. (Caselles-Dupré et. Al 2019) (Paraphrased): Symmetry based Disentanglement needs interaction with the environment

The False Dichotomy of Generative Models

Data	Generative Model
Symmetry Group e.g. invariances, equivariances	Model Class —i.e. Implicit, Prescribed, Fully Observed
Geometric Structure e.g. Riemannian manifold	Learning Principle e.g. Variational, MLE, Contrastive
Dataset curation: Size, diversity, and temporal	Algorithms e.g. VAE, GANs, RBM

The False Dichotomy of Generative Models



The False Dichotomy of Generative Models

Data

Symmetry Group e.g. invariances, equivariances

Geometric Structure e.g. Riemannian manifold

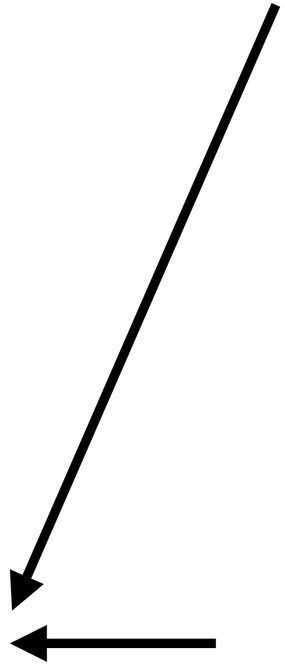
Dataset curation: Size, diversity, and temporal

Generative Model

Model Class —i.e. Implicit, Prescribed, Fully Observed

Learning Principle e.g. Variational, MLE, Contrastive

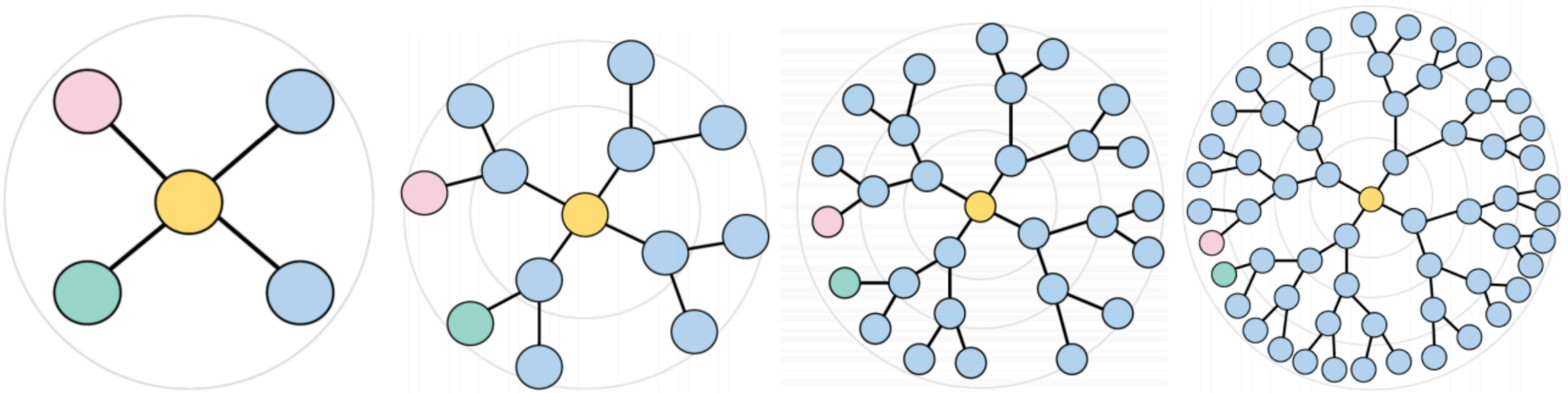
Algorithms e.g. VAE, GANs, RBM



Benefits of modelling data in a Generative Model

Example with Hyperbolic Geometry:

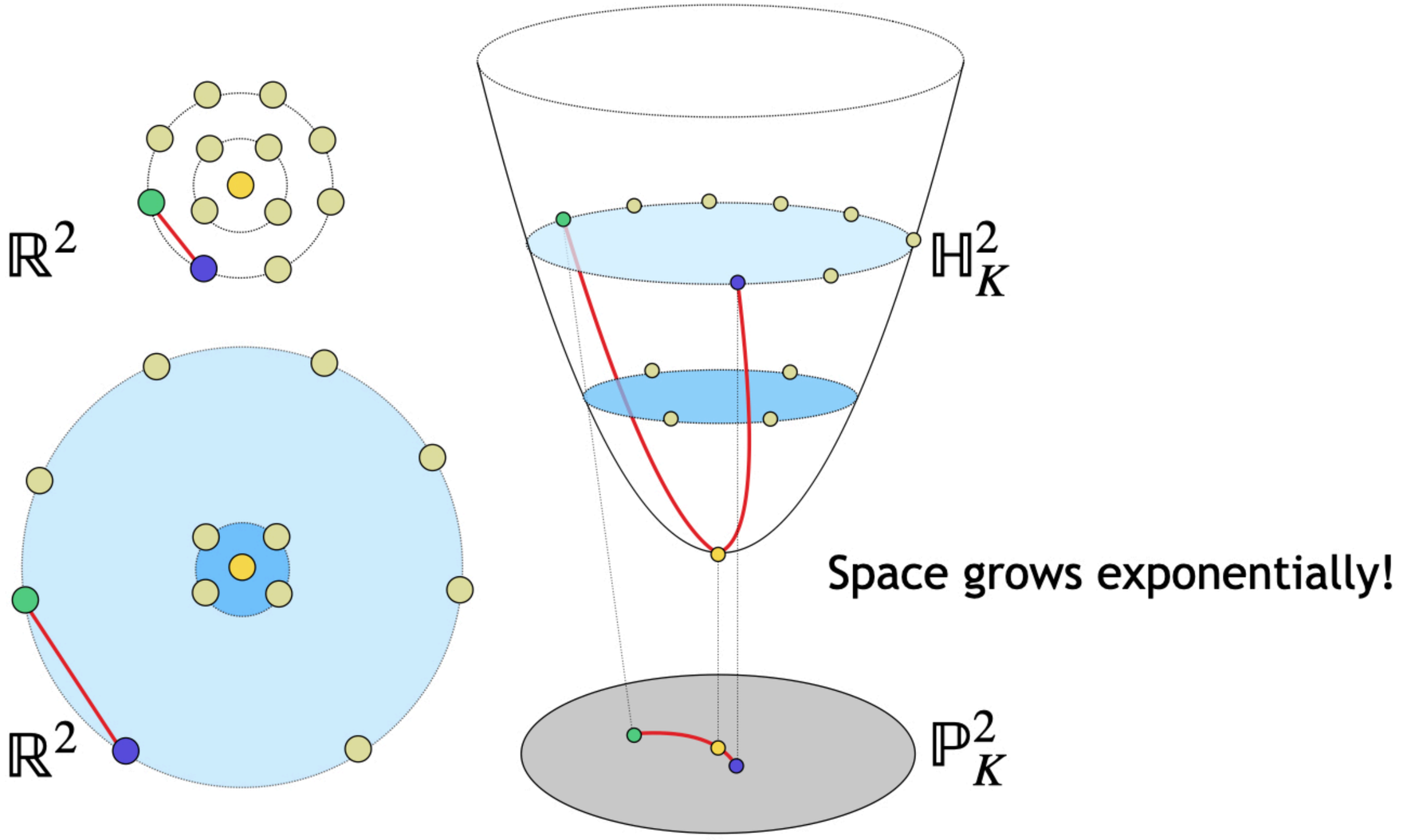
Embedding Hierarchies in Euclidean space



We quickly run out of space! Node Embedding Distance does not respect graph distance!

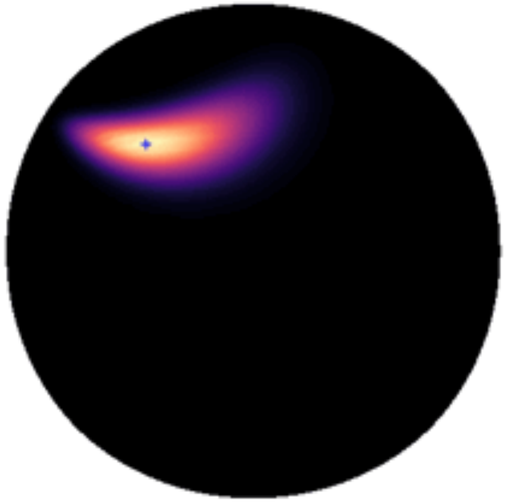
Fig credit: https://openreview.net/pdf?id=BJg73xHtvr

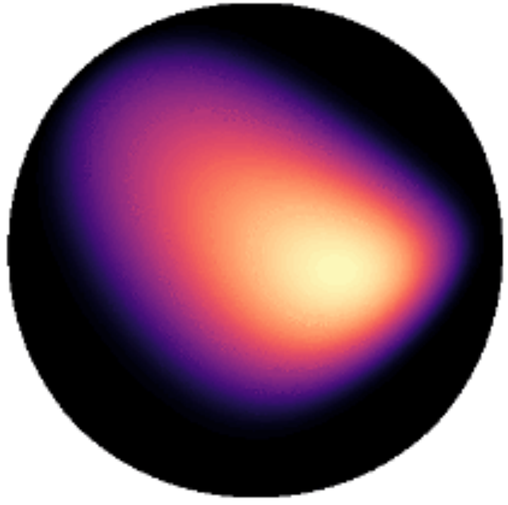
Embedding Hierarchies in Hyperbolic Space



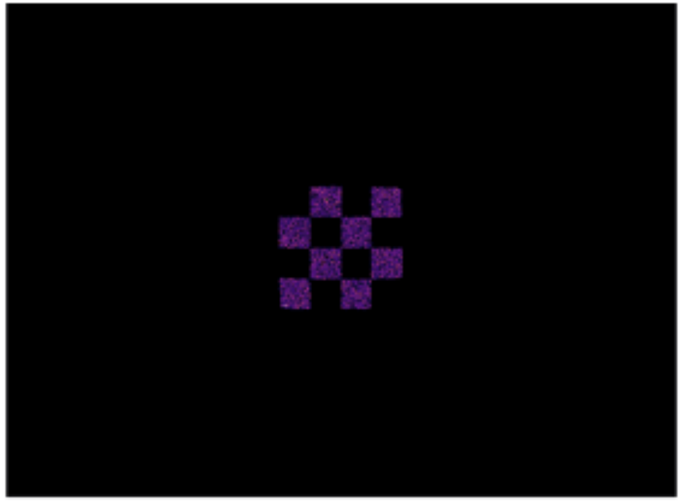
Density Estimation on $\mathbb{H}^n_K$

Wrapped Gaussian





Checkerboard



2D Spiral