NOTAGAN

Normalizing flows for simul**t**aneous m**a**nifold learnin**g** **a**nd density estimatio**n**

Johann Brehmer
New York University

Kyle Cranmer
New York University

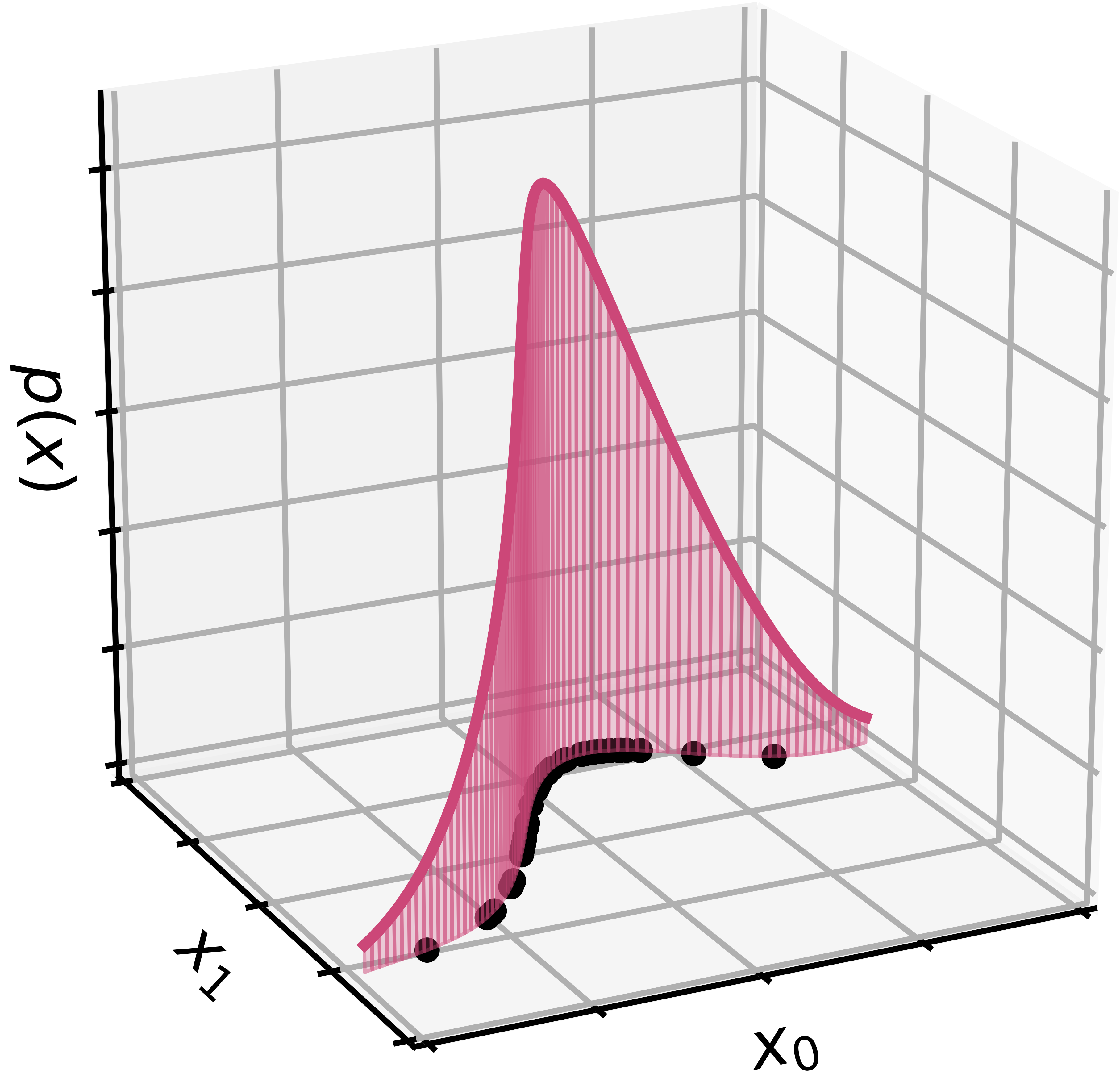
The SCAILFIN Project
scailfin.github.io

Scientific Data and Computing Center

$\mathcal{M}$-flow:

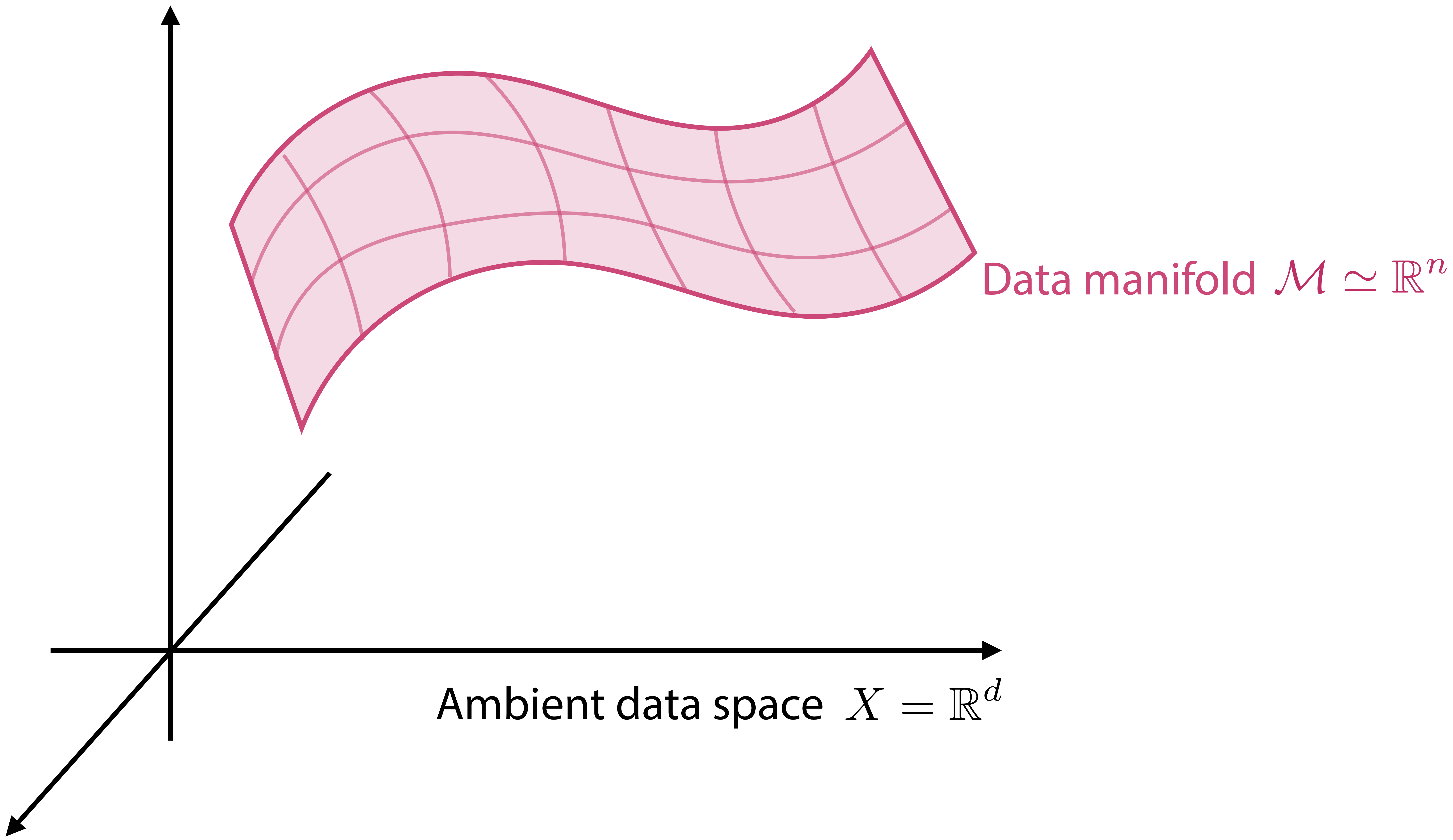
A normalizing flow that

- describes data as a probability density over a lower-dimensional manifold
- learns manifold and density from data
- has a tractable density

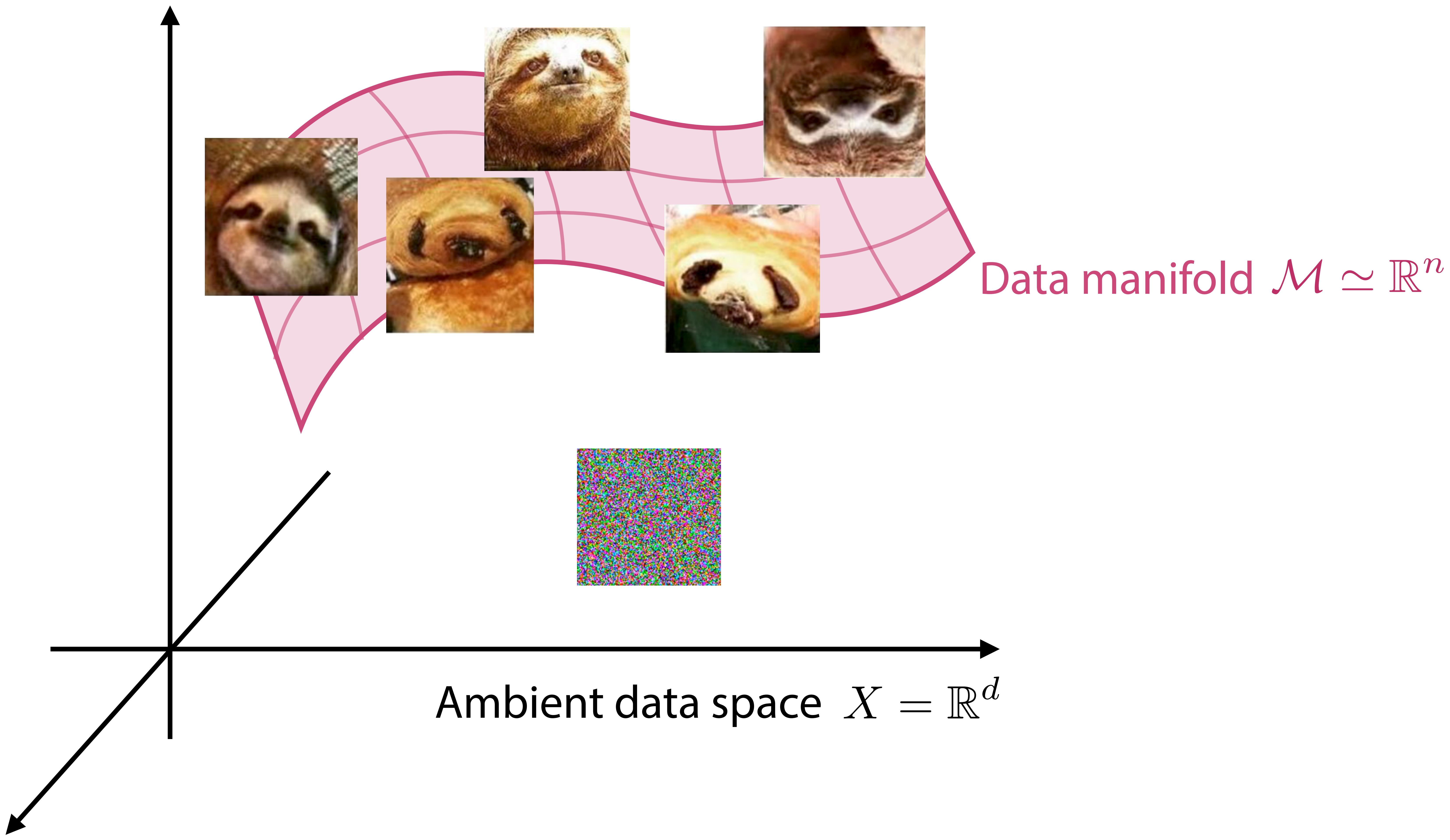


m-flo:

[m-flo / Rhythm Zone]

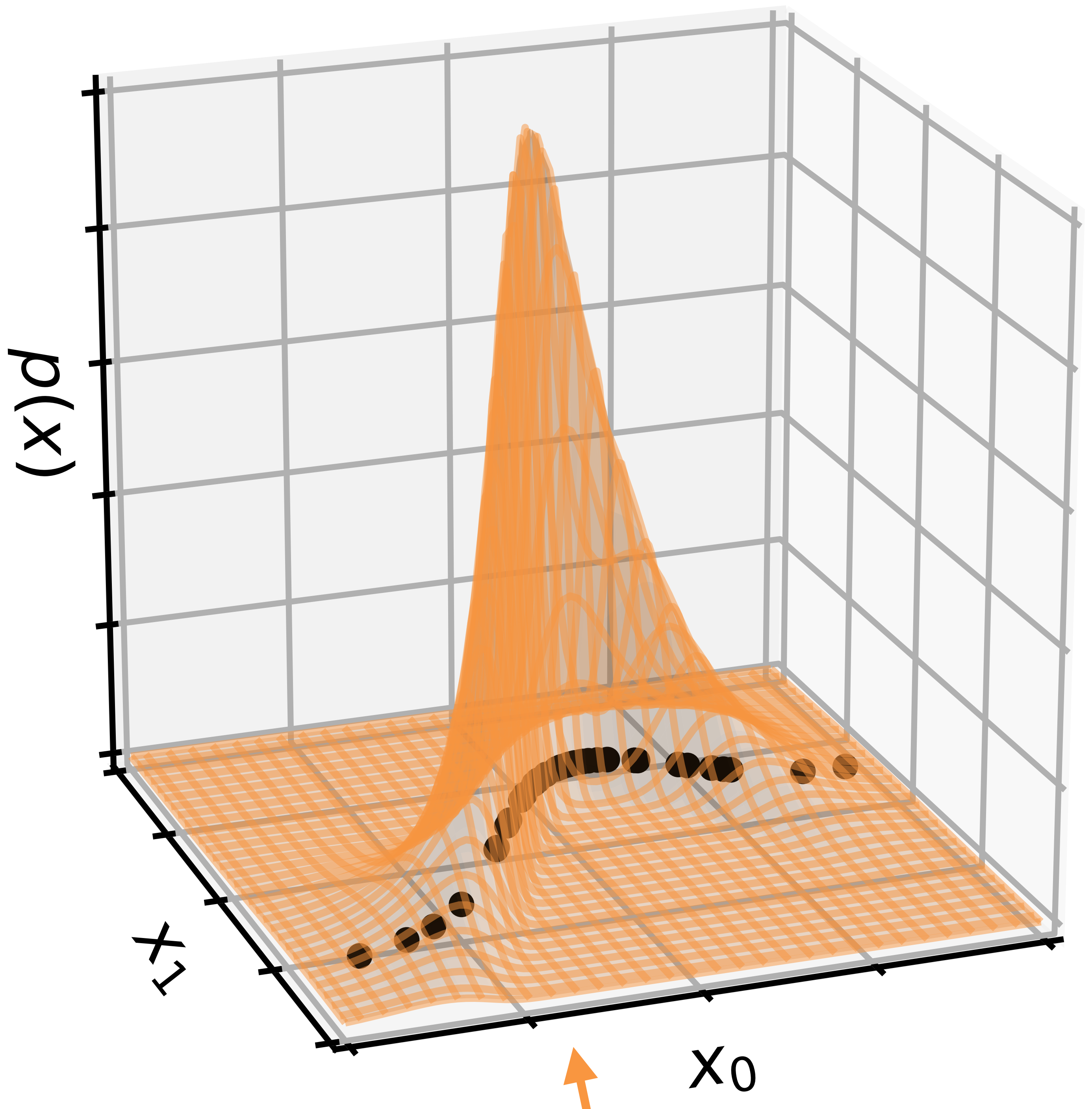


Data manifold $\mathcal{M} \simeq \mathbb{R}^n$
Ambient data space $X = \mathbb{R}^d$



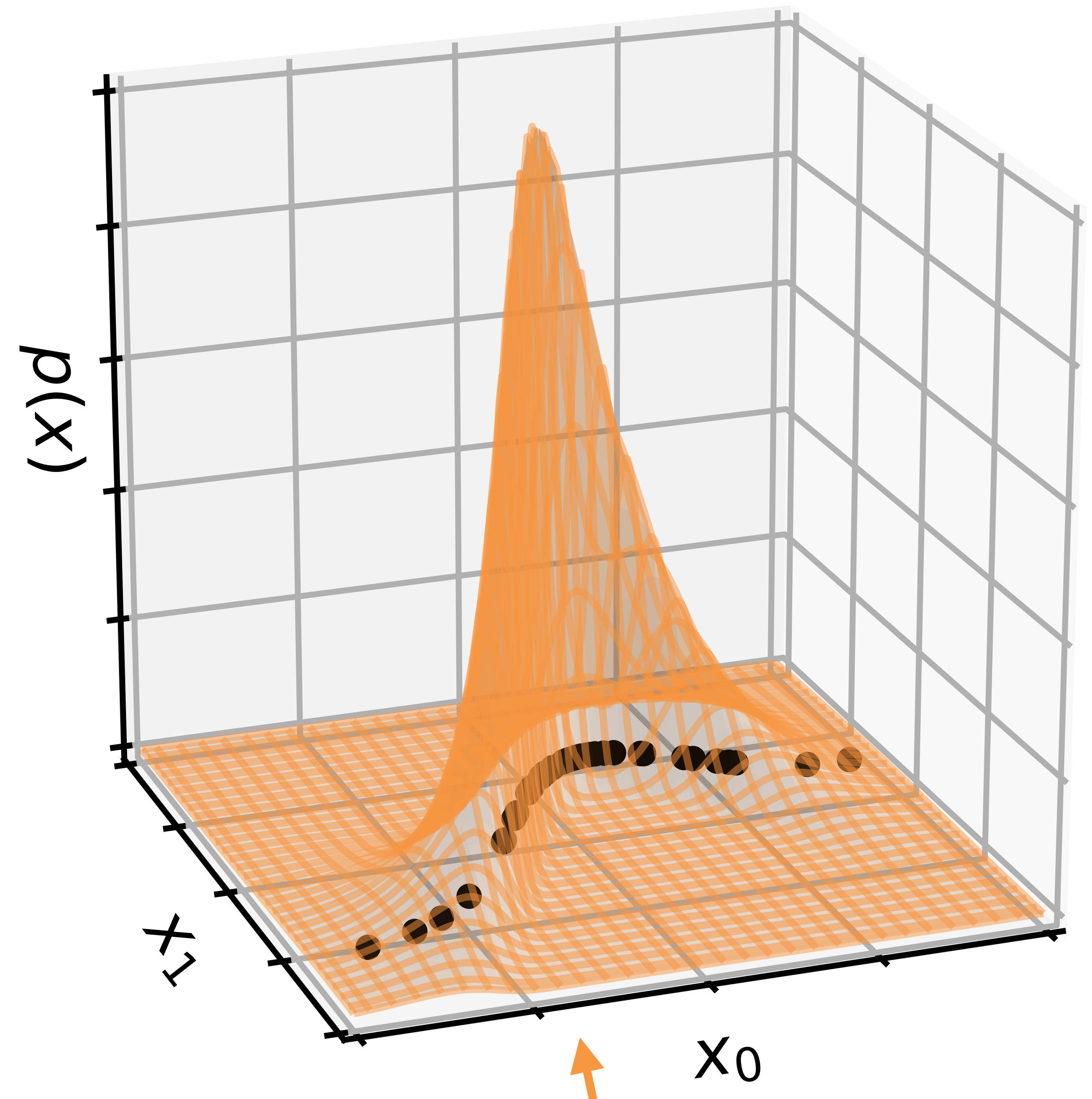
[Original meme by K. Zack]

Standard flows in ambient data space



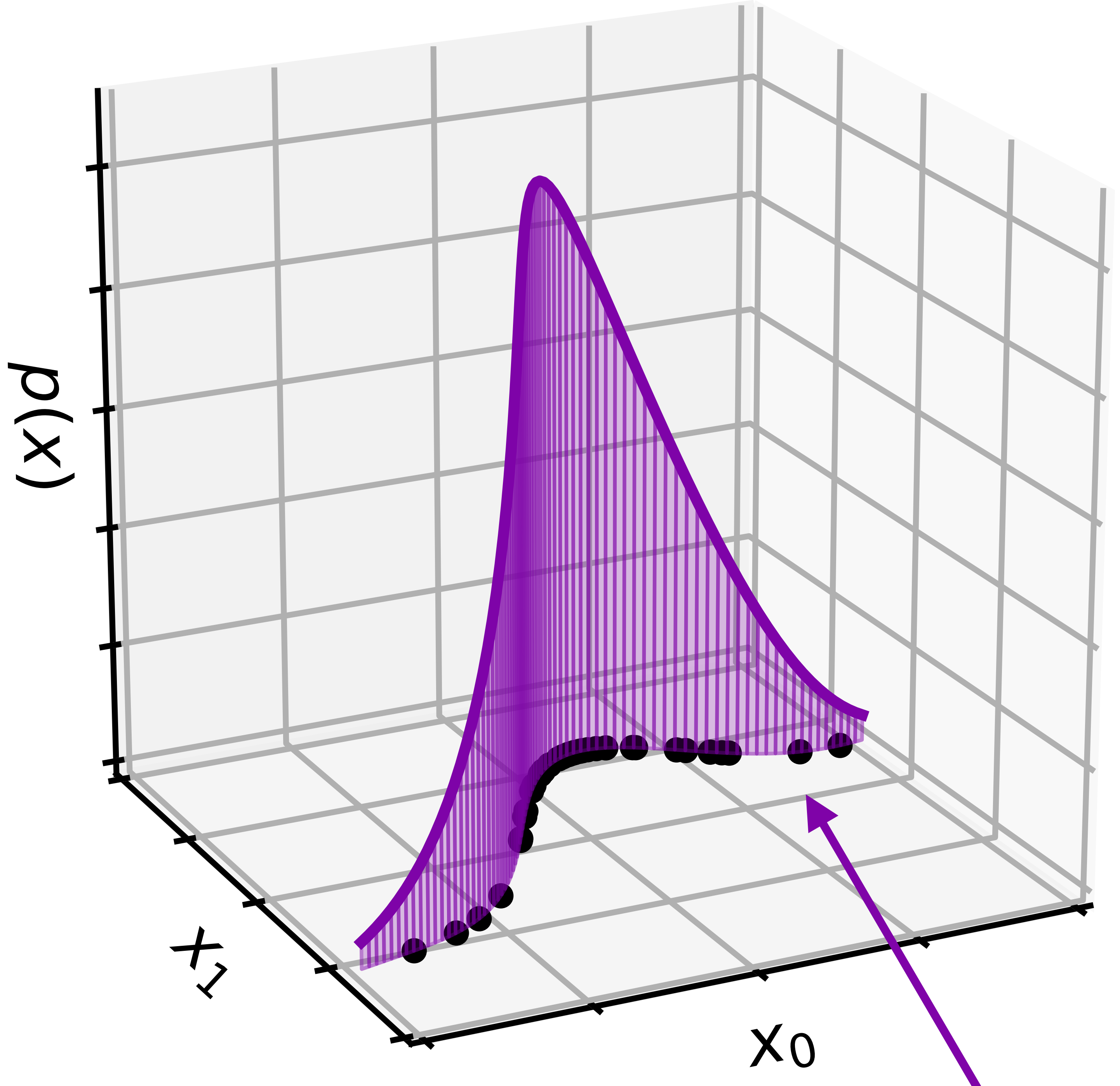
Density has support
off the data manifold

Standard flows in ambient data space



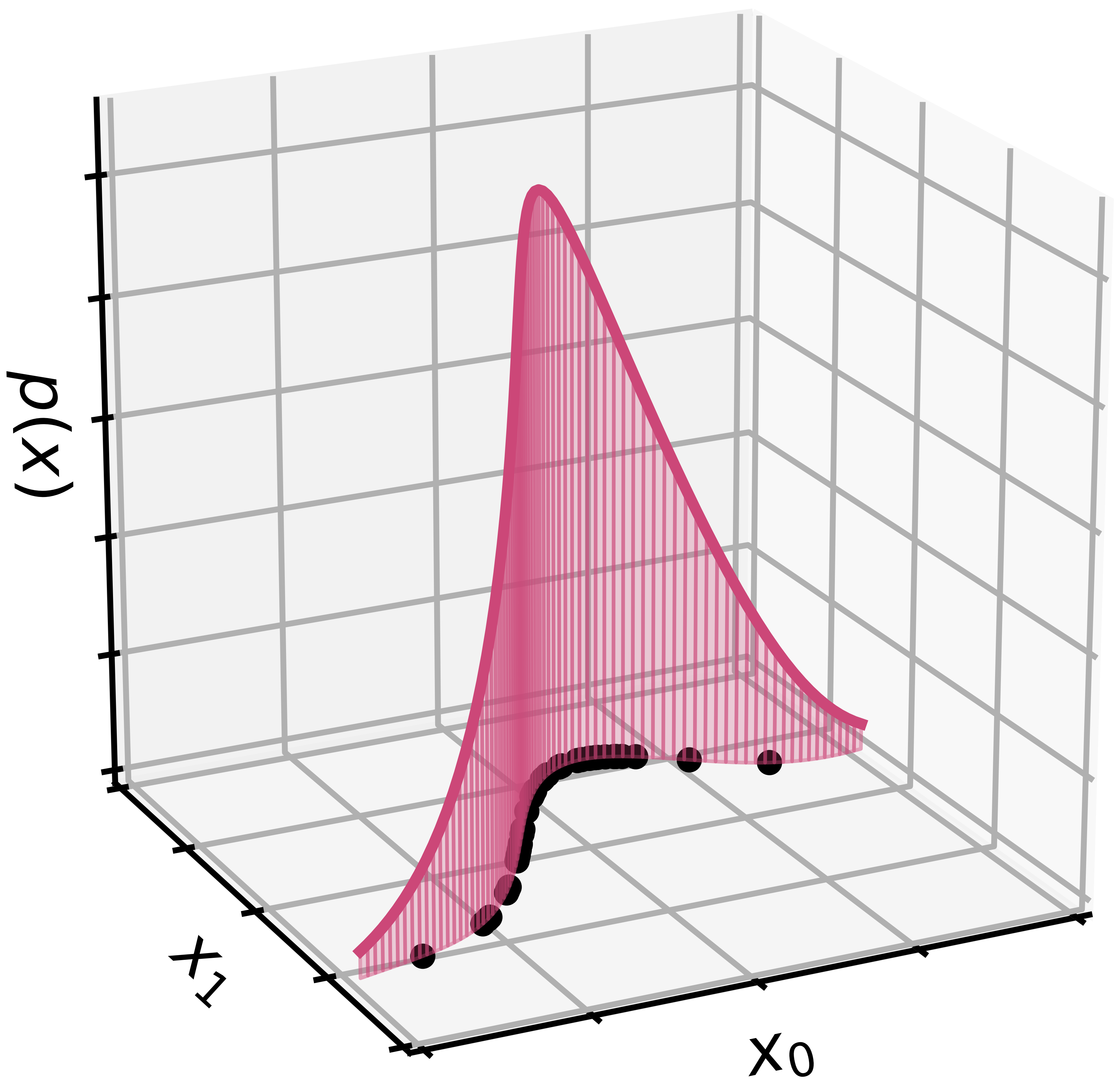
Density has support off the data manifold

Flows on prescribed manifolds [M. Gemici et al 1611.02304]



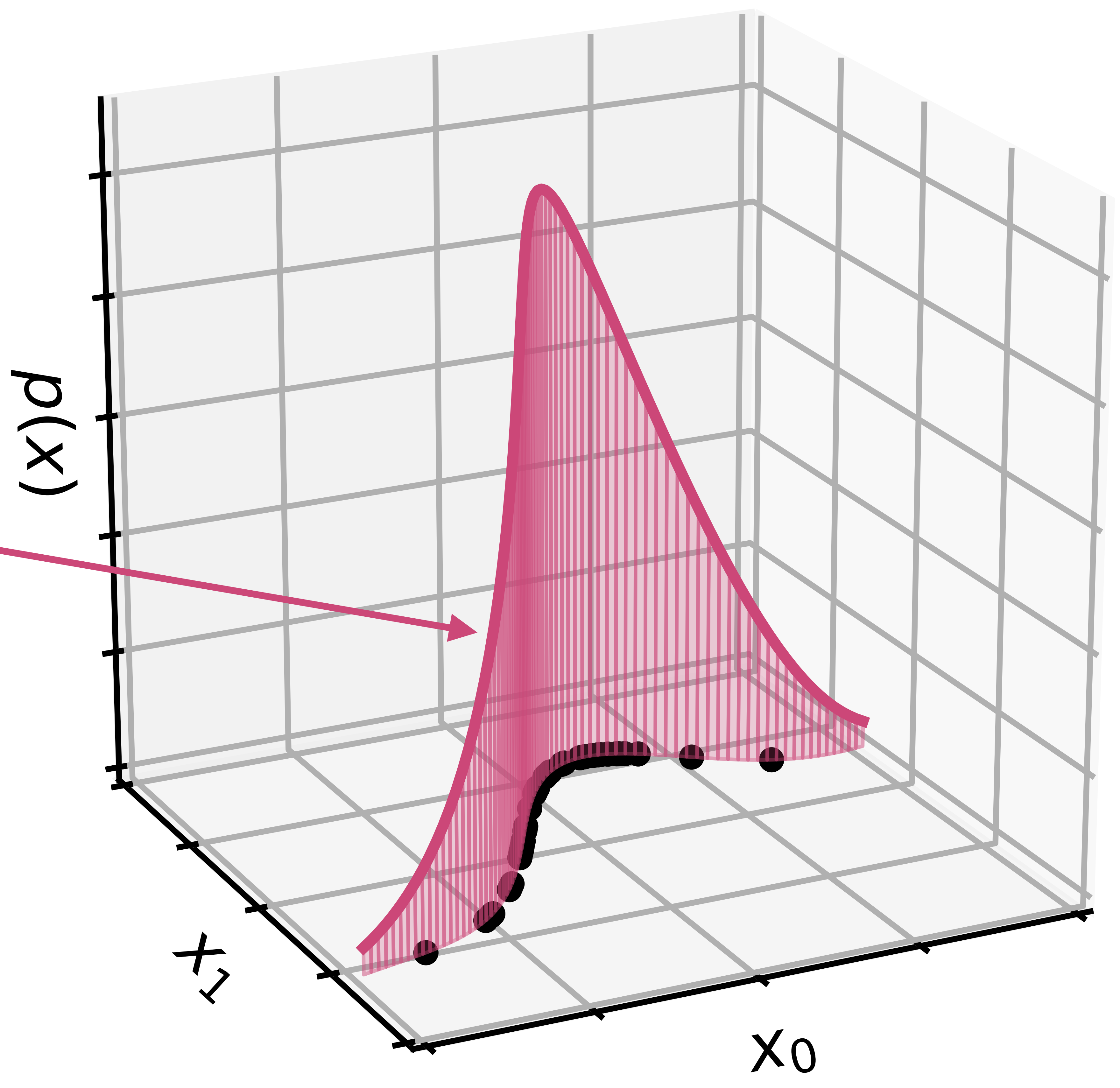
Need to know manifold / chart in advance

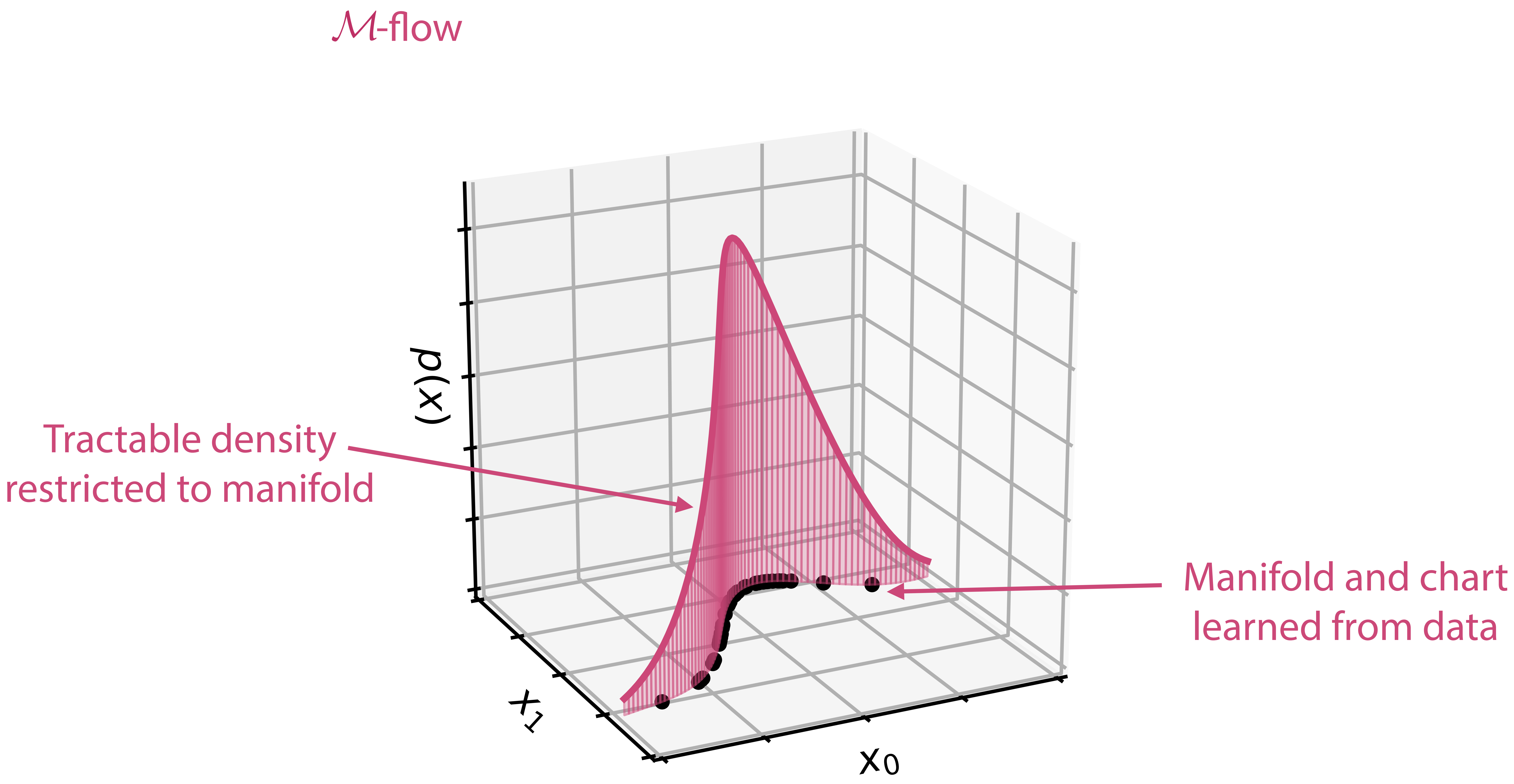
$\mathcal{M}$-flow

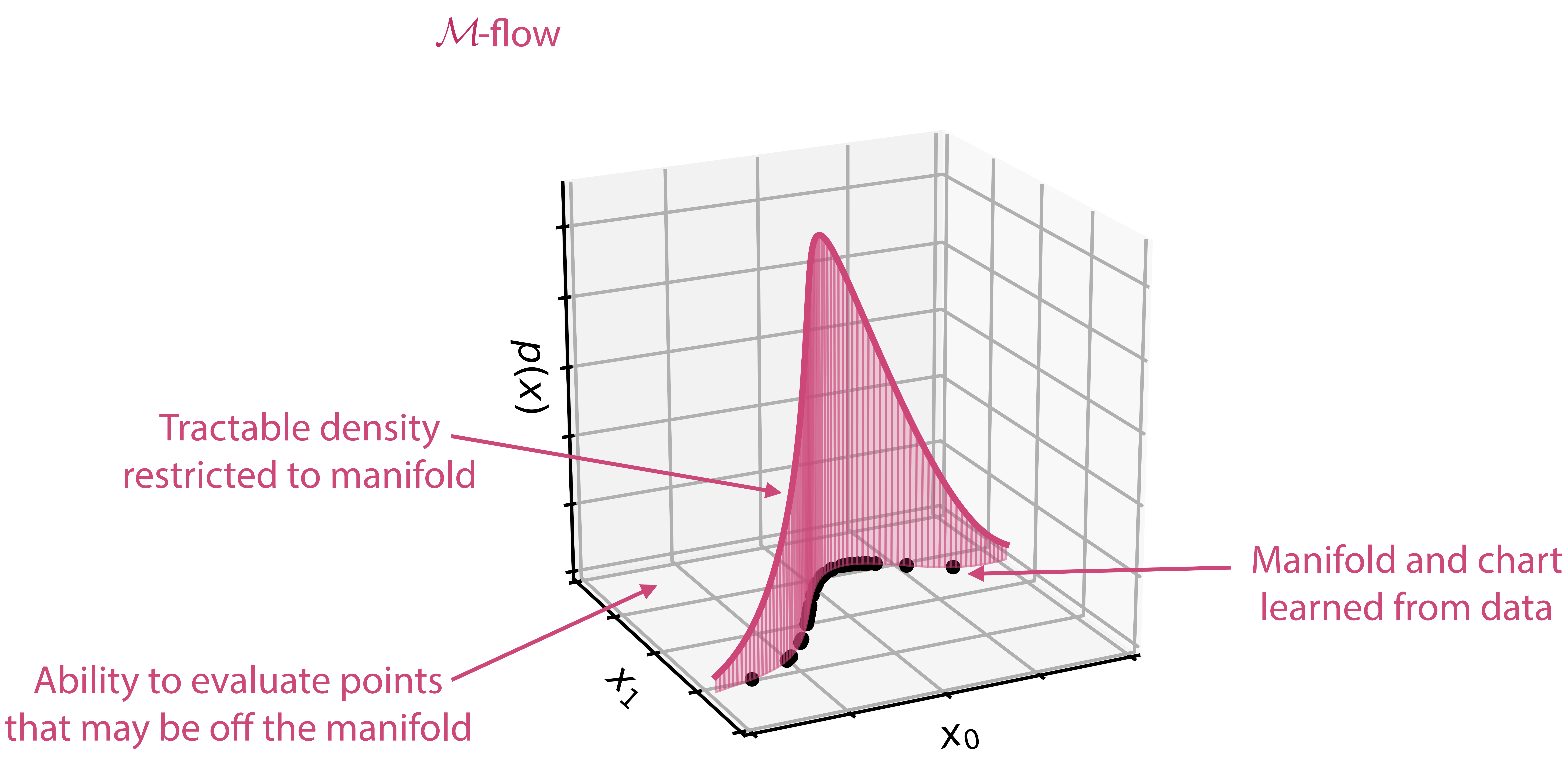


$\mathcal{M}$-flow

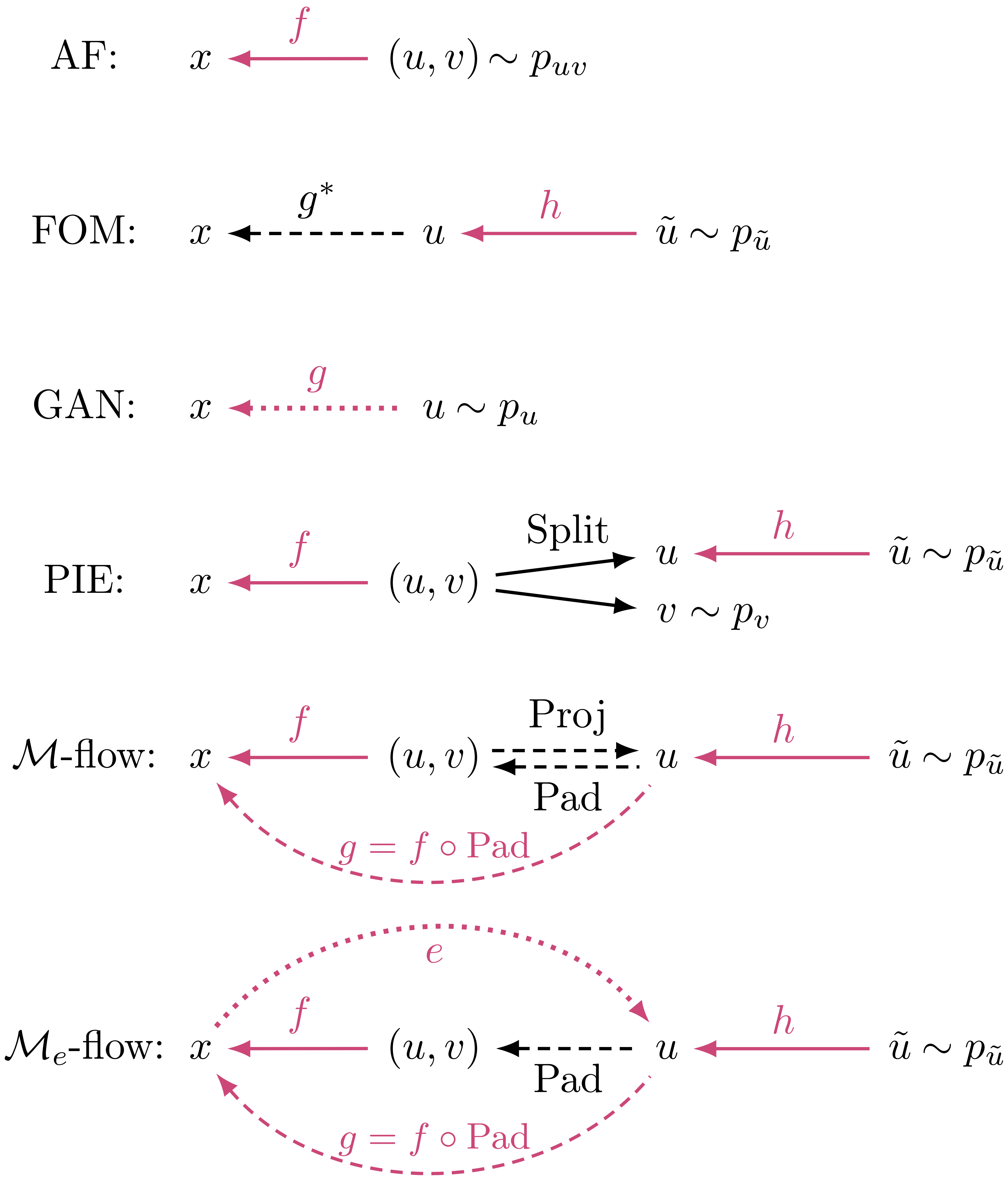
Tractable density restricted to manifold

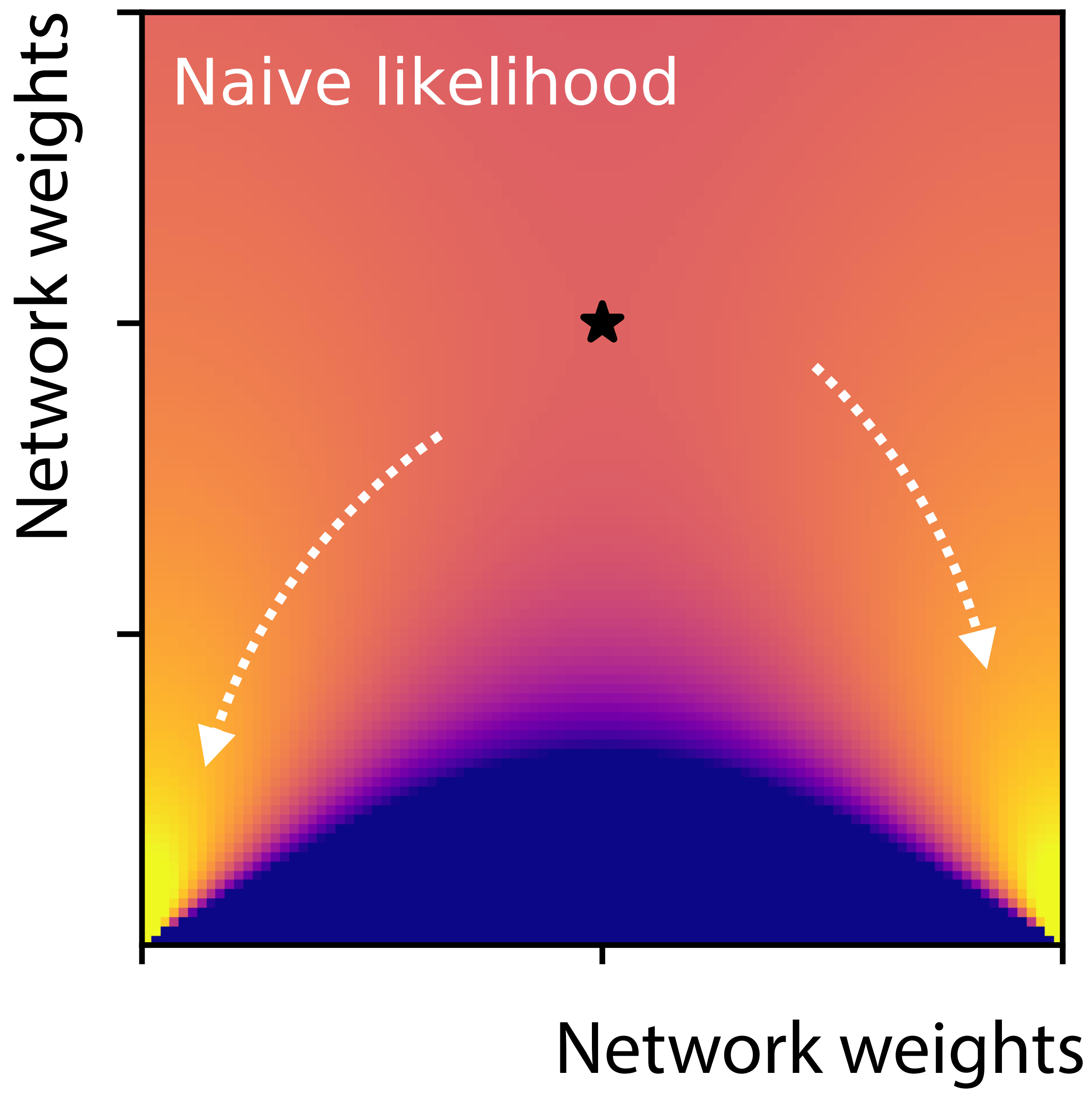


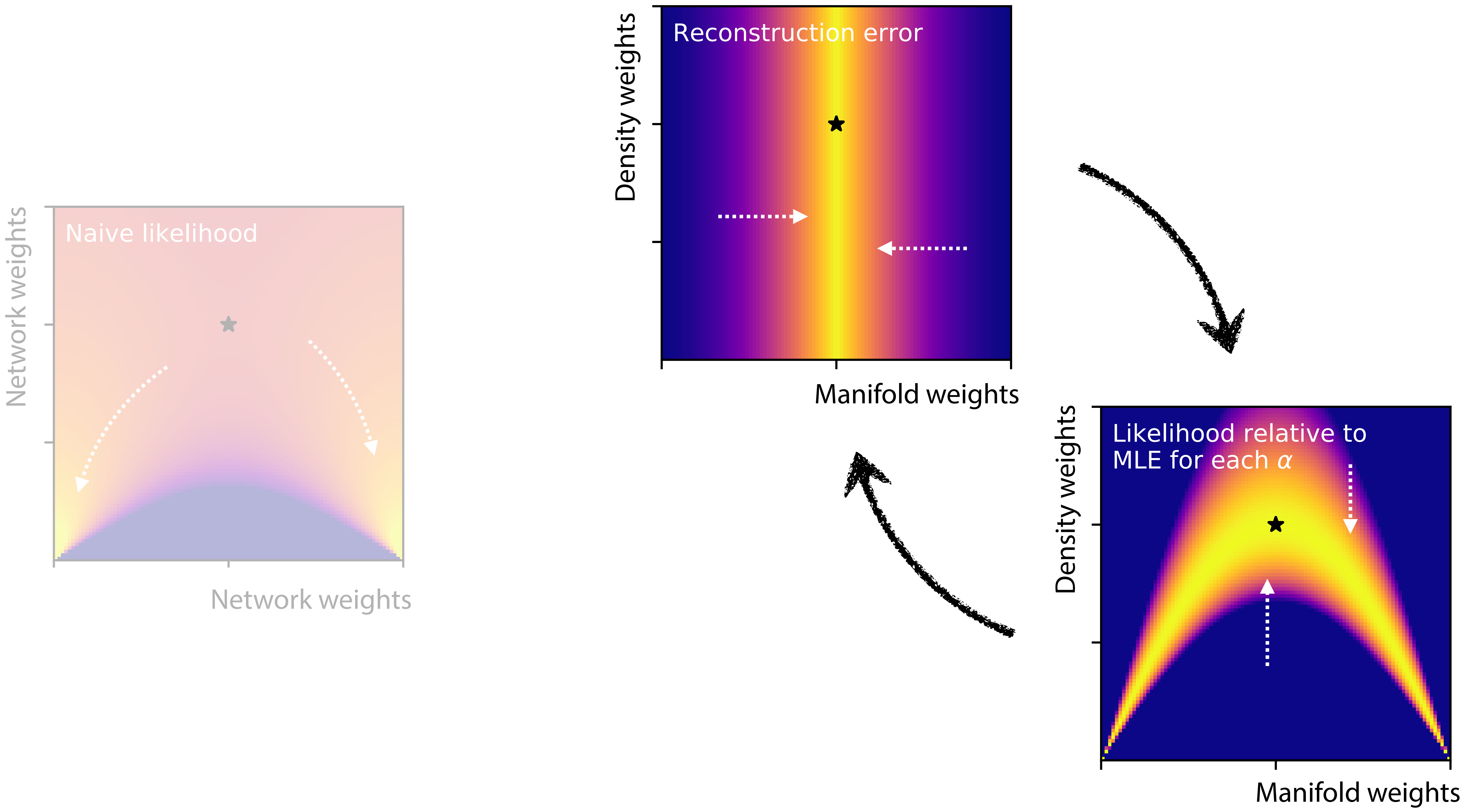

$\mathcal{M}$-flow
$p(x)$
Tractable density
restricted to manifold
Manifold and chart
learned from data
x_1
x_0

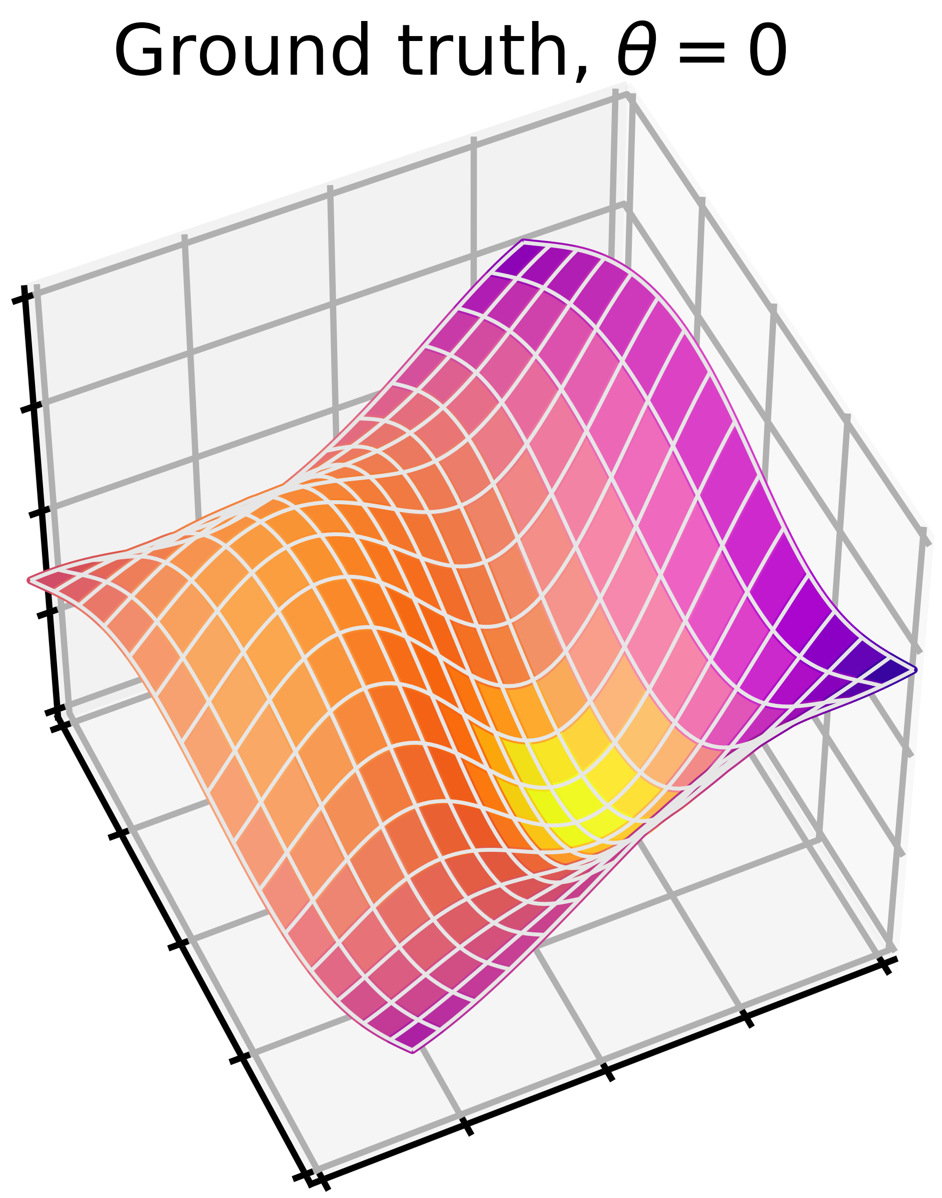


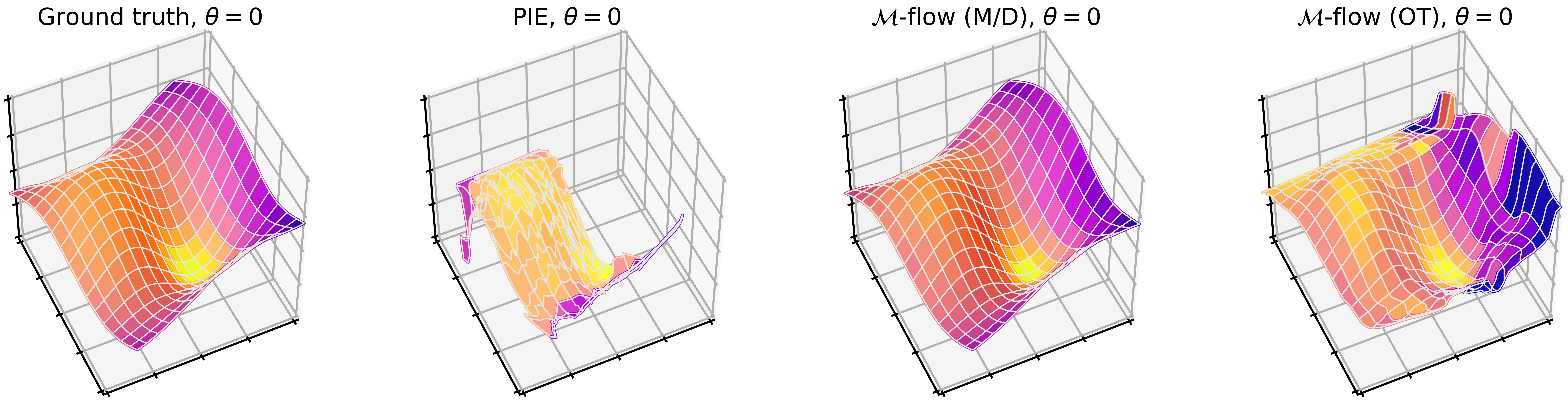
$\mathcal{M}$-flow
Tractable density
restricted to manifold
Manifold and chart
learned from data
Ability to evaluate points
that may be off the manifold
$p(x)$
x_1
x_0

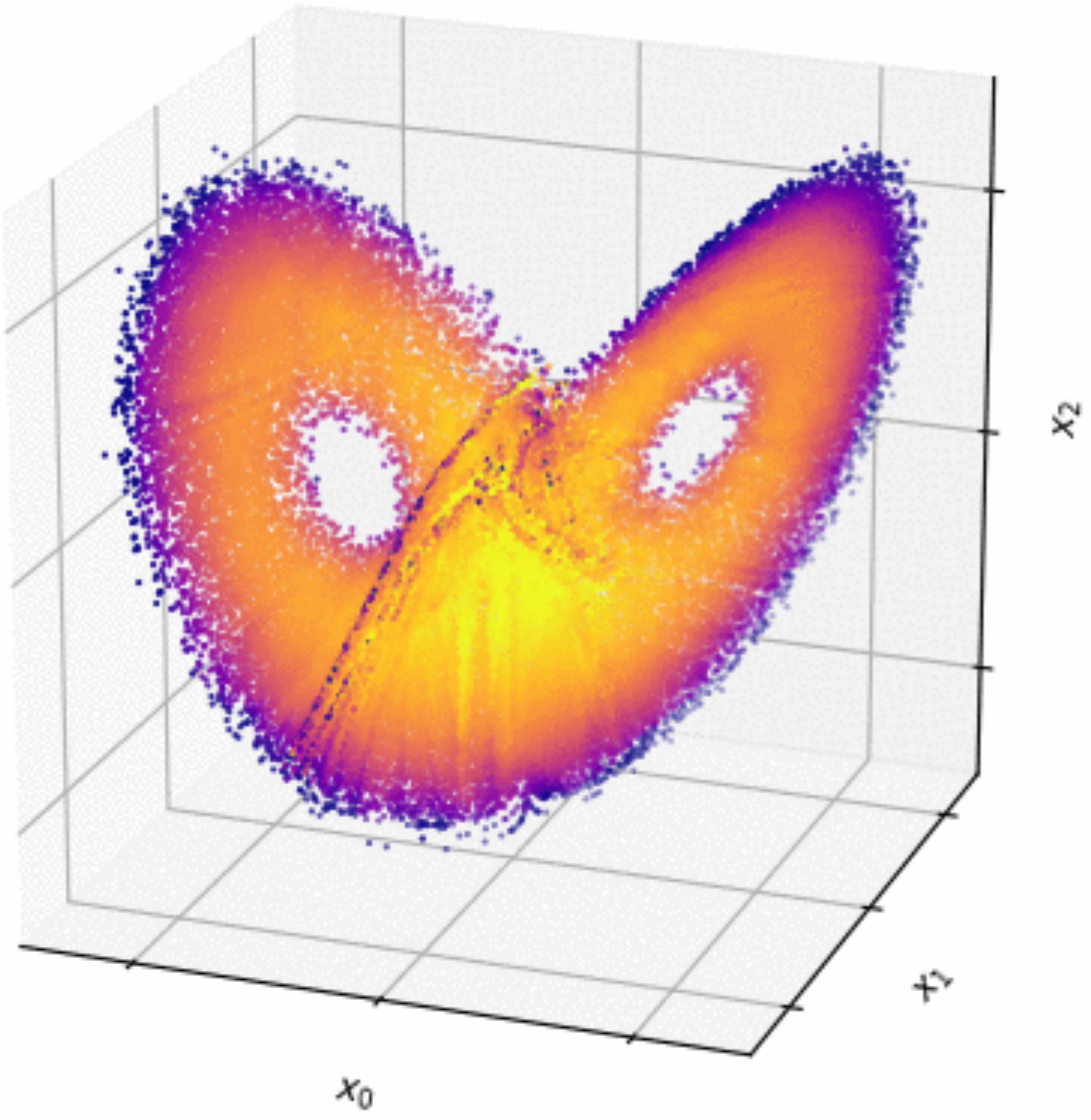

AF:
x
f
$(u, v) \sim p_{uv}$
FOM:
g^*
u
h
$\tilde{u} \sim p_{\tilde{u}}$
GAN:
g
$u \sim p_u$
PIE:
Split
(u, v)
$v \sim p_v$
$\mathcal{M}$-flow:
Proj
Pad
$g = f \circ \mathrm{Pad}$
$\mathcal{M}_e$-flow:
e


Naive likelihood
Network weights
Network weights

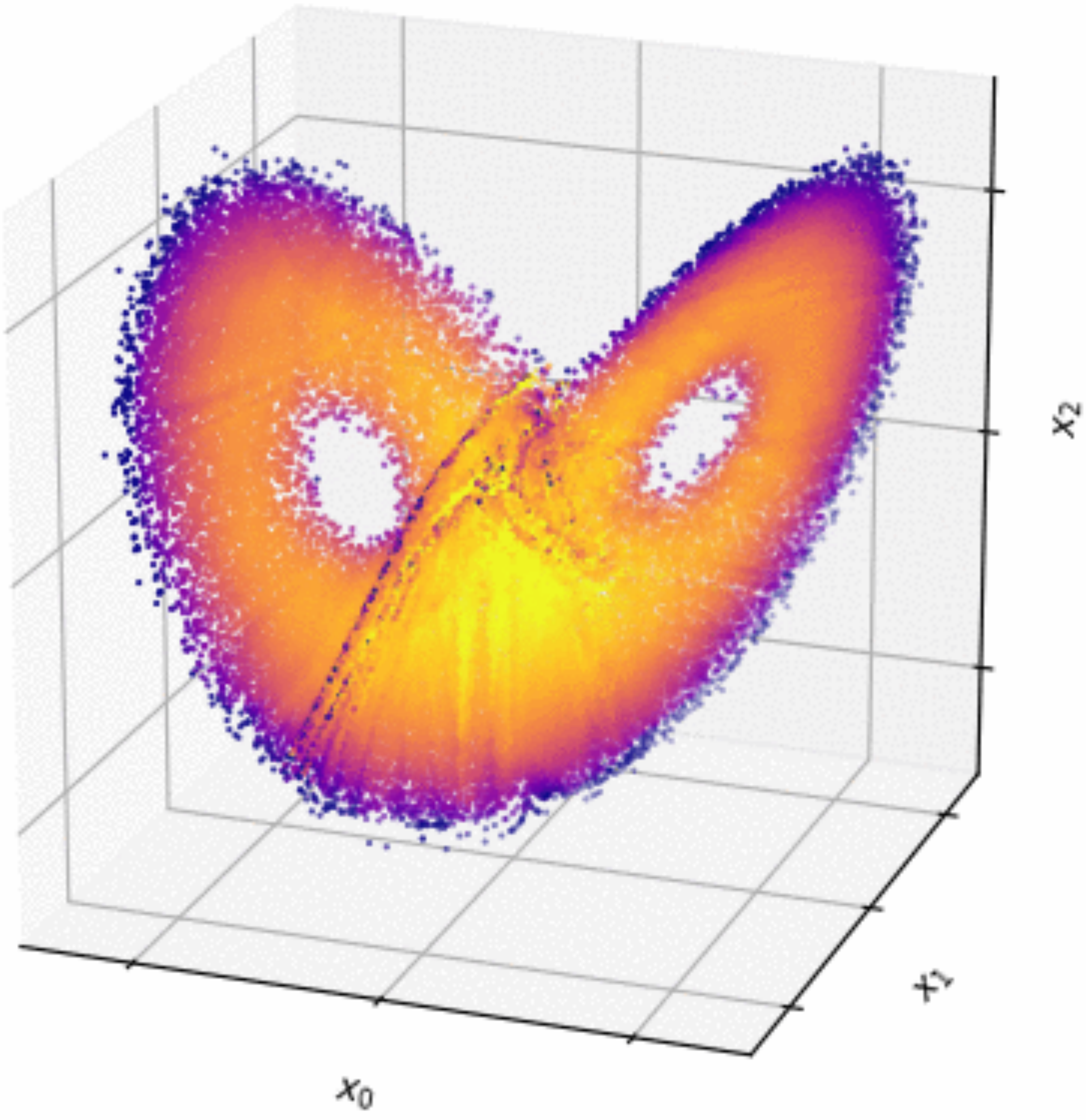

Naive likelihood
Network weights
Network weights
Reconstruction error
Density weights
Manifold weights
Likelihood relative to
MLE for each α
Density weights
Manifold weights


Ground truth, $\theta = 0$

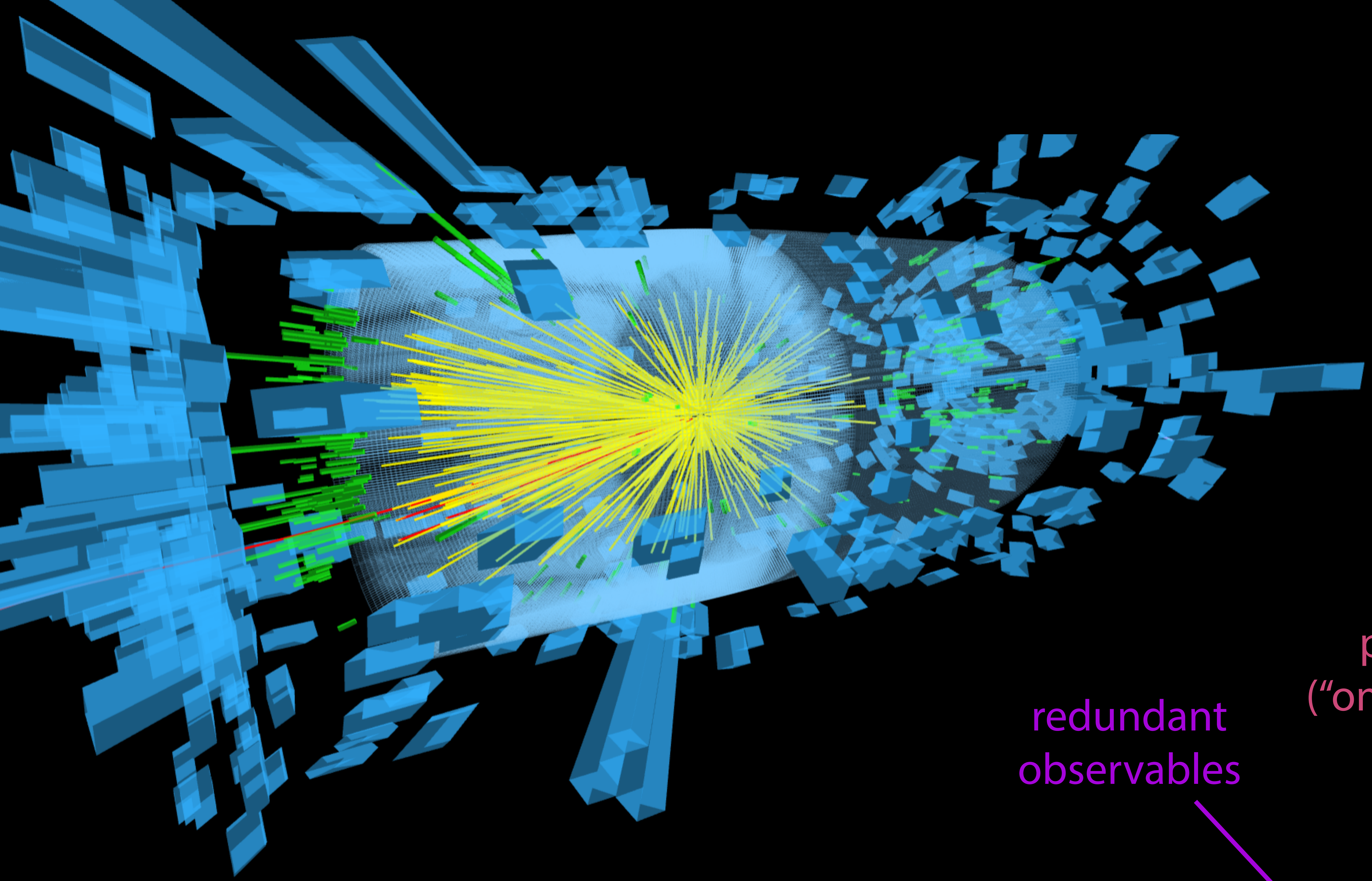

Ground truth, $\theta = 0$
PIE, $\theta = 0$
$\mathcal{M}$-flow (M/D), $\theta = 0$
$\mathcal{M}$-flow (OT), $\theta = 0$



x_0
x_1
x_2



x_2
x_1
x_0



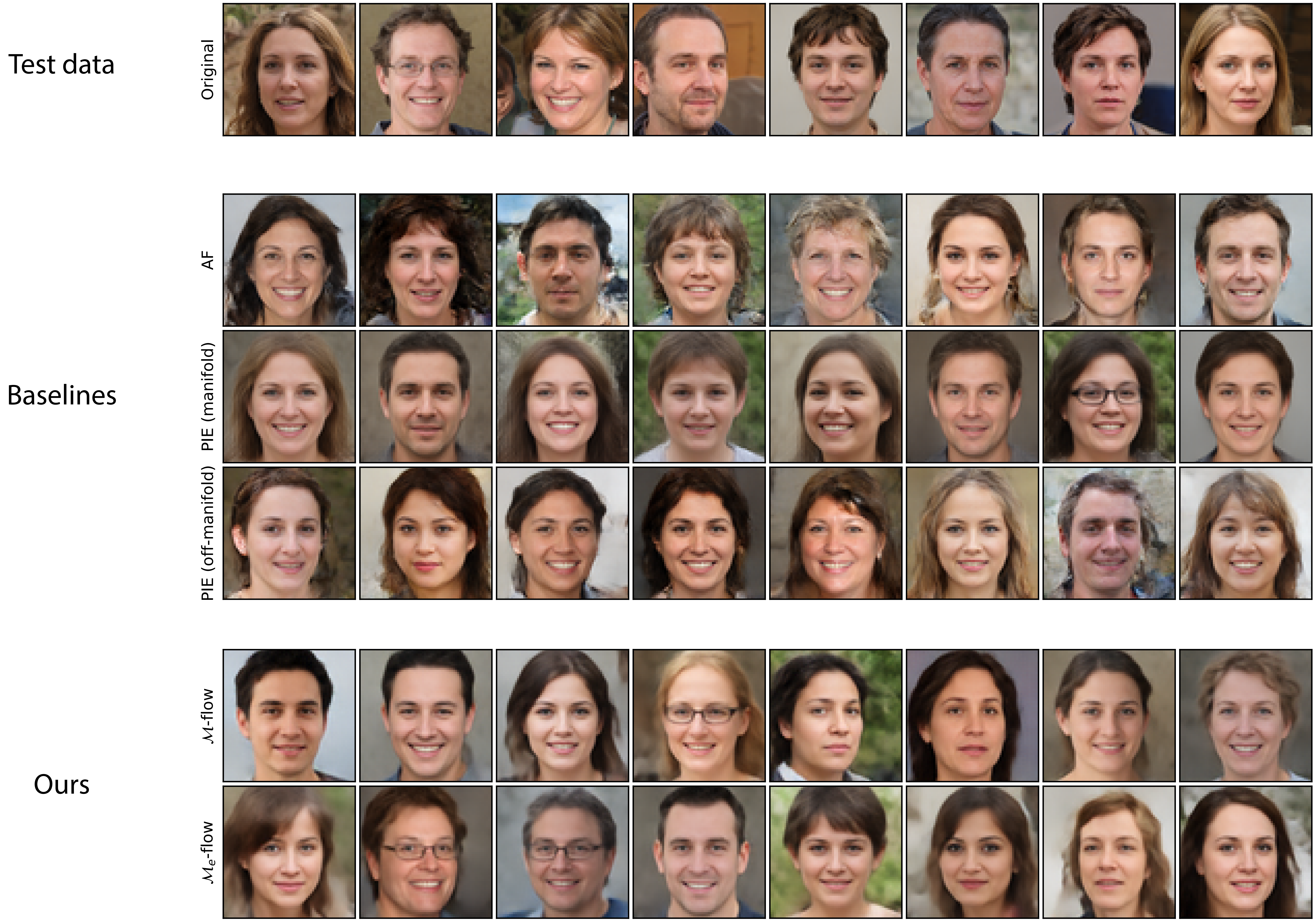
particle masses
("on-shell condition")

redundant
observables

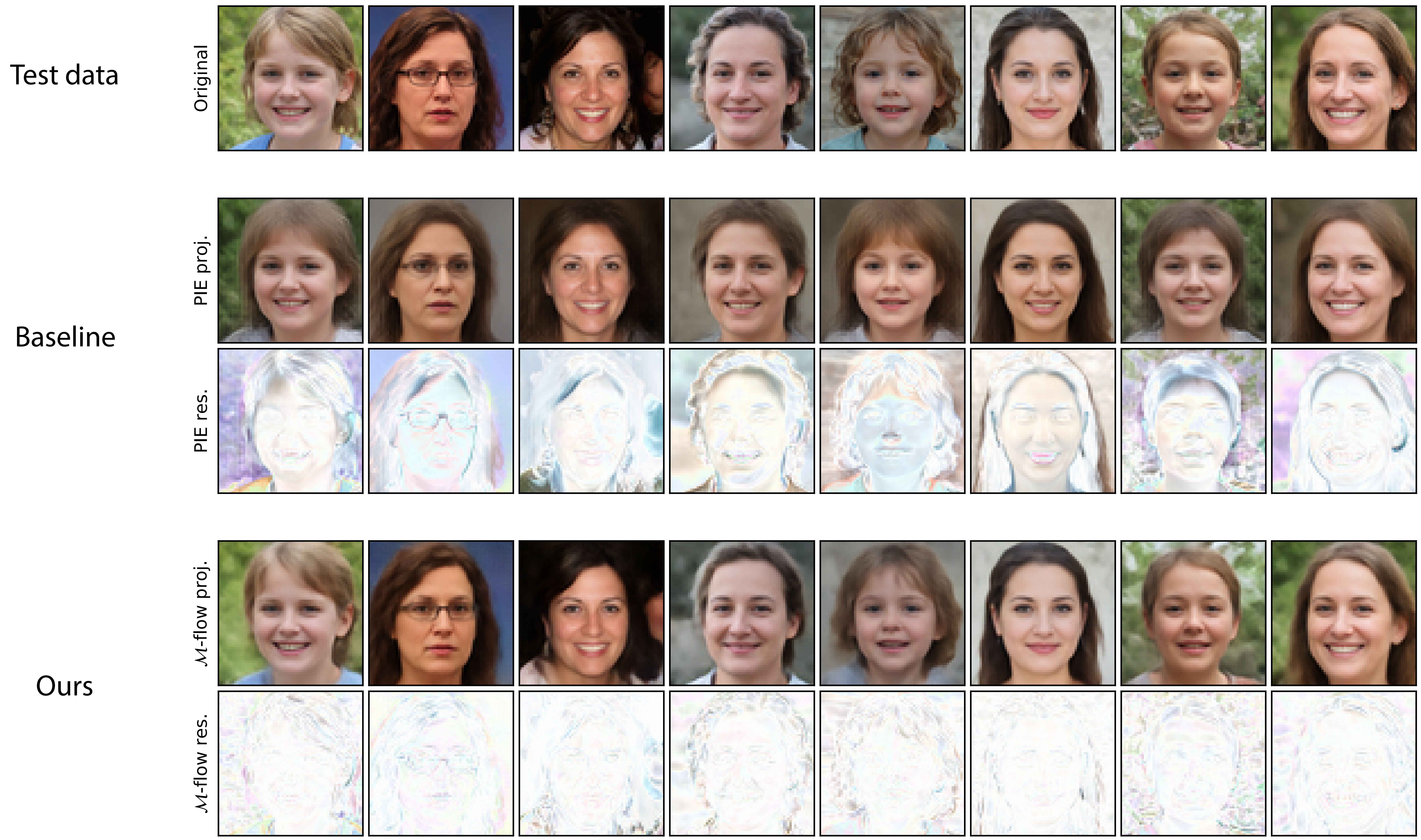
energy-momentum
conservation

14-dimensional manifold
embedded in 40-dimensional
data space

[CMS collaboration]



Test data
Original
Baselines
AF
PIE (manifold)
PIE (off-manifold)
Ours
$\mathcal{M}$-flow
$\mathcal{M}_e$-flow


Test data
Original
Baseline
PIE proj.
PIE res.
Ours
$\mathcal{M}$-flow proj.
$\mathcal{M}$-flow res.

Extended version on arXiv: 2003.13913

Flows for simultaneous manifold learning and density estimation

Johann Brehmer[a,b,1] **and Kyle Cranmer**[a,b]

[a]Center for Data Science, New York University, USA; [b]Center for Cosmology and Particle Physics, New York University, USA

June 16, 2020

We introduce manifold-learning flows ($\mathcal{M}$-flows), a new class of generative models that simultaneously learn the data manifold as well as a tractable probability density on that manifold. Combining aspects of normalizing flows, GANs, autoencoders, and energy-based models, they have the potential to represent datasets with a manifold structure more faithfully and provide handles on dimensionality reduction, denoising, and out-of-distribution detection. We argue why such models should not be trained by maximum likelihood alone and present a new training algorithm that separates manifold and density updates. In a range of experiments we demonstrate how $\mathcal{M}$-flows learn the data manifold and allow for better inference than standard flows in the ambient data space.

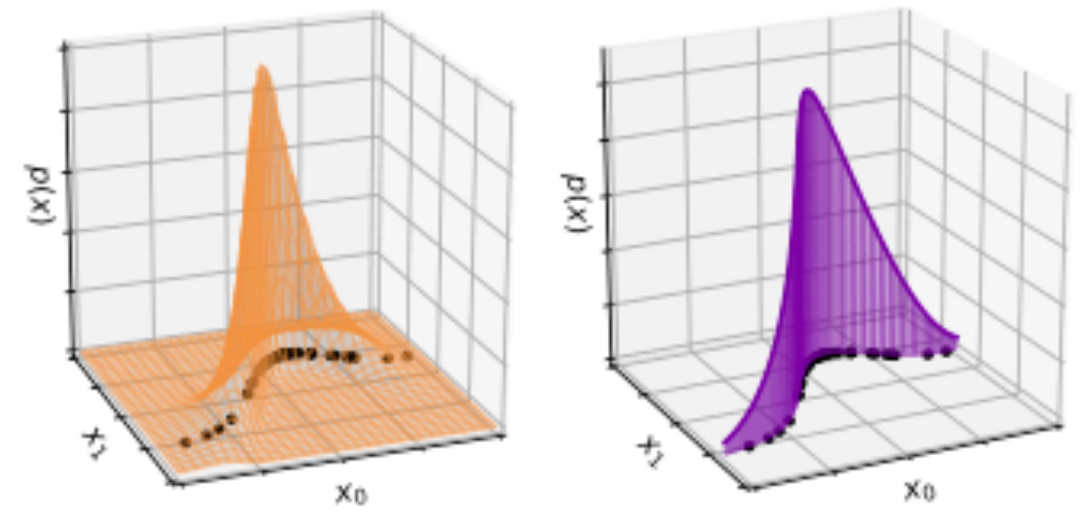


Fig. 1. Sketch of how a standard normalizing flow in the ambient data space (left, orange surface) and an $\mathcal{M}$-flow (right, purple) model data (black dots).

Contents

1. Introduction

Inferring a probability distribution from example data is a common problem that is increasingly tackled with deep generative models. Generative adversarial networks (GANs) (1) and variational autoencoders (VAEs) (2) are both based on a lower-dimensional latent space and a learnable mapping from