An introduction to generative flow models

From flow matching to diffusion

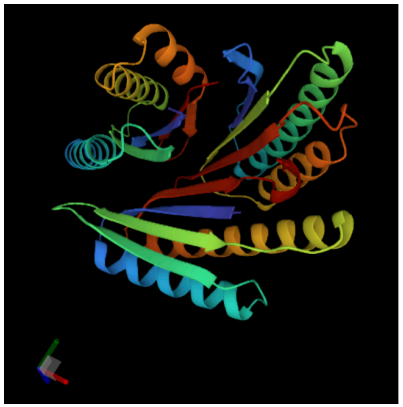
Julian Brandon - Les Gustins 2026

Neural networks and generative AI

- Image generation

Stable diffusion XL

- Protein modelling

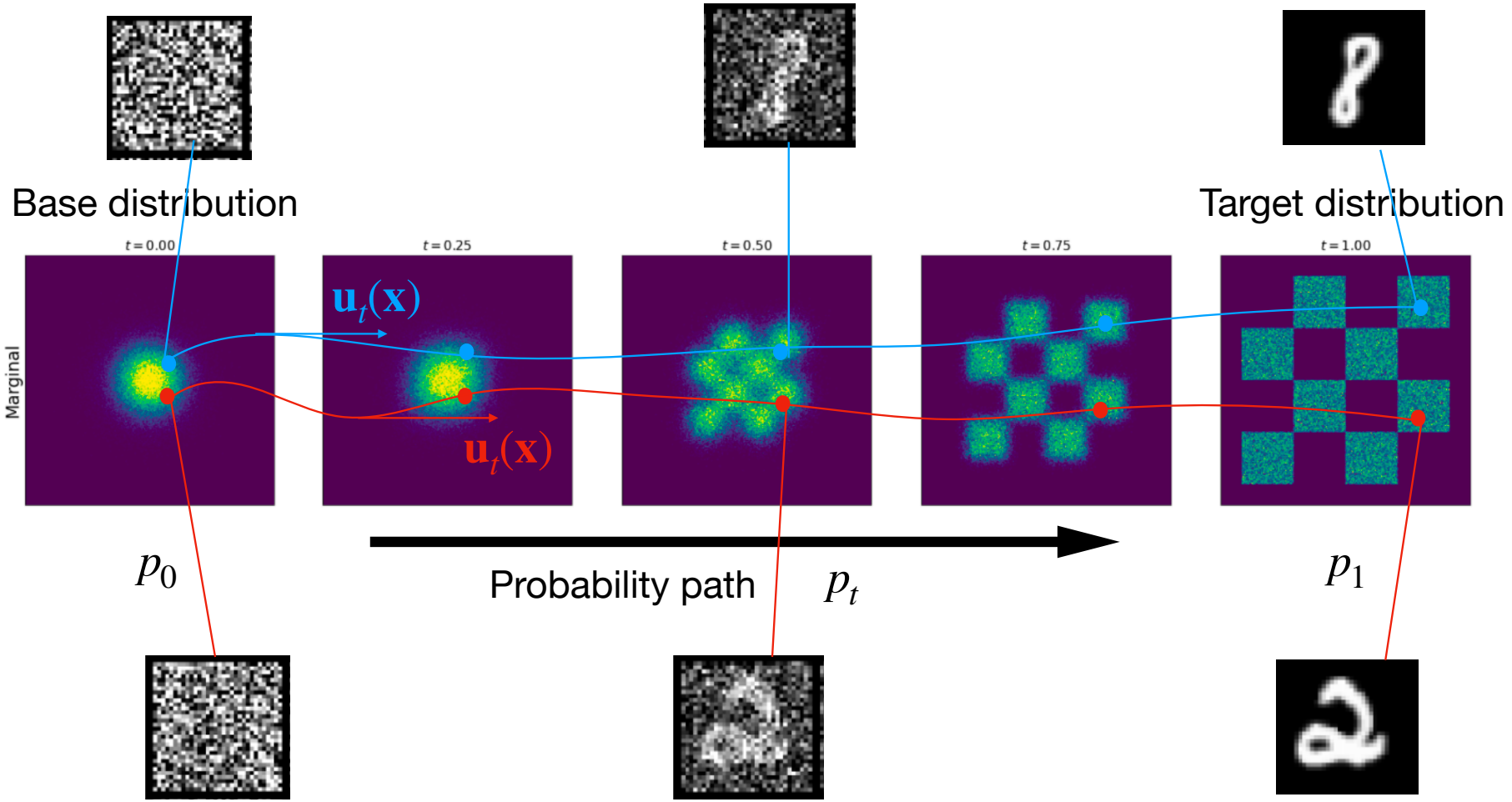


Chroma

- Robotics

Diffusion CCSP

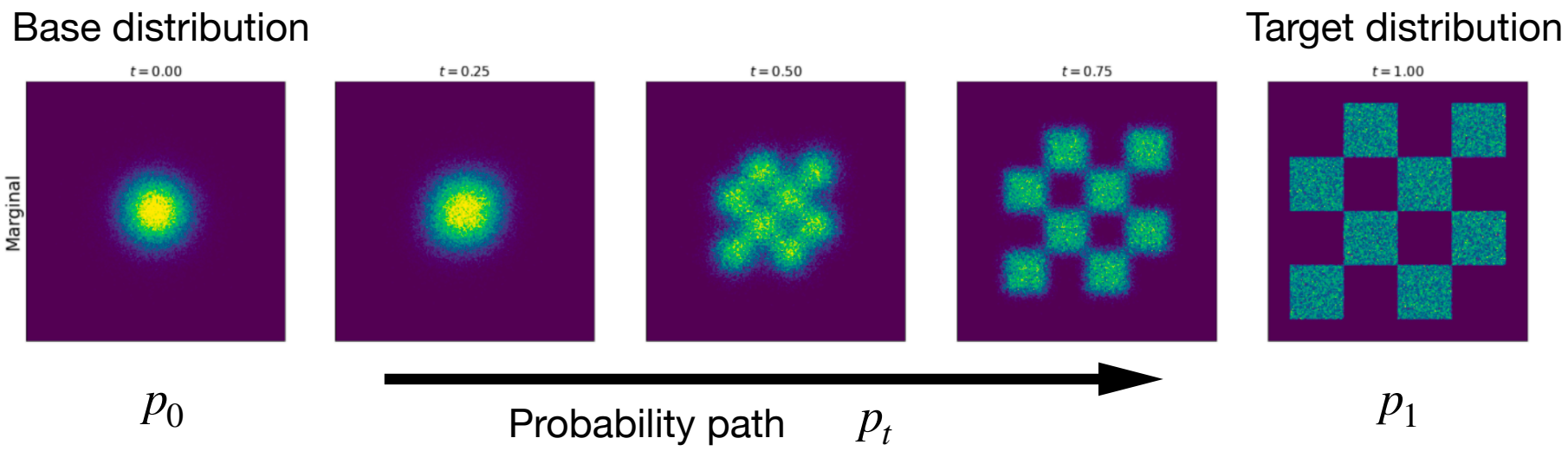
Generative flow models



Questions

1. What should be the form of p_t and $\mathbf{u}_t$?
2. How to train a neural network, $\mathbf{u}_t^\theta$, to approximate $\mathbf{u}_t$?
3. How can we extend to stochastic dynamics?

Constructing a probability path



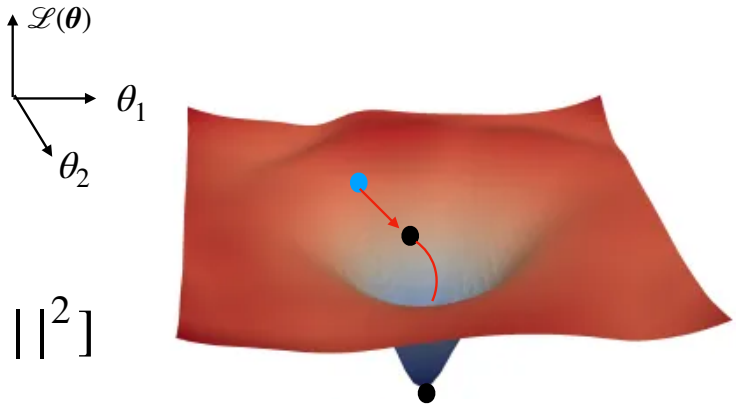
Training a neural network with gradient descent

Neural network: $\mathbf{u}_t^{\boldsymbol{\theta}}(\mathbf{x})$

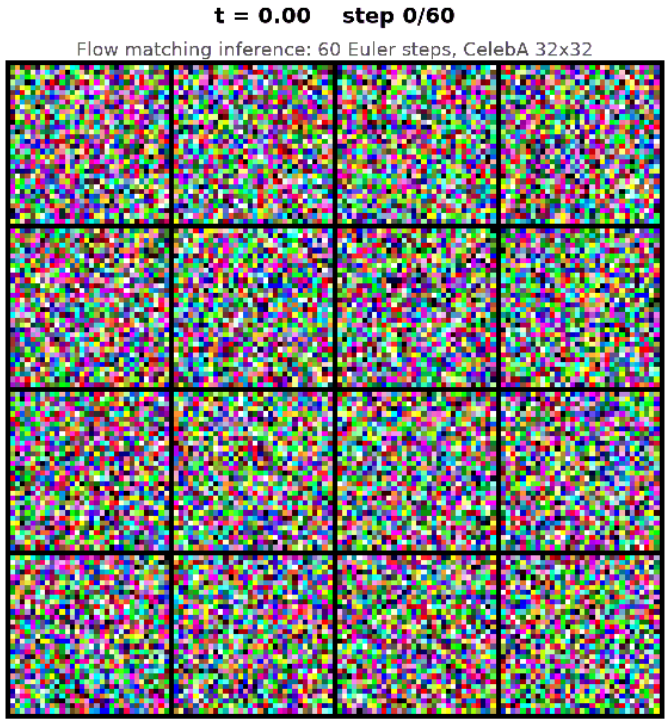
Target velocity field: $\mathbf{u}_t^{\text{target}}(\mathbf{x})$

Objective fn: $\mathscr{L}(\theta) = \mathbb{E}_{\mathbf{x}\sim p_t,t}[\,||\mathbf{u}_t^{\boldsymbol{\theta}}(\mathbf{x}) - \mathbf{u}_t^{\text{target}}(\mathbf{x})||^2]$

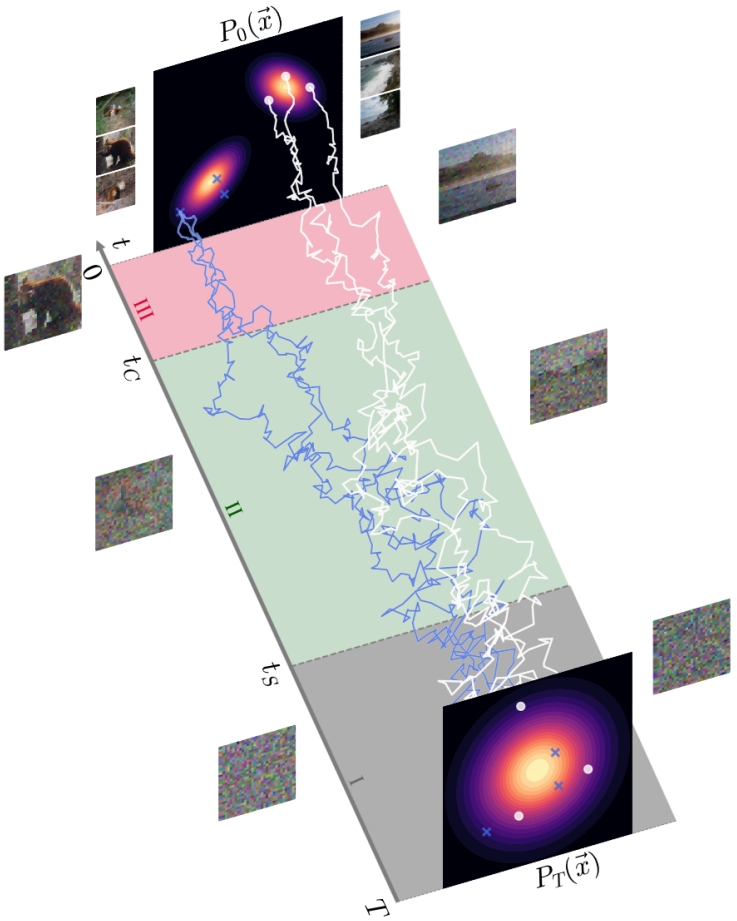
Parameter update: $\boldsymbol{\theta}_{\tau+1} = \boldsymbol{\theta}_\tau - \eta \nabla_\theta \mathscr{L}(\boldsymbol{\theta}_\tau)$



Flow matching



Diffusion



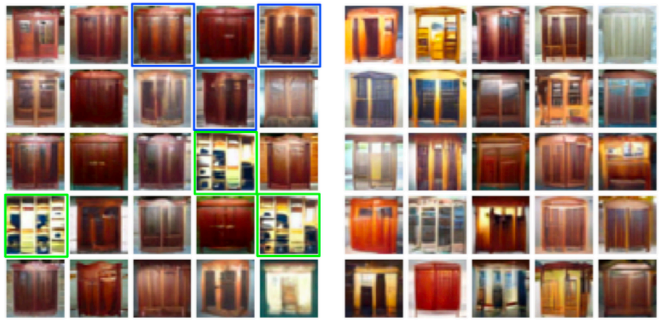
Biroli et al. 2025

Generative models show biases

Memorisation of data samples

Gu et al. 2025 TMLR

Mode collapse



Qin et al. 2023, CVPR

These models are in principle *expressive enough* to learn the right data features.
This is a bias coming from the learning dynamics.

An operator view of gradient flow

$$\mathcal{L}(W) = \mathbb{E}_{\mathbf{x}_0, \mathbb{B}}\left[\int_0^1 l(\mathbf{x}(t), t)\, dt\right]$$

Theorem (Operator View of Gradient)

The gradient of $\mathcal{L}$ with respect to a linear parameter $W \in \mathbb{R}^{m \times k}$ can be written as the composition of two linear operators:

$$\nabla_W \mathcal{L} = A \circ X$$

$$A : L^2(\mathbb{R}) \to \mathbb{R}^m, \qquad X : \mathbb{R}^k \to L^2(\mathbb{R})$$

Theorem (Stable rank bound)

The stable rank of the gradient can be bounded as:

$$\mathrm{SRank}(\nabla_W \mathcal{L}) \leq \frac{\|A\|_2 \|X\|_2}{\|A \circ X\|_2} \min\{\mathrm{SRank}(A), \mathrm{SRank}(X)\}$$

where A and X are the adjoint and state operators.

Next, we consider the flow-matching objective:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{\mathbf{x}_0}\left[\int_0^1 ||\mathbf{u}^*(\mathbf{x}^\star(t), t) - \mathbf{u}_\theta(\mathbf{x}^\star(t), t)||^2 dt\right]$$

For our linear model that is:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{\mathbf{x}_0}\left[\int_0^1 ||W^*\mathbf{x}^\star(t) - W\mathbf{x}^\star(t)||^2 dt\right]$$

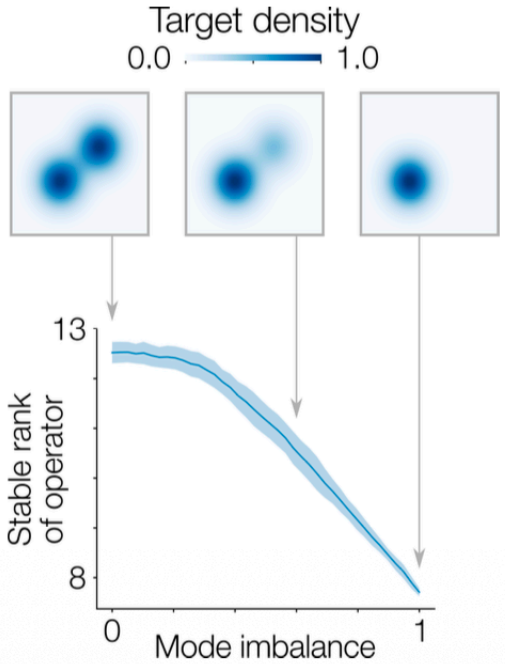
- *Data*: We will consider unequal weights in our Gaussian mixture:

$$p^*(\mathbf{x}) = \sum_{i=1}^{m} \pi_i \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_i, \sigma_i I)$$

- *Network:* We will add an additional layer to our neural network:

$$\mathbf{u}(\mathbf{x}^\star(t)) = W_2 W_1 \mathbf{x}^\star(t)$$

In that case, the operator rank continuously varies with the mixture weights.



Theorem (*Informal*)

The i'th mode takes $O\left(\pi_i^{-2}\right)$ *iterations to be learned.*