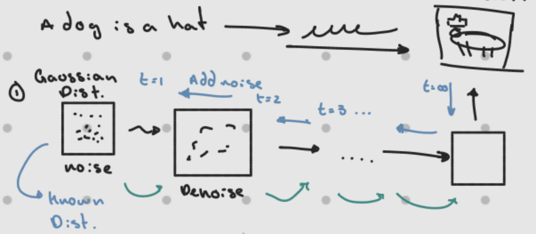
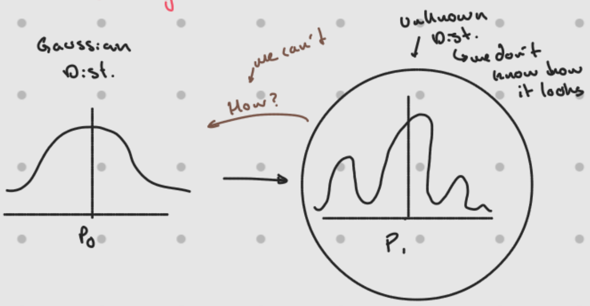
Flow matching.

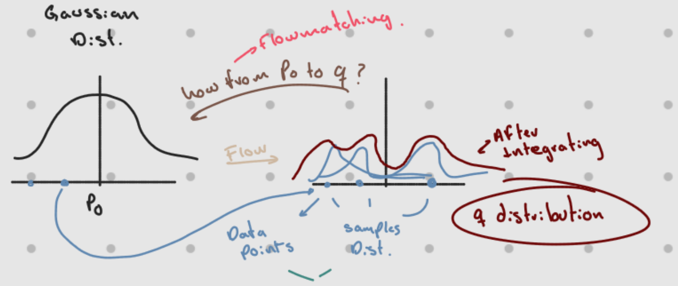
Common Diffusion models:



↳ we trace back the steps to south diff original Data

Flow-matching:





but, how we make Data-Samples? CONDITIONAL FLOWS.

Continuos Normalizing Flows

1) Probability Density Path:

$P: [0,1] \times \mathbb{R}^d \rightarrow \mathbb{R}_{>0}$

time variable

Data dist (Input)

Time-Dependent Prob. Density.

Time-Dependent function.

→ we give a time, and place in data space tells ↳ Prob. Density at that place/time.

2) Time-Dependent vector field.

$v: [0,1] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ → same but goes to Data Dist.

3) Time-Dependent diffeomorphic map [Flow]:

$\phi: [0,1] \times \mathbb{R}^d \rightarrow \mathbb{R}^d$

↳ Flow connected to vector field by:

$$\frac{d}{dt}\phi_t(x) = v_t(\phi_t(x))$$

change flow — time-Dependent; vector field — ↳ time-Dependent

$$\phi_0(x) = x$$

↳ Flow at time 0, at any point is that point

but how do we get from t_0 to t_1 dist.

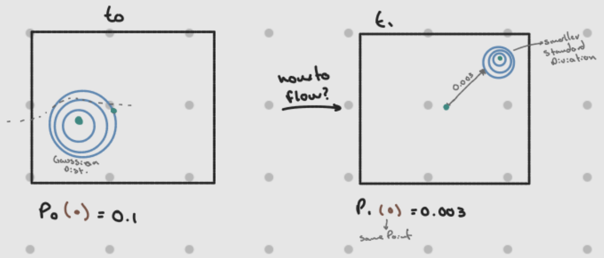
↳ ussing ODE SOLVER to solve flows until We get to $t = x$

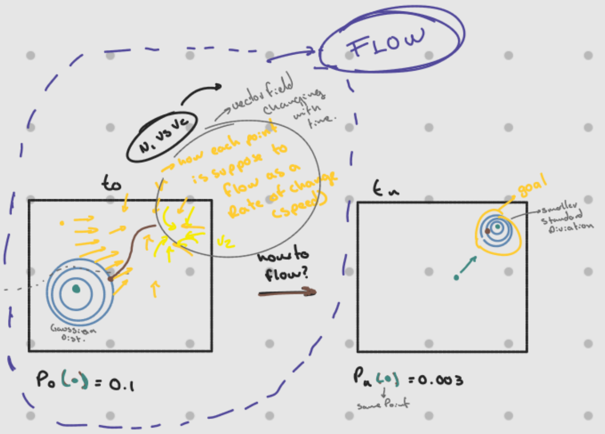
∴

$$P_t = [\phi_t]_* P_0$$

↳ Goal Dist. = Flow * Original Dist.

2D-space Data





HOW TO GET VECTOR FIELD ?

↳ what Algorithm must learn

Flow-matching:

$$\mathcal{L}_{FM}(\theta) = \mathbb{E}_{t, p_t(x)} \| v_t(x) - u_t(x) \|^2$$

how to get? — match for each pos. and time — ← how to get? ; we can't know, for now, the Prob density of target dist.

→ mide error usando Distribucion global

Loss function FM — Prob. Dens. Path — neural network — vector field (Generates Prob. Dens Path)

↳ we can Approximate by Samples with Gaussian models.

$$p_1(x) \approx q(x)$$

Constructing p_t, u_t from Conditional Probability Paths and Vector fields:

* we can Define prob Density path and Vector field in terms Individual samples.

↳ $p_t(x \mid x_1)$ → Conditional prob. path
→ $t=1$ → Data Dist.

↳ At $t=\emptyset$; $p_0(x \mid x_1) = p(x)$ → Original source no matter target Dist. and at $t=1$; $p_1(x \mid x_1)$ is a Dist. concentrated around $x = x_1$ (Data point)

↳ E.g. $p_1(x \mid x_1) = N(x \mid x_1, \sigma^2 I)$ — normal Dist. — mean Datapoint — small standard Diviation → Creates small Gaussian at point.

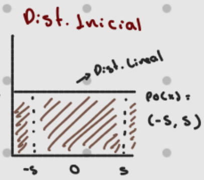
Entonces; tenemos muchos trayectorias Condicionales $p_t(x \mid x_1)$, una para cada poss. valor x_1 y queremos combinarlas en una traye. marginal $p_t(x)$

↳ Para eso Integramos todas Dist. condicionales por una dist $q(x_1)$: $p_t(x) = \int p_t(x \mid x_1) q(x_1) \, dx$.

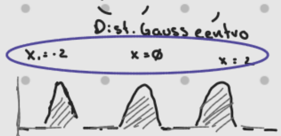
Combinación gaussianas pero con peso en Distribuciones cerca De target

Prob. density path. — marginalize target Data. — weight target Dist.

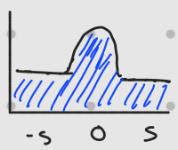
Dist. Inicial



Conditional Trajectories
* Para diff valores x_1
$x_1 = -2$; $x_1 = 0$; $x_1 = 2$
Dist. Gauss centro

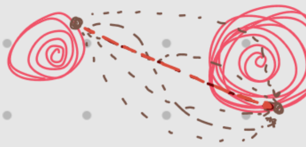


Marginalización.



→ Como tener muchas manos que jalan la dist $p(x)$ hacia varios x_1 y nosotros decidimos cuanta fuerza darle a cada mano con $q(x_1)$
Al final $p_t(x)$ es el prom. ponderado De todas las fuerzas
↳ Trayectorias cond.

* how to pass from a Sample dist to a Data dist? Straight Line.



At $t=1$ — Data Dist
$p_1(x) = \int p_1(x \mid x_1) q(x_1) dx_1 \approx q(x)$

* How to Define Vector field? By marginalizing it

$$u_t(x) = \int u_t(x \mid x_1) \frac{p_t(x \mid x_1) q(x_1)}{p_t(x)} dx.$$

we build this. — → Gaussian mixture Dist. data. — ↳ Assuming we have it — ↳ calc this.

→ how to Agregate All vector fields Across All diff. samples and get new general Vector field to put in Source Dist. to get target Dist?

* General Idea:
To balance all poss. Vector fields Into One!

Key Idea:

Marginal vector field generates marginal probability Path.
$p_t(x)$

Conditional flow matching

* Unfortunately, it is not possible to compute u_t

↳ for this Conditional flow matching was proposed.

↳ Flow matching on individual samples.

mide error usando Distribuciones pequeños condicionales y promedia.

Samples. — match. in one sample, one path.

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t,q(x_1),p_t(x|x_1)} \| v_t(x) - u_t(x|x_1) \|^2$$

target data point — Prob. path from one point source Dist going somehow to target Dist. — ↳ Neural Net. — ↳ vector field

* FM and CFM has identical Gradients.

↳ up to a constant independent θ, $\mathcal{L}_{CFM}$ and $\mathcal{L}_{FM}$ are equal! Hence Gradients of Neural Net params are equal.

params

↳ Siempre que la Dist. sea positiva, No importa que loss fn agarremos, siempre es el mismo Resultado (mismo gradients).

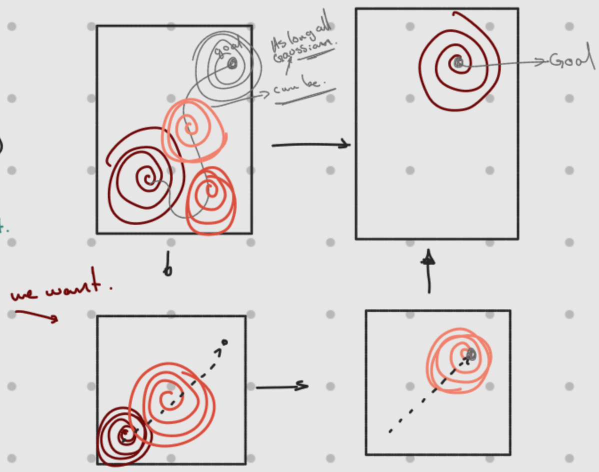
Conditional Probability Paths and Vector fields.

But, How to get Prob. Path? Until now its just Assuming how to get from some prob. path from one sample to target sample.

↳ They choose to use Gaussian — can be any!

$$p_t(x|x_1) = \mathcal{N}(x \mid \mu_t(x_1), \sigma_t(x_1)^2 I)$$

mean of Gaussian for any point — ↳ standard Diviation any point.



→ At $t=0$; $p(x) = \mathcal{N}(x|0,I)$ — match original Dist.

→ At $t=1$, $\mu_1(x_1) = x_1$, and $\sigma_1(x_1) = \sigma_{min}$

mean At Data point — small std to center gaussian to Data point

How to move (transform) the Distribution in a soft way?

$$\psi_t(x) = \sigma_t(x_1)\,x + \mu_t(x_1)$$

initial point — objective

x pos in time t in a Trayectory. — escale factor, tells us how much x we want. — movement term, where we want to go.

→ we are Interpolating. x and x_1 in time Dependent. where $\psi_t(x)$ tells path from x to x_1

we need a push foward function to actually move it!

$$[\psi_t]_* p(x) = p_t(x|x_1)$$

flow:

Then, the flow gives us a vector field that generates conditional Prob. Path.

$$\frac{d}{dt}\psi_t(x) = u_t(\psi_t(x)|x_1).$$

∴ Loss function:

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t,q(x_1),p(x_0)} \| v_t(\psi_t(x_0)) - \frac{d}{dt}\psi_t(x_0) \|^2$$

Let $p_t(x|x_1)$ be Gaussian prob. paths and ψ_t its flow map, THEN VECTOR FIELD

$$u_t(x|x_1) = \frac{\sigma_t'(x_1)}{\sigma_t(x_1)}\left(x - \mu_t(x_1)\right) + \mu_t'(x_1)$$

!!

Generates Gaussian paths $p_t(x|x_1)$

↳ vector field at any point in time!

But, how to get straight line?

↳ Direction Vector field remains Constant

↳ Optimal Transform (OT)