Marilyn CHAHINE, 21517286
Jean MACQUET, 3532408

TP 3-a – Flow Matching

Summary

In this TP, we studied flow matching models, a type of neural network that generates data step by step by learning a velocity field mapping an initial distribution to a target distribution. Each step corresponds to one evaluation of the network, allowing the model to progressively refine samples.
We compared linear and cosine schedules for interpolating between distributions. Linear schedules do not preserve variance, which causes trajectories to curve inward and outward during generation, while cosine schedules maintain constant variance, resulting in smoother, more direct trajectories. We saw that in both Gaussian and make-moons target distributions, where cosine schedules produce straighter paths, and linear schedules produce curved trajectories.
Finally, we contrasted flow matching and diffusion models with GANs. GANs generate samples in a single forward pass and require balancing generator and discriminator performance. Flow matching/diffusion models, though computationally more demanding due to stepwise generation, train directly on the target distribution and provide better coverage, more accurate representations, and a more straightforward learning process.

Questions

1. The border conditions:

$$\alpha_0 = \beta_1 = 1 \text{ and } \alpha_1 = \beta_0 = 0$$

And we have that:

$$\alpha_t = \cos(at), \beta_t = \sin(bt)$$

So, the smallest values we can have for a and b while respecting the border conditions are:

- α_0 = cos(a*0) = cos(0) = 1 for any value of a
- α_1 = cos(a) = 0 so a = pi/2
- β_0 = sin(b*0) = sin(0) = 0 for any value of b
- β_1 = sin(b) = 1 so b = pi/2

Marilyn CHAHINE, 21517286
Jean MACQUET, 3532408

2. We have:
   $X_t = \alpha_t X_0 + \beta_t X_1$

   Which gives

   $\frac{dX_t}{dt} = \dot{\alpha}_t X_0 + \dot{\beta}_t X_1$

   With

   $\alpha_t = cos(\frac{\pi}{2}t)$ so $\dot{\alpha}_t = -\frac{\pi}{2}\sin(\frac{\pi}{2}t)$

   And

   $\beta_t = \sin(\frac{\pi}{2}t)$ so $\dot{\beta}_t = \frac{\pi}{2}\cos(\frac{\pi}{2}t)$

3. If $t = \frac{1}{2}$ then $X_t = \frac{1}{2}(X_0 + X_1)$ so $X_t = \frac{1}{2}\big(\mathrm{N}(0,1) + \mathrm{N}(1,1)\big) = \frac{1}{2}N(1,2) = N(\frac{1}{2},\frac{1}{2})$
   So interpolation of X_t at t=0.5 changed the variance of the data, which means the linear schedule is not variance preserving.

4. With the cosine schedule, $X_t = \cos\left(\frac{\pi}{2}t\right)X_0 + \sin\left(\frac{\pi}{2}t\right)X_1$

   So $Var(X_t) = \cos\left(\frac{\pi}{2}t\right)^2 1 + \sin\left(\frac{\pi}{2}t\right)^2 1 = 1$

   So, the variance is constant for all t. Cosine schedule is variance preserving.

5. Varying variances means the generated points don't lie across a constant distance from the origin during data generation. With the linear schedule, when t = 0.5, variance dips to 0.5, which means the points get closer to the origin and then get further again after t=0.5 (since variance goes back to 1).
   Meanwhile, for the cosine schedule, the variance is always equal to 1, so the generated points lie at an equal distance from the origin for all timesteps.
   That results in smoother trajectories being generated in the case of cosine schedule since the flow 'slides' across a circle instead of going inward and outward.

Marilyn CHAHINE, 21517286
Jean MACQUET, 3532408

6. The geometric differences we mentioned are exactly what we see in our code.

   In the case of Gaussian target data, when using the linear schedule, we can see that the variance changes lead to trajectories that are curved (Figure 1). The variance dip around 0.5 pushes the trajectory towards the 'center' of the distribution and then it goes back to variance 1 so it gets further away from the center again. Meanwhile the cosine schedule trajectories for gaussian target data are straight since there are no variance changes across time (Figure 2). We can thus see that variance preservation leads to shorter trajectories since path generation is not perturbed by inward/outward motion.

   Similarly, in the make moons target data case, the linear schedule trajectories show clear curves (Figure 3) while the cosine schedule ones take the shortest path (Figure 4).

   We also computed the variance of the generated points at each timestep in each of the four cases, and found, as expected, that the variance of the points gradually decreases and then goes back up when using the linear schedule (from 0.86 at t=1 to 0.52 at t=6 and back to 0.91 at t=10 for Gaussian data, and from 0.87 at t=1 down to 0.38 at t=6 and back to 0.57 at t=10 for make moons). For the cosine schedule, the variance of the generated points stays consistent around 1.05 at all timesteps for gaussian data, but it decreases for make moons, which can be explained by the fact that the make moons target distribution is not gaussian so the variance preservation argument does not hold.

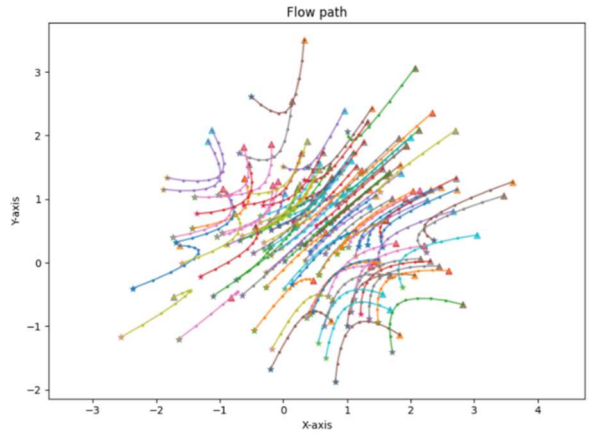

Figure 4 Trajectories generated by using the linear schedule with make moons as target distribution

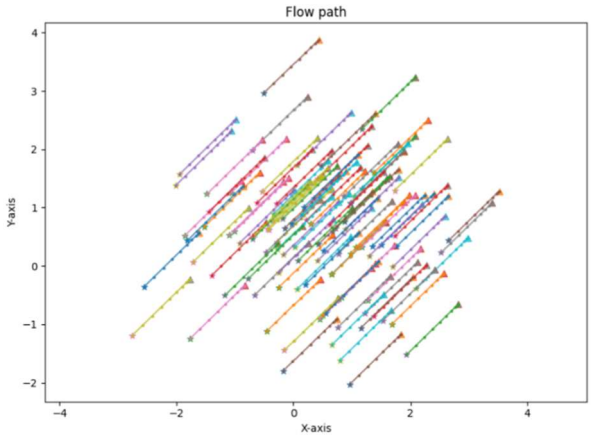

Figure 3 Trajectories generated by using the cosine schedule with make moons as target distribution

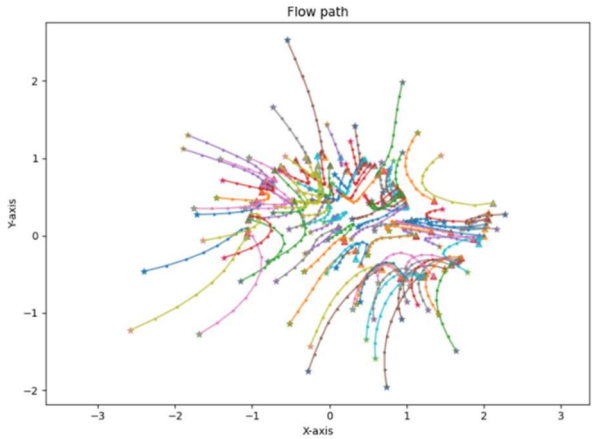

Figure 1 Trajectories generated by using the linear schedule with N(1,1) as target distribution

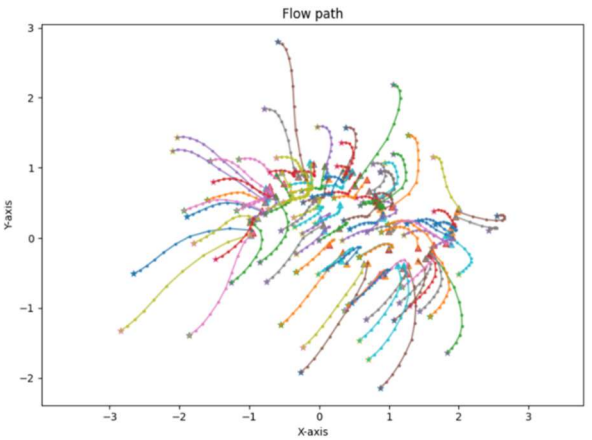

Figure 2 Trajectories generated by using the cosine schedule with N(1,1) as target distribution

Marilyn CHAHINE, 21517286
Jean MACQUET, 3532408

7. GANs rely on a minimax adversarial optimization scheme which requires balancing generator and discriminator performance. Flow matching and diffusion models are more straightforward, they are trained directly on the target data using a simple regression loss function, without needing to balance two different networks competing each other. Flow matching also guarantees a better coverage of the target distribution and less mode collapse since it maps the initial distribution to all the points in the target.

8. Computationally, if both a GAN and flow matching/diffusion model have the same number of parameters (as seen in Q9), the GAN can generate an image through one forward pass of the network while the FM/Diff model will need to go through a pass of the network for each timestep. So, the FM/Diff model would require a lot more computation. Therefore, FM/Diff models require significantly more computation at sampling time.

9. GANs generate an image from latent space through a single forward pass of the network, while Flow Matching divides the generation process into multiple small transformations from the initial distribution at t=0 the final generated image at t=1. This decomposition allows the model to progressively refine the generated sample, with each step making subtle corrections. As a result, even with the same network size, Flow Matching can represent complex image structures more accurately than GANs, which must generate the entire image in one pass.