Metric learning - contrastive learning

Julie Digne

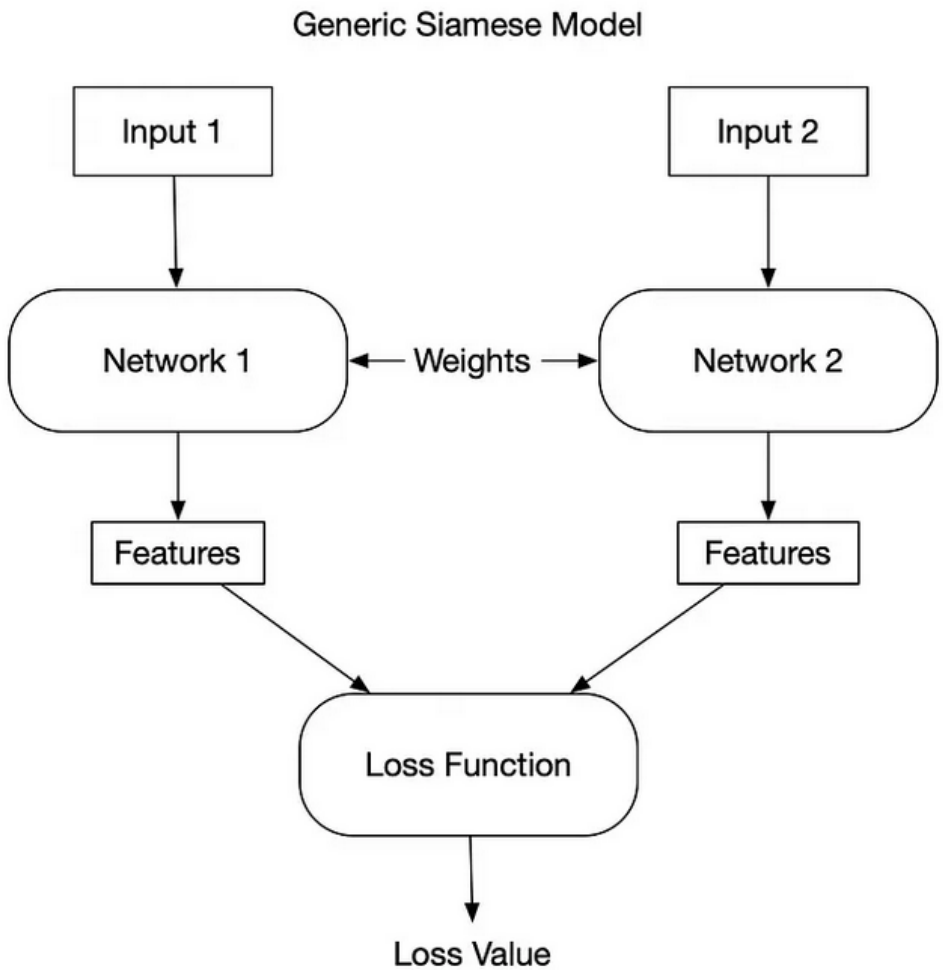
Master ID3D
LIRIS - CNRS
Équipe Origami

20/11/2025

Learning a distance

- Embed the objects in a high dimensional space
- Compute the distance between two objects
- Various cases:
  - Regular distance is too hard/expensive to compute (e..g. optimal transport...)
  - You cannot compute the distance (e.g. style...)

Siamese Network



Medium/Rinki Nag

Generic siamese network.

Loss functions: Distance

- The distance is known from pairs of points (p_i, q_i)
- Use this distance to train the embedding

Distance loss

network f_θ, pairs (x_i, y_i) with known distance d_i for $i = 1 \cdots n$

$$l(\theta, (x_i, y_i)) = \sum_i \Big| |f_\theta(x_i) - f_\theta(y_i)| - d_i \Big|$$

Loss function: pairwise siamese loss

- The distance cannot be computed..
- But one can have positive/negative examples

Pairwise loss $L(X)$

$$L^+(x, x^+) = \|f_\theta(x) - f_\theta(x^+)\|_2 \text{ et } L^-(x, x^-) = max(0, m - \|f_\theta(x) - f_\theta(x^+)\|_2)$$

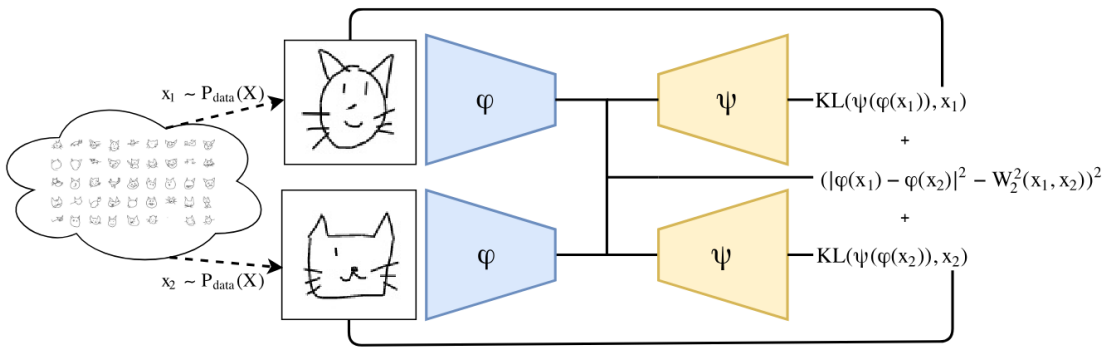
So that:

$$L(X) = \sum_i L^+(x_i, x_i^+) + \sum_j L^-(x_j, x_j^-)$$

Loss functions: triplet loss

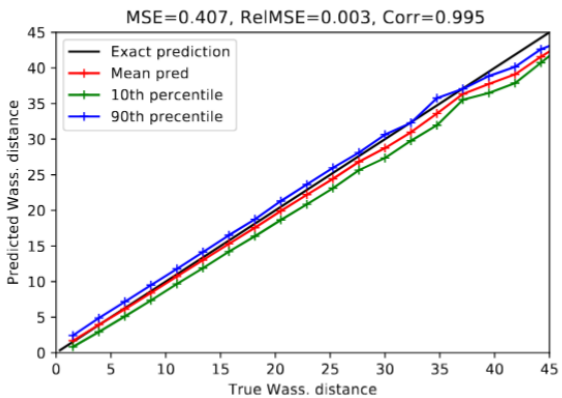
- The distance cannot be computed..
- But one can have positive/negative examples

An example in Optimal Transportation [Courty 2018]



Wasserstein distances are too expensive to compute: learn them.

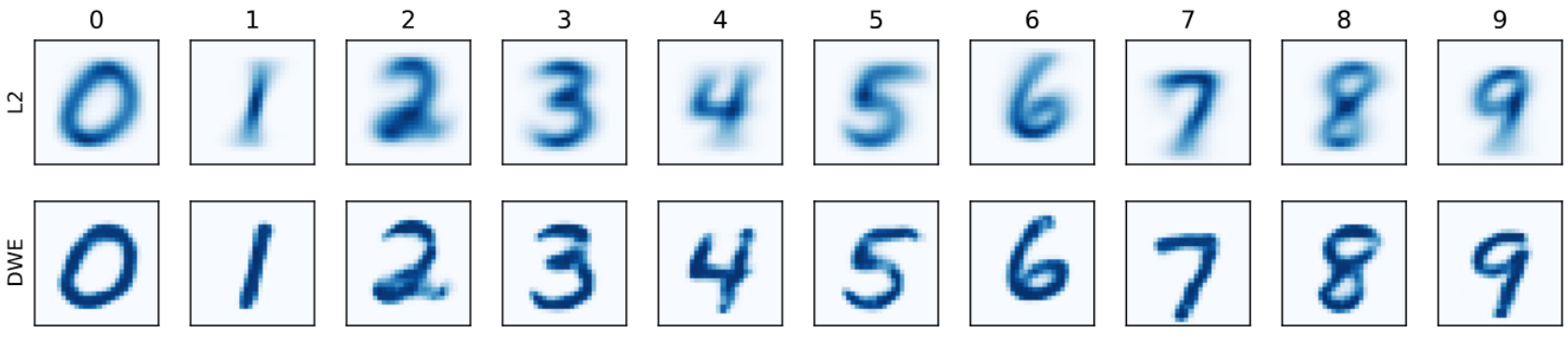
Time comparison



Method	W_2^2/sec
LP network flow (1 CPU)	192
DWE Indep. (1 CPU)	3 633
DWE Pairwise (1 CPU)	213 384
DWE Indep. (GPU)	233 981
DWE Pairwise (GPU)	10 477 901

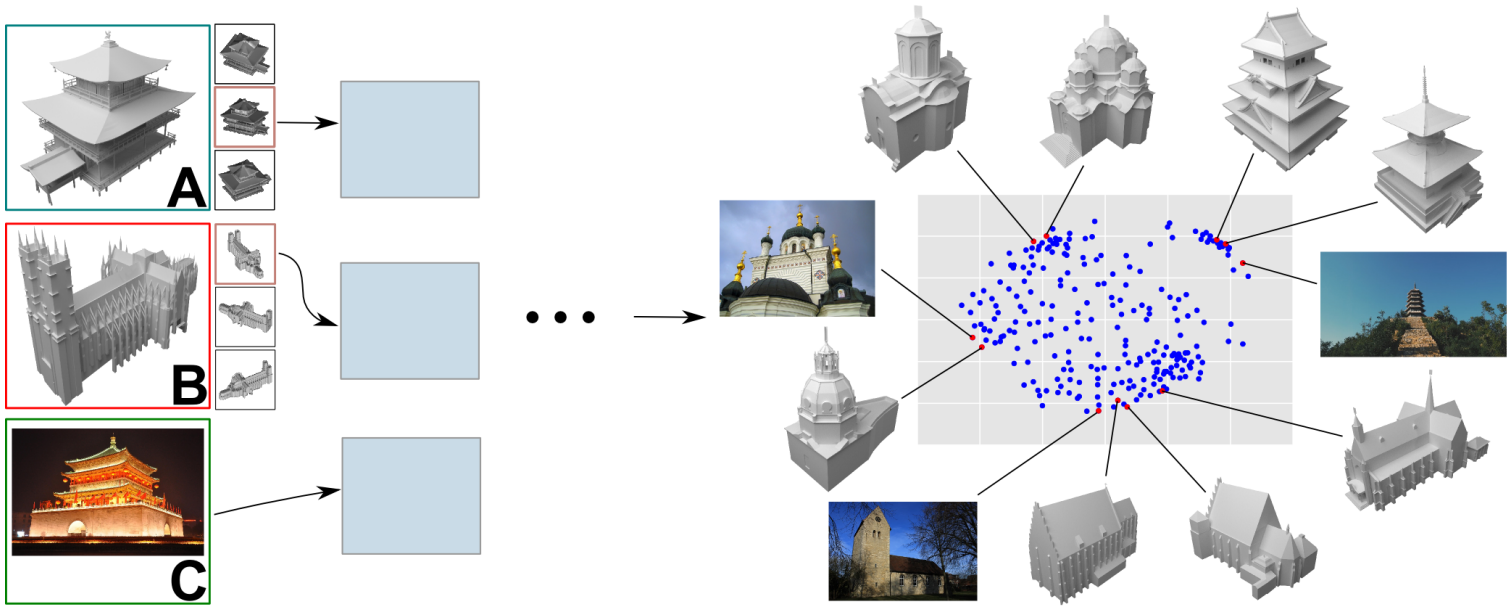
[Courty 2018]

Results



[Courty 2018]

An example for style comparison [Lim 2016]

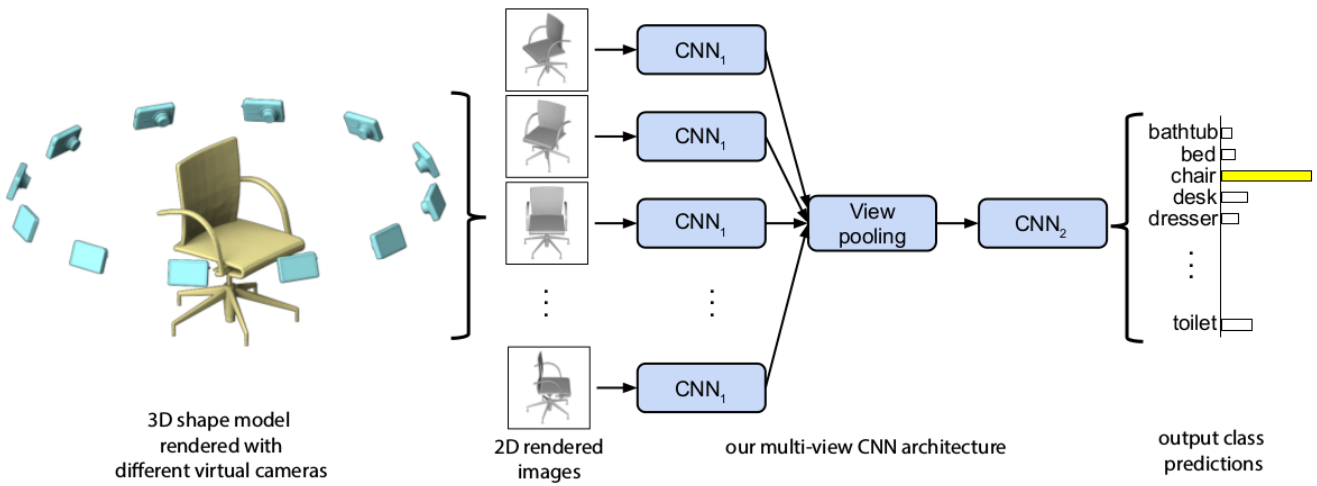


[Lim 2016]

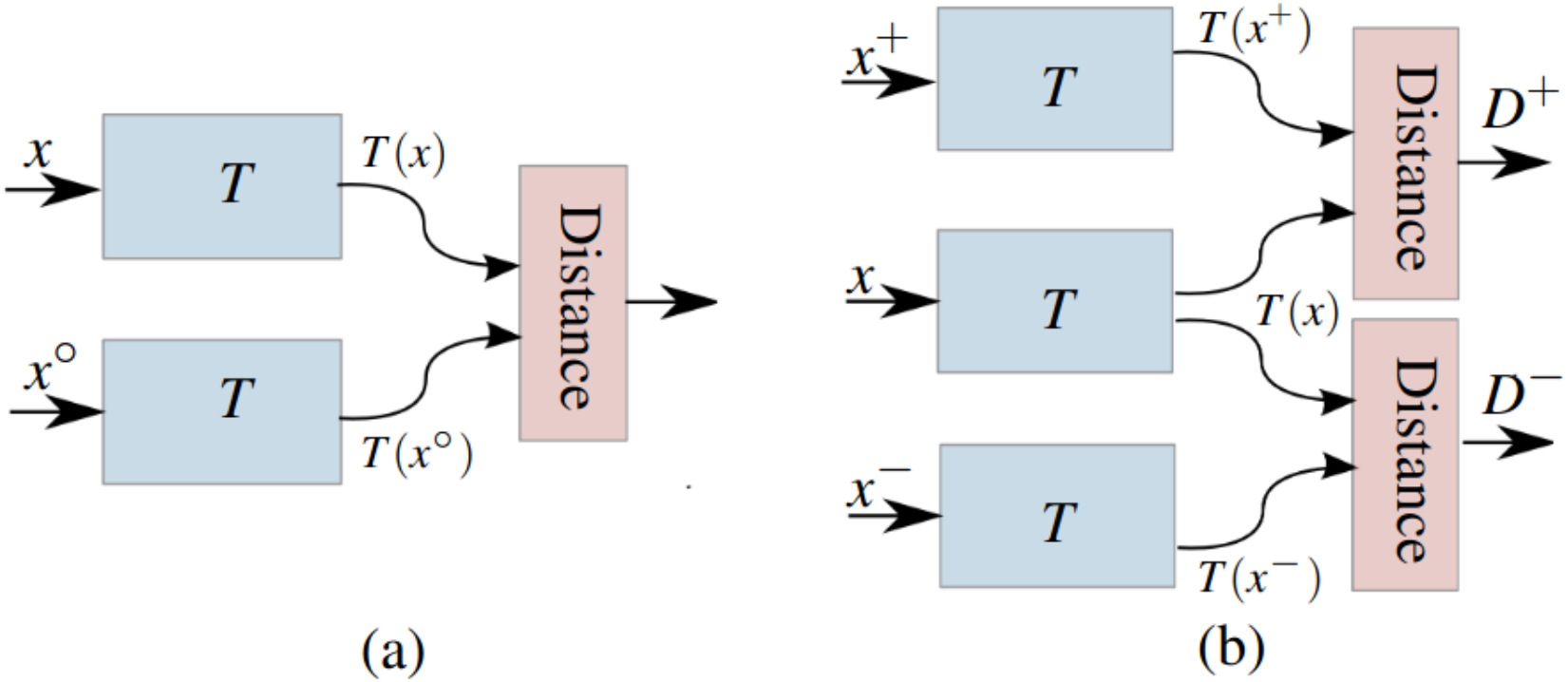
- Converting the perceived style similarity into a distance measure.

Data

- A set of 3D shapes with metadata: 2 shapes look alike of not.
- Works with set of renderings (multiview cnn).

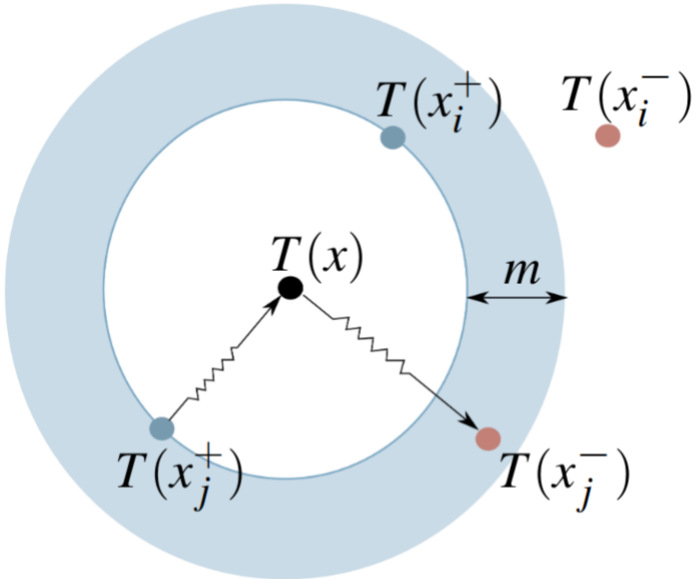


Triplet loss



- Siamese network: Learn from pairs (x, x^-) et (x, x^+)
- Triplet network: Learn from triplets (x, x^+, x^-)

Triplet loss



Loss $L(X)$

$$D^+(x, x^+) = \|T(x) - T(x^+)\|_2 \text{ and } D^-(x, x^-) = \|T(x) - T(x^-)\|_2$$

$$L(x, x^+, x^-) = max(0, m + D^+(x, x^+) - D^-(x, x^-))$$

Finally:

$$L(X) = \frac{1}{n} \sum_i L(x_i, x_i^+, x_i^-)$$

- Attract things that look alike and spreads apart things that are different

Résultats

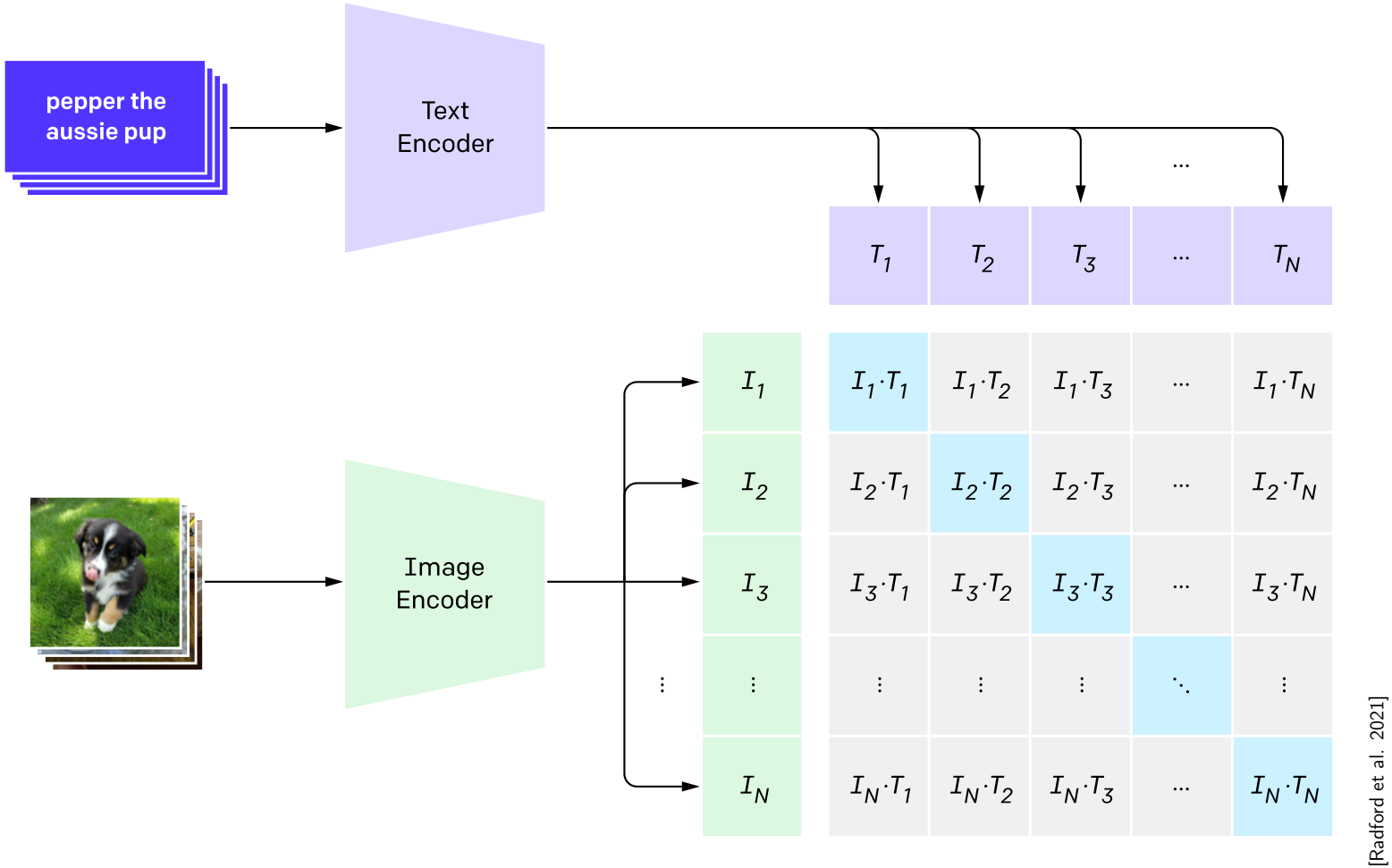
Contrastive Loss

- A set of data x_i with labels y_i
- Minimize the same-label distance and maximize the inter label distance

$$L_\theta(x, y) = 1_{y_i = y_j} \| f_\theta(x_i) - f_\theta(x_j) \|_2^2 + 1_{y_1 \neq y_j} max(0, \varepsilon - \| f_\theta(x_i) - f_\theta(x_j) \|_2)$$

CLIP [Radford et al. 2021]

1. Contrastive pre-training



[Radford et al. 2021]

Learns which caption goes with which image.

CLIP

Crucial to text-to-image methods. See Nicolas' course on Diffusion model