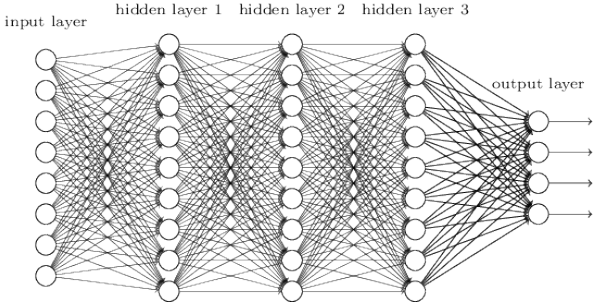
Neural Networks and Deep Learning: Manifold Untangling & Visualization

Nicolas Thome

Conservatoire National des Arts et Métiers (Cnam)
Département Informatique

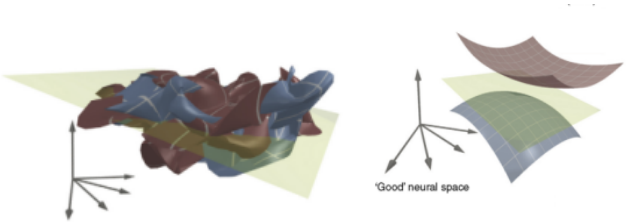
Representation Learning & Deep Learning

- X-class classification, K classes: last hidden layer size $L \to K$
- Classification layer: linear projection + soft-max activation
  - **In $\mathbb{R}^L$ space, linear separation between classes**
  - **Deep Learning (backprop) supports learning representations that gradually project data to $\mathbb{R}^L$ spaces where linear separation possible**



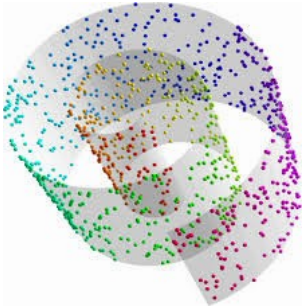
Deep Learning & Manifold Untangling

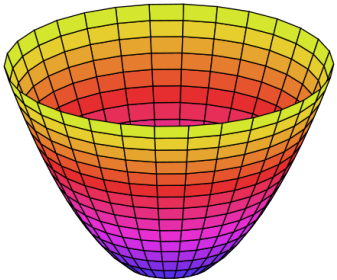
- DL: gradually projecting data to $\mathbb{R}^L$ spaces where linear separation possible
- **This is the definition of manifold untangling!**



- ConvNets: manifold untangling easier!
  - Conv/Pool *prior*: stability

Manifold Untangling Visualization

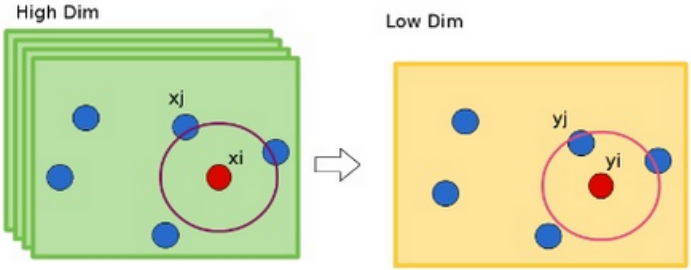




- We want to visualize each layer activation for each class
- high-dimensional visualization?
  $\Rightarrow$ Projection to lower (*e.g.* 2d) dimensions

t-distributed Stochastic Neighbor Embedding (t-SNE)

- t-SNE [van der Maaten and Hinton, 2008]: non linear projection
- Intuitively: close distances in initial space $\Rightarrow$ close distances in projected (2d) space
  - Distance preservation
  - Neighborhood preservation *i.e.* small distance



t-SNE [van der Maaten and Hinton, 2008]

- Similarity between points $(\mathbf{x}_i, \mathbf{x}_j)$ in initial space, *e.g.* $\mathbb{R}^d$:

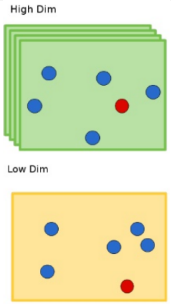
$$p_{ij} = \frac{e^{-\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{2\sigma^2}}}{\sum_{k \neq l} e^{-\frac{||\mathbf{x}_k - \mathbf{x}_l||^2}{2\sigma^2}}} \qquad P = \{p_{ij}\}_{(i,j) \in N \times N}$$

- Similarity between points $(\mathbf{y}_i, \mathbf{y}_j)$ in projected space, *e.g.* $\mathbb{R}^2$:

$$q_{ij} = \frac{(1 + ||\mathbf{y}_i - \mathbf{y}_j||^2)^{-1}}{\sum_{k \neq l} (1 + ||\mathbf{y}_k - \mathbf{y}_l||^2)^{-1}} \qquad Q = \{q_{ij}\}_{(i,j) \in N \times N}$$

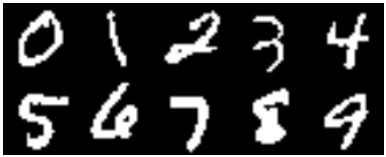
- Loss function: Kullback-Leiber divergence $KL(P||Q)$

$$C = \sum_i KL(P||Q) = \sum_i \sum_j p_{ij} \; log \frac{p_{ij}}{q_{ij}}$$



t-SNE Visualization: MNIST example

- MNIST dataset: 28×28 grayscale images of digits
- 10 classes $\Leftrightarrow$ digit number $\in \{0; 9\}$
- Input space dimension: $28^2 = 784$
- Projection in 2d (3d) space for visualization



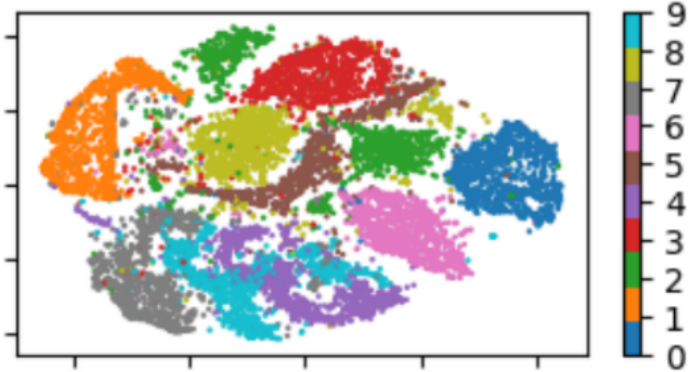
- t-SNE for computing projection: gradient descent

$$\frac{\partial C}{\partial y_i} = 4 \sum_i (p_{ij} - q_{ij})(\mathbf{y}_i - \mathbf{y}_j)(1 + ||\mathbf{y}_i - \mathbf{y}_j||^2)^{-1}$$

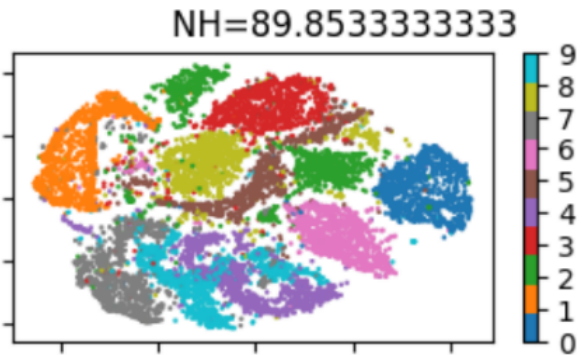
- Optimization (projection) for a given closed dataset $\Rightarrow$ transductive learning

t-SNE Visualization: MNIST example

- Application of t-SNE in the test set of MNIST (10000) images
- Color $\Leftrightarrow$ class ID



t-SNE Visualization: MNIST example

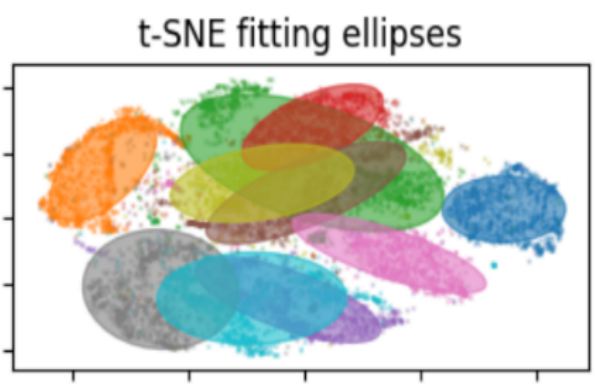


- Classes visually appear in 2d space, **BUT overlap**
- How to measure class separability?

Neighborhood Hit [Paulovich et al., 2008]:

$$\text{NH} = \frac{\#\text{ pts in knn of the same class}}{\#\text{ pts in knn}}$$

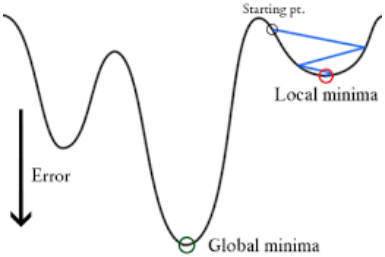
t-SNE Visualization: MNIST example



- How to measure class separability?
  - Fitting ellipses to each class points
  - **Ellipses non-overlap ⇒ linear separability**

Manifold Untangling & Visualization: Conclusion

- Deep Learning: gradual learning of untangled representations
- t-SNE projection: High-dimensional space ⇒ 2d
  - Separability Measure: NH, ellipses overlap
  - **Better separability of learned deep representations? ⇒ following**
- **Deep Learning Weaknesses?**
  ⇒ following!



References I

Paulovich, F. V., Nonato, L. G., Minghim, R., and Levkowitz, H. (2008).
Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping.
IEEE Trans. Vis. Comput. Graph., 14(3):564–575.

van der Maaten, L. and Hinton, G. E. (2008).
Visualizing high-dimensional data using t-sne.
Journal of Machine Learning Research, 9:2579–2605.