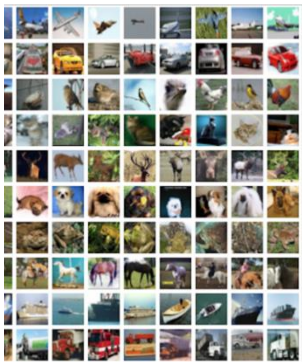
Stochastic Deep Networks

10th July 2018 – Mokaplan

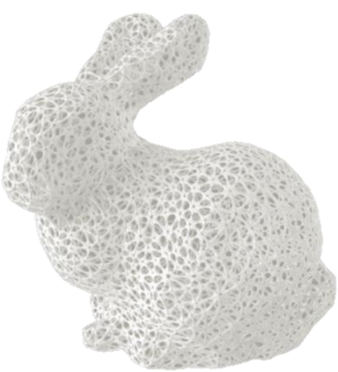
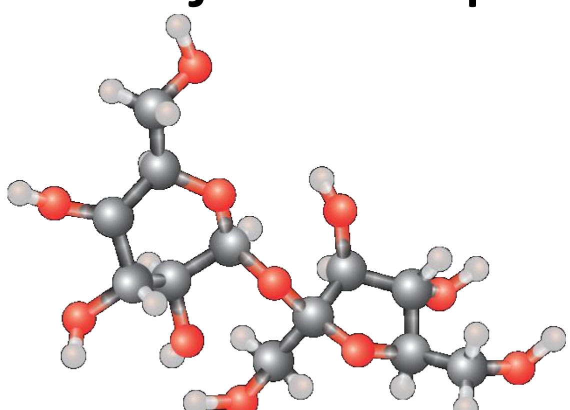
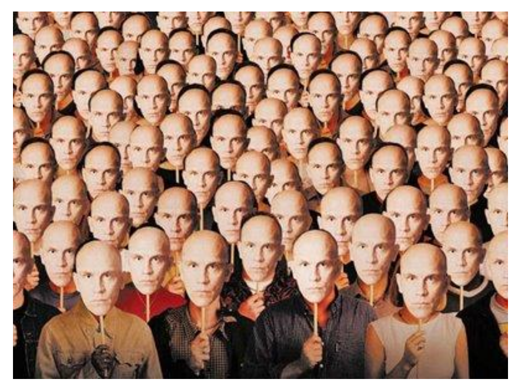
Gwendoline De Bie, with Gabriel Peyré and Marco Cuturi

Introduction

- DL performs well on data with Euclidean structure



- But in many applications, **densities**: appropriate objects to perform ML on

State of the art

- Point sets

  → Unordered, permutation invariance/equivariance, locality

- Characterization of layers equivariant w.r.t permutations / groups of transformations: S. Ravanbakhsh et al (2016,2017)
- **Pairwise interactions** come in handy: Mallat et al (2014, 2016), N. Guttenberg et al (2016): $\text{pooling} \circ f(x_i, x_j)$ $(\sum \text{or max})$
- Augment training data by permutations or find « best » ordering: O. Vinyals et al (2015)
- **PointNet** and PointNet++ (C. Qi): $\gamma \circ \max_{x_i \in C} h(x_i)$

State of the art

- As **graphs**: Mémoli, Sapiro (2005), Bronstein et al (2006) to Bruna et al (2015), Y. Li (2015), Bronstein et al (2017)

- **Wasserstein** metrics in DL:
  - **Generative** purposes: Bottou et al (2017), Genevay et al (2017)
  - **Dynamic** networks: Frogner et al (2015), Hashimoto et al (2016)

Outline

I. Proposed layers

II. Theoretical results

III. Applications

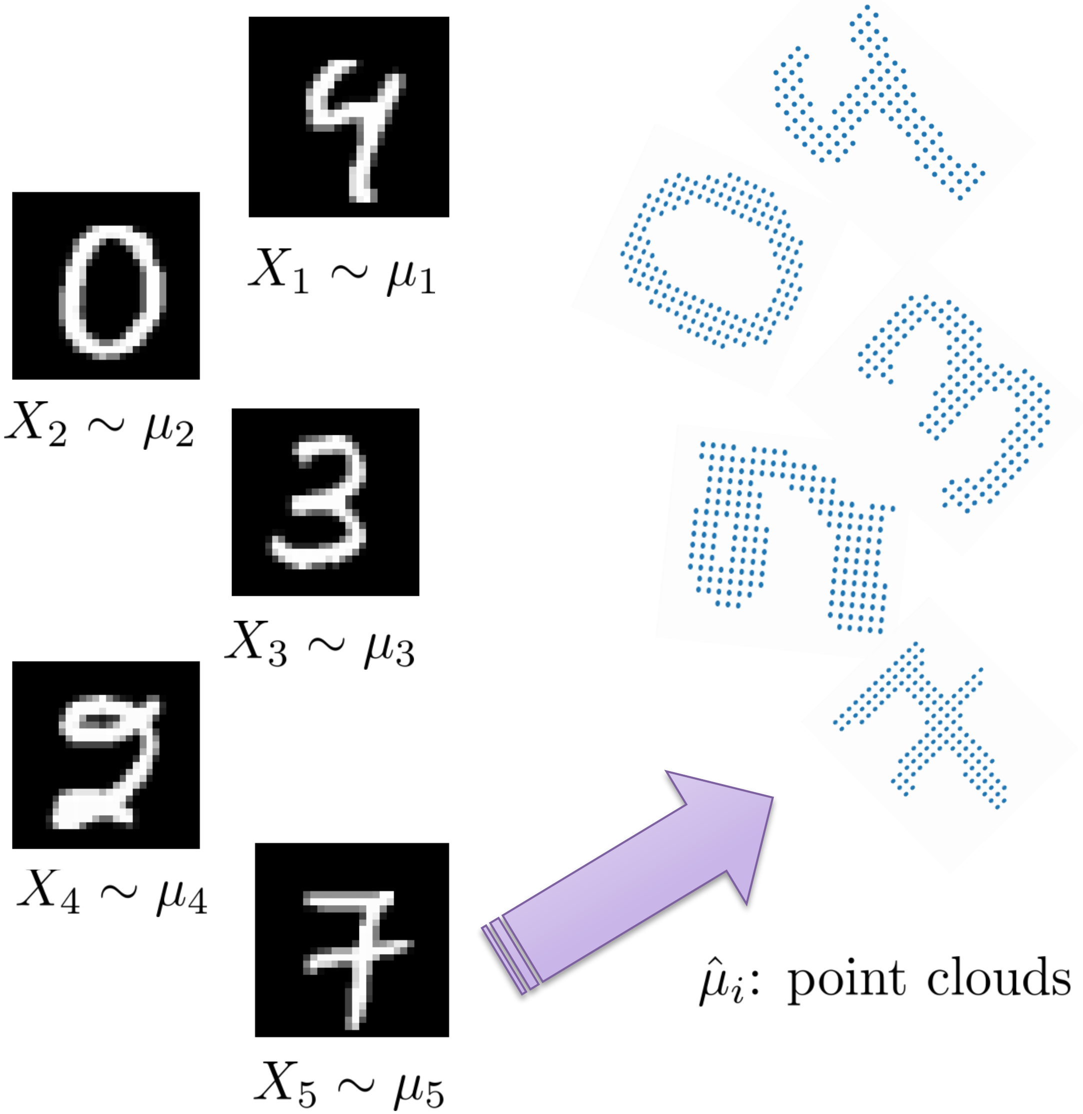
Outline

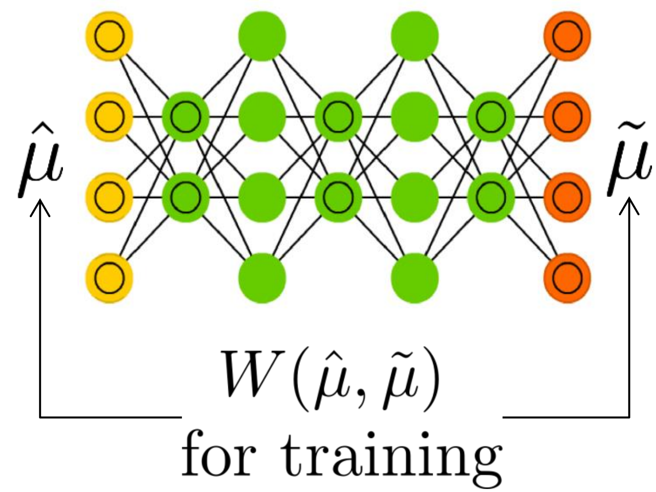
I. Proposed layers

II. Theoretical results

III. Applications

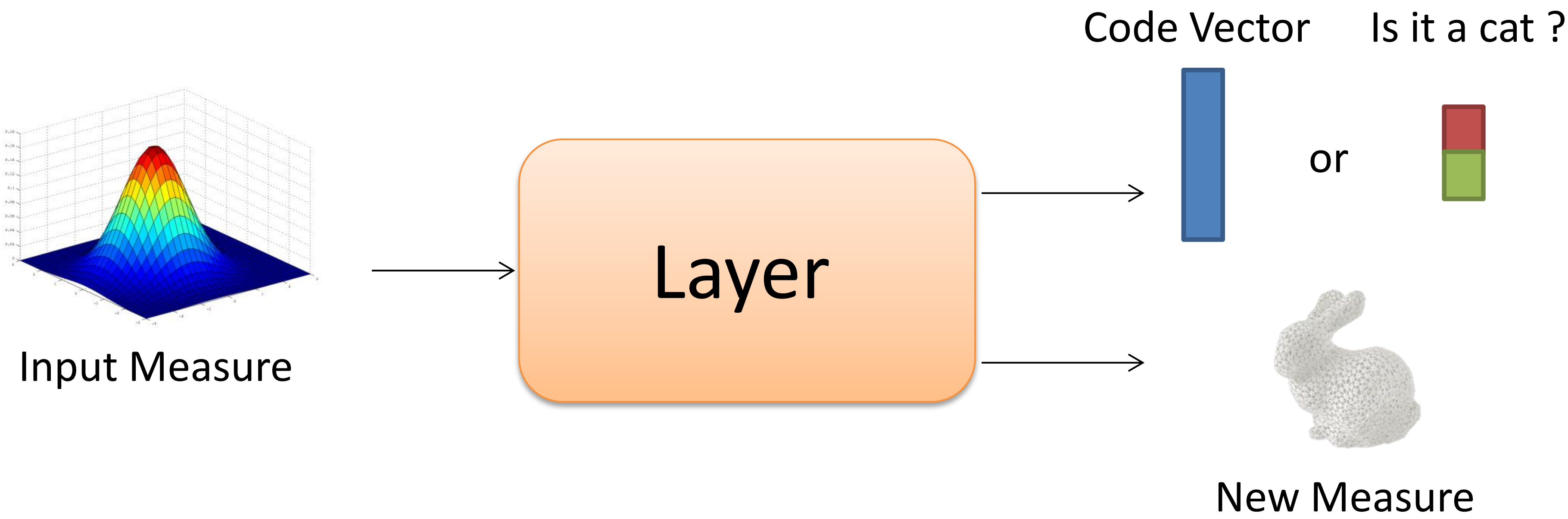
General Formalism



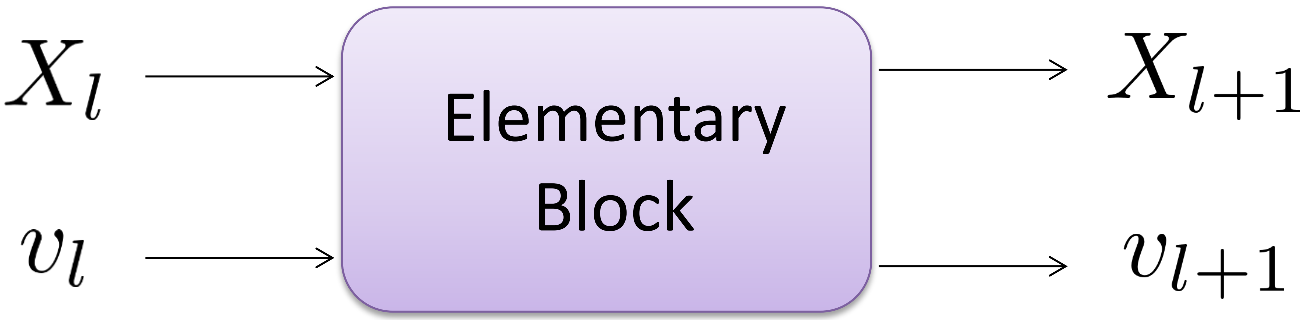


weak-$*$ topology
for maths

Aim



Proposed Layer



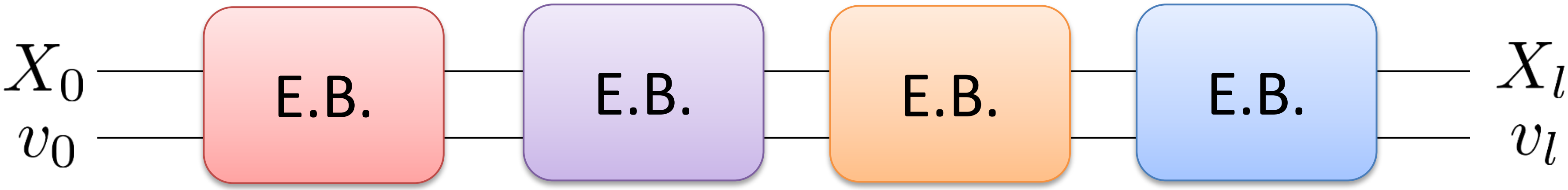
where

- $X_{l+1} = f_l(X_l, v_l)$
- $v_{l+1} = \mathbb{E}(g_l(X_l, v_l))$

Fully connected case:

- $f_l(x, u) = (\lambda_l(y_i))_i$, where
- $y = A_l[x, u]^\intercal + b_l$
- and λ_l: pointwise non-linearity

Proposed Architectures



Tasks		
Discriminative	**Generative**	**Dynamic**
Set $v_0 = 0$	Set X_0: noise	Set $v_0 = 0$
Discard X_l	Discard v_l	Discard v_l

Main Building Blocks

Push-forward.

Modification of support while maintaining geometry

Integration.

Agglomerate information and enforce permutation invariance/equivariance

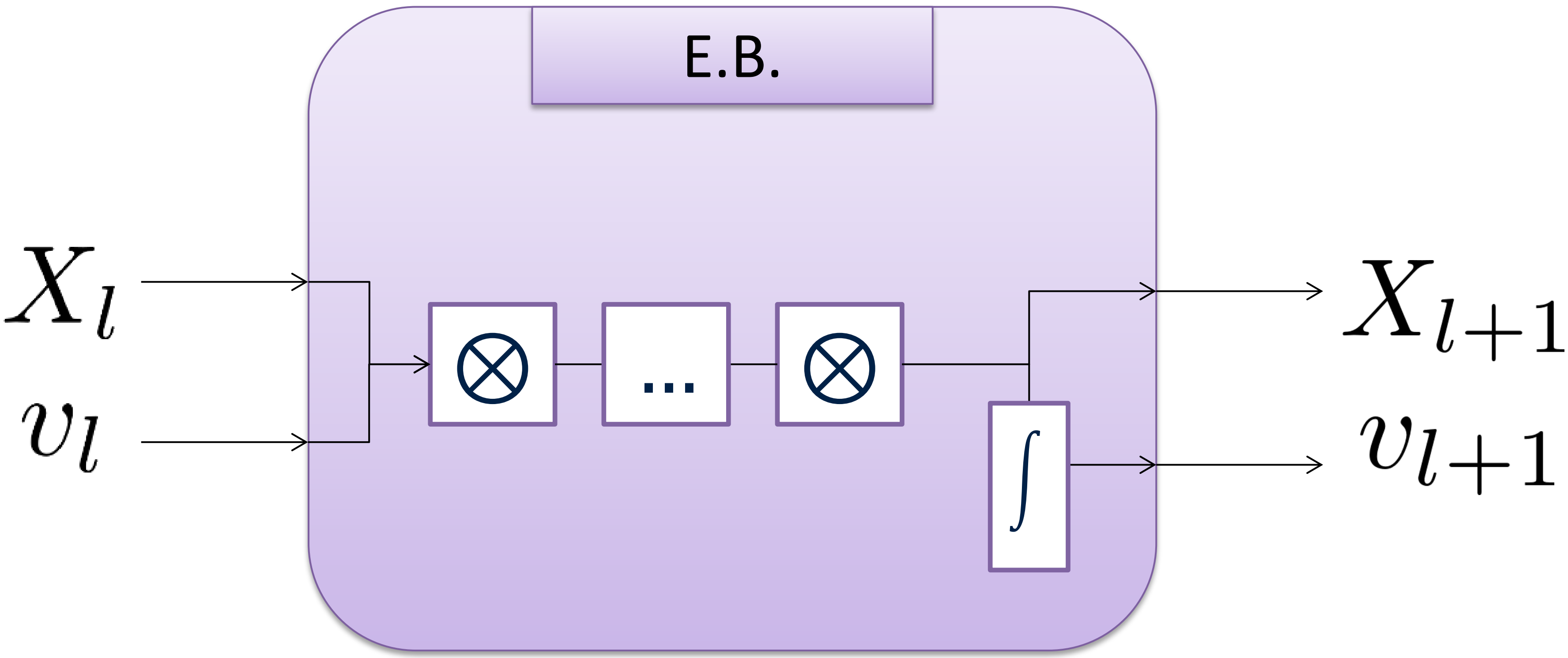
Modulation.

Mass modification of support (mass destruction or creation)

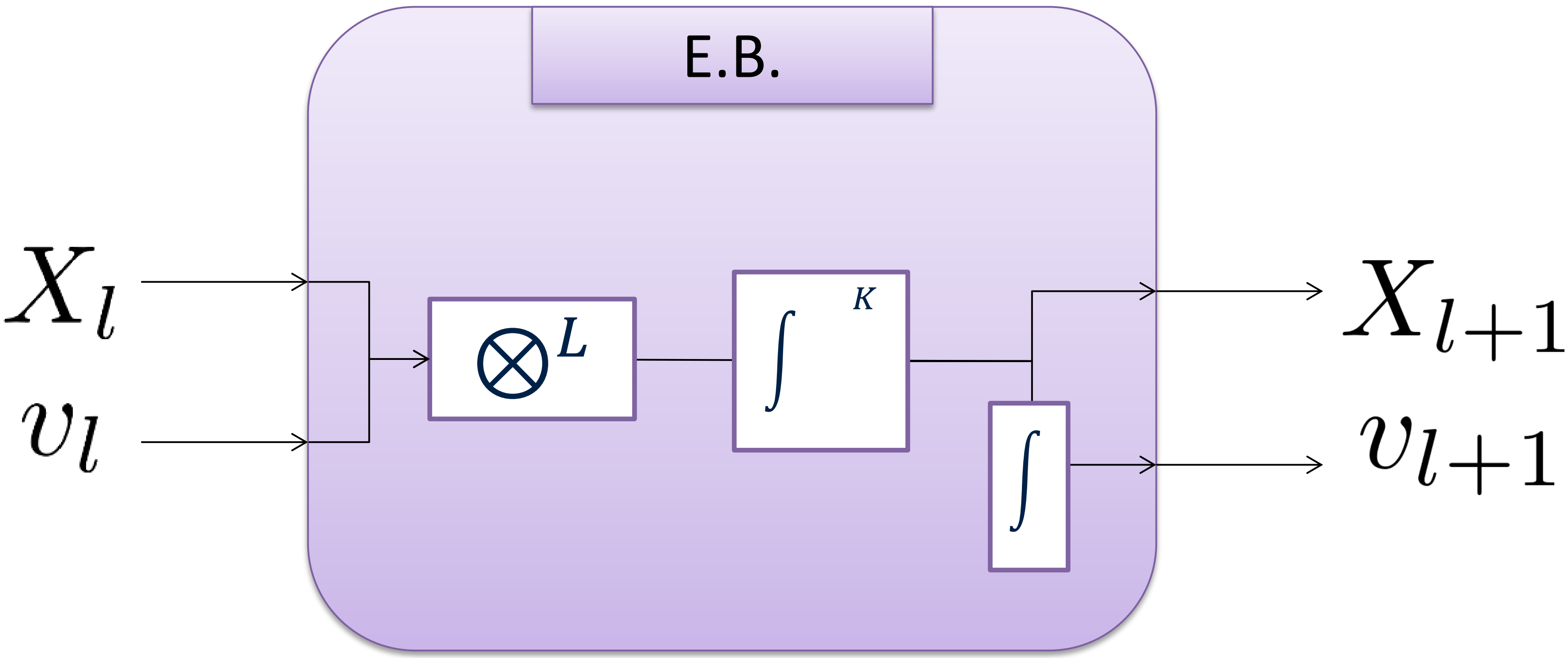
Tensorization.

Generates interactions between points.

Combining Building Blocks



Combining Building Blocks



Outline

I. Proposed layers

II. Theoretical results

III. Applications

Building Blocks Properties

Proposition. For $f : \mathcal{X} \to \mathcal{Y}$

$$W_{1,\mathcal{Y}}(f_{\#}\mu, f_{\#}\nu) \leq \mathrm{Lip}(f) W_{1,\mathcal{X}}(\mu, \nu)$$

$$W_{1,\mathcal{Y}}(f_{\#}\mu, g_{\#}\mu) \leq ||f - g||_{L^1(\mu)}$$

Proposition. For $f \in \mathcal{C}(\mathcal{Z} \times \mathcal{X}, \mathbb{R}^d)$, a fixed probability measure ζ, $f[\cdot, \mu] \triangleq \int_{\mathcal{X}} f(\cdot, x) \mathrm{d}\mu(x) : \mathcal{Z} \to \mathbb{R}^d$

$$||f[\cdot, \mu] - f[\cdot, \nu]||_{L^1(\zeta)} \leq d \cdot \mathrm{Lip}(f) \cdot W_{1,\mathcal{X}}(\mu, \nu)$$

Proposition.

$$W_{1,\mathcal{X}\times\mathcal{Y}}(\mu \otimes \nu, \mu' \otimes \nu') \leq W_{1,\mathcal{X}}(\mu, \mu') + W_{1,\mathcal{X}}(\nu, \nu')$$

Approximation Property

Theorem (Approximation property).

Let Ω a compact subset of $\mathbb{R}^d$ and $f : \mathcal{M}(\Omega) \to \mathbb{R}$ weak-$*$ continuous. $\forall \epsilon > 0$, there exists

- matrices A_1, A_2
- vectors b_1, b_2
- pointwise non-linearities ϕ_1, ϕ_2

s.t.

$$\forall \mu \in \mathcal{M}(\Omega), |f(\mu) - \phi_2 \left(A_2 \mathbb{E}_{X \sim \mu}(\phi_1(A_1 X + b_1)) + b_2\right)| < \epsilon$$

Outline

I. Proposed layers

II. Theoretical results

III. Applications

Numerical procedure

Classification.

Cross-Entropy loss:

$$\mathrm{loss}(x, \mathrm{class}) = -\log\left(\frac{\exp(x[\mathrm{class}])}{\sum_j \exp(x[j])}\right)$$

or its weighted version

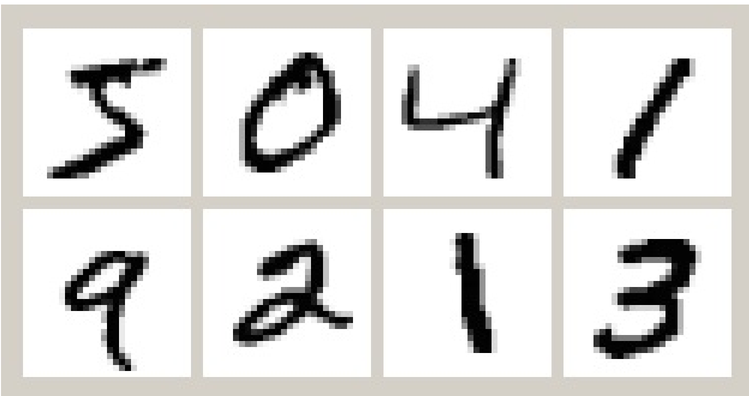
Generation/Dynamic Networks.

Sinkhorn Divergence loss:

$$\mathrm{loss}(\mu, \tilde{\mu}) = W(\mu, \tilde{\mu}) - \frac{1}{2}W(\mu, \mu) - \frac{1}{2}W(\tilde{\mu}, \tilde{\mu})$$

Loss Computation → Backpropagation Autodiff + Stochastic Gradient Descent

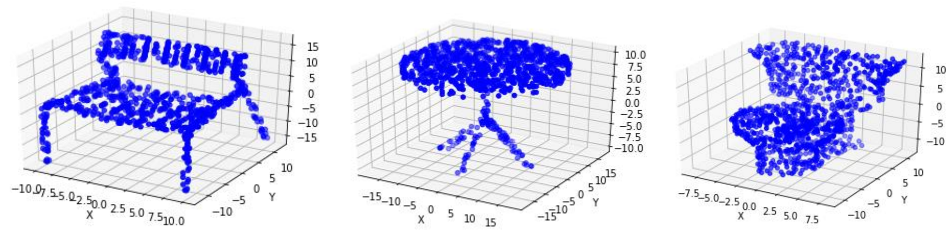
Applications in classification



MNIST – Digits
10 classes, 100 points

2 Elementary Blocks + 3 f.c. layers

Accuracy: 97.5%

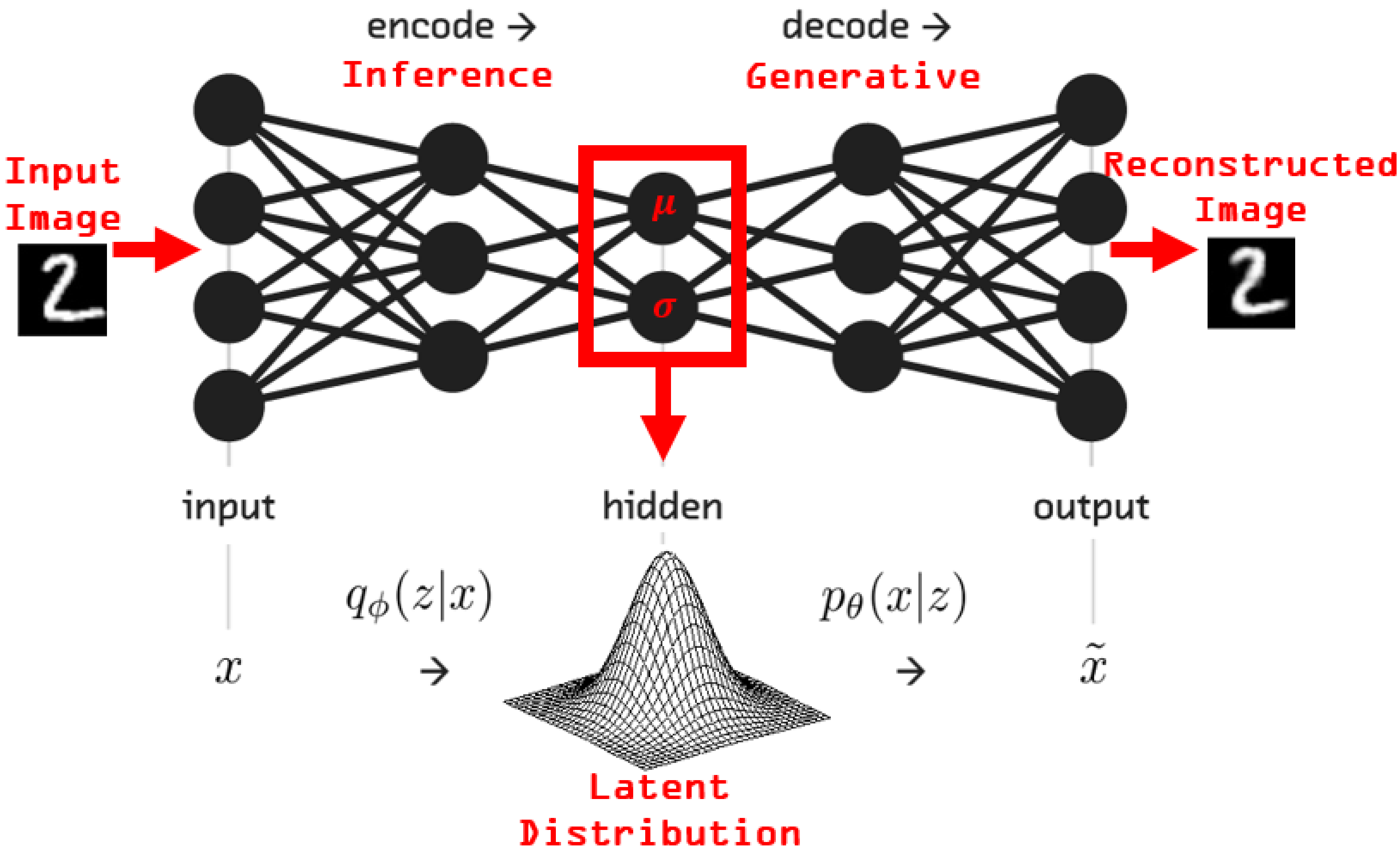


ModelNet10 – 3D shapes
10 classes, 1024 points

2 Elementary Blocks + 3 f.c. layers

Accuracy: 91.2%

Generative networks



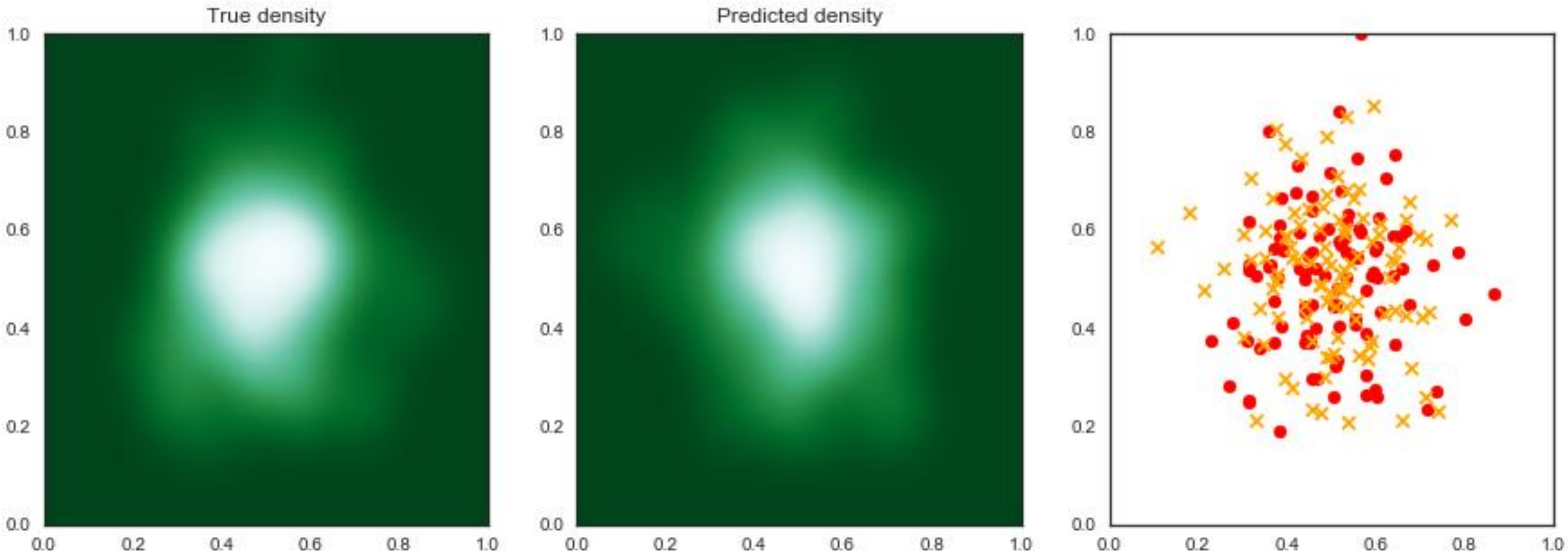
Generative networks

Some examples generated
With 2 elementary blocks + 3 f.c. layers

Dynamic networks

Prediction of positions at t=2
of particles following a Cucker-Smale
flocking model

3 Elementary Blocks



Perspectives

- Dynamics prediction on a real dataset
- Understand roles of each block
- Investigate rotation/translation equivariance
- Further theoretical results

Main References

- Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: ***Deep learning on point sets for 3d classication and segmentation.*** arXiv preprint arXiv:1612.00593, 2016.
- Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. ***Pointnet++: Deep hierarchical feature learning on point sets in a metric space***. Advances in Neural Information Processing Systems, pages 5105-5114, 2017.
- Tatsunori Hashimoto, David Giord, and Tommi Jaakkola. ***Learning population-level diffusions with generative rnns***. International Conference on Machine Learning, pages 2417-2426, 2016.
- Nicholas Guttenberg, Nathaniel Virgo, Olaf Witkowski, Hidetoshi Aoki, and Ryota Kanai. ***Permutation-equivariant neural networks applied to dynamics prediction***. arXiv preprint arXiv:1612.04530, 2016.
- Siamak Ravanbakhsh, Je Schneider, and Barnabas Poczos. ***Deep learning with sets and point clouds.*** arXiv preprint arXiv:1611.04500, 2016.
- Aude Genevay, Gabriel Peyre, and Marco Cuturi. ***Learning generative models with sinkhorn divergences.*** International Conference on Articial Intelligence and Statistics, pages 1608-1617, 2018.