Deep Generative Models for Knowledge Transfer

Evgeny Burnaev

Head of ADASE group

Skoltech

Skoltech
Skolkovo Institute of Science and Technology

Manifold Learning

Manifold learning – Data Analysis technology based on **geometrical model** about high-dimensional data [1]

A. The world is multidimensional

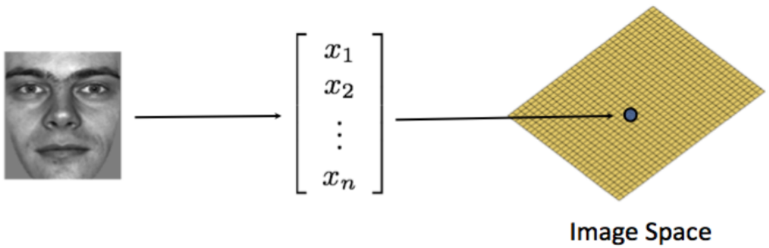
B. Multidimensional data are difficult to use

C. Real-world data have low-dimensional structure

D. The world is not flat (nonlinear)

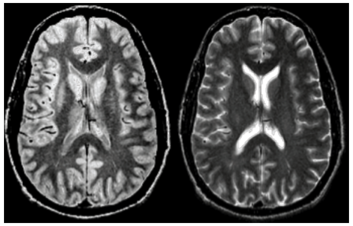
A. The world is multidimensional

Real-world data



1024×1024: $d \approx 10^6$

64×256: $d = 16\ 384$



fMRI: $d \approx 1.4 \times 10^6$/sec

B. Multidimensional data are difficult to analyze

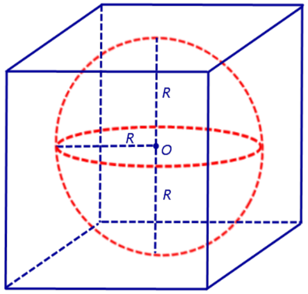
1) **Regression:** (Ibragimov, Khasminskii (1979); Stone (1982); etc.)

If $\mathbf{F} = \{\psi : [0,1]^d \to \mathbb{R}^1,\ \psi \text{ is Lipschitz}\}$

then for any estimator $\hat{\psi}$ of any kind from n known measurements $\{(x_i, \psi(x_i))\}$:

$$\sup_{\psi \in \mathbf{F}} \mathbb{E}\left(\psi(x) - \hat{\psi}(x)\right)^2 \geq \text{Const} \times n^{-2/(2+d)}$$

The lower bound is nonasymptotic!



2) MSE in case of KDE ~ $O(n^{-4/(d+4)})$

3) Empty space phenomenon, curse of dimensionality

C. Low-dimensional structure helps!!!

- Data from ‘**natural**’ sources occupy usually a **small part** $\mathbf{X}$ in the ‘observation space’ $\mathbb{R}^d$

- $\mathbf{X}$ has **small ‘intrinsic dimension’** $s < d$

- Data can be described by a small number s of parameters (features)

$d \approx 10^6$

$s = 84$

$s = 40$

Low-dimensional structure of real-world data

- **to find a low-dimensional structure of Data space**

  ✓ to estimate an Intrinsic dimension s of $\mathbf{X} \subset \mathbb{R}^d$

  ✓ to construct a s-dimensional features $z = h(x)$ describing $x \in \mathbf{X}$

- **to use extracted low-dimensional structure** to solve **specific Data analysis tasks**

Principal Component Analysis

Face-vector $x \in \mathbb{R}^{2061}$

$\approx$

$= \sum$

$(z_1 \times \mathbf{e}_1) + (z_2 \times \mathbf{e}_2) + (z_3 \times \mathbf{e}_3) + (z_4 \times \mathbf{e}_4) + (z_5 \times \mathbf{e}_5) + (z_6 \times \mathbf{e}_6) + (z_7 \times \mathbf{e}_7) + (z_8 \times \mathbf{e}_8)$

x_{PCA} = projection on L_{PCA} defined by features $z = (z_1, z_2, \ldots, z_s)$

$$x(face) \leftrightarrow z = (z_1, z_2, \ldots, z_s) \in \mathbf{R}^s$$

$$L_{PCA,s}(faces) = \{mean\,face + \sum_{i=1}^{s} EigenFace_i \times z_i\}$$

Skoltech

Skolkovo Institute of Science and Technology

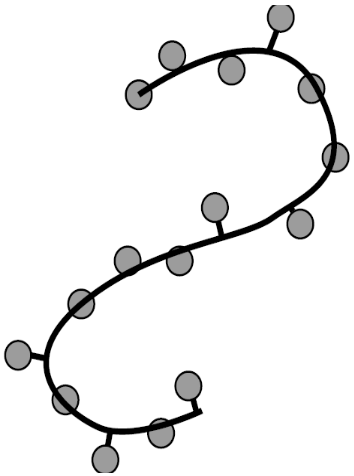
Principal Component Analysis (cont.)

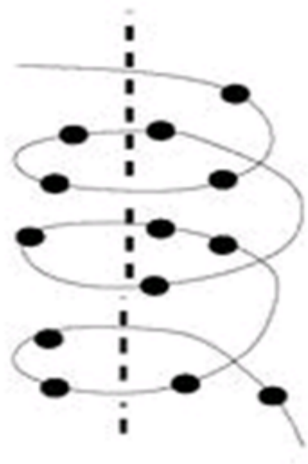
Original face described by 10^6-dimensional vector

Left to right: the same face described by ***s*** reduced features

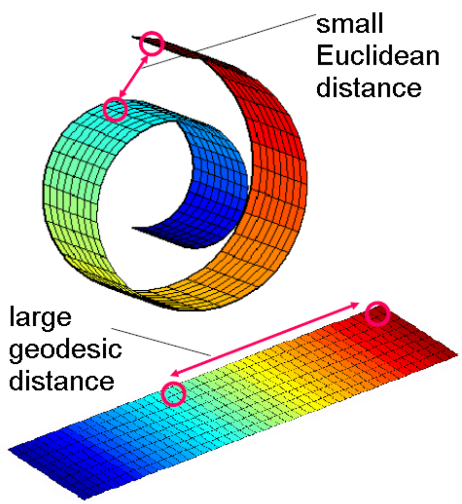
s **= 84** ***s*** **= 40** ***s*** **= 20** ***s*** **= 3** ***s*** **= 2** ***s*** **= 1**

D. The world is not flat





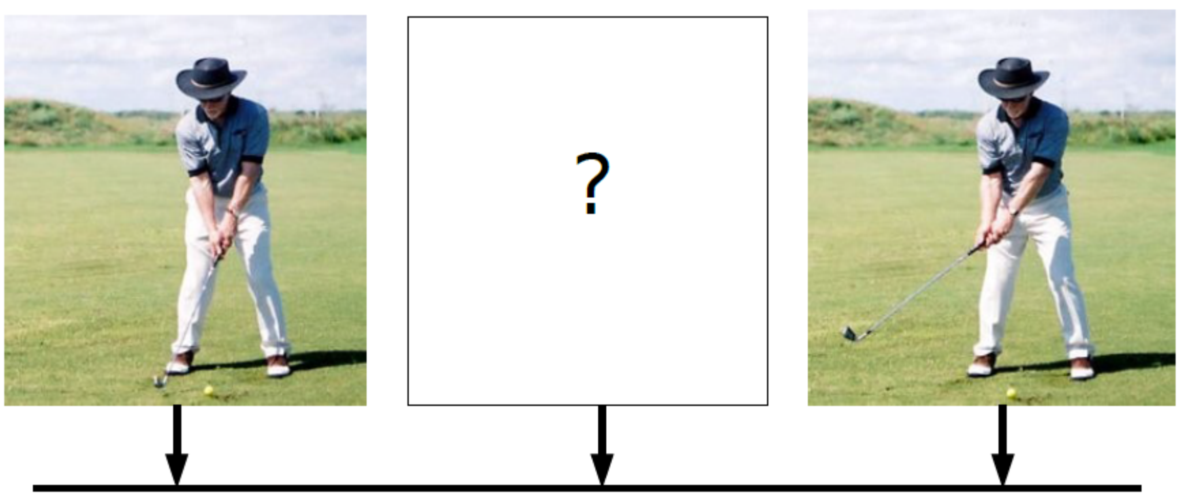
- **Linear methods like PCA do not work**



- **Math: Multivariate Statistical analysis consider mainly ‘linear methods’**

The world is not flat (cont.)

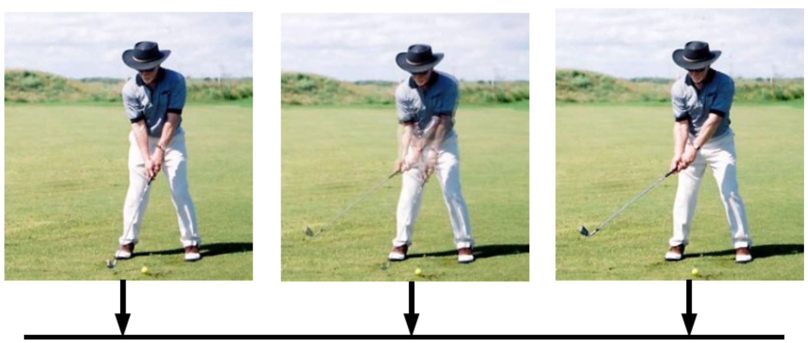
Video stream



Frame rate conversion based on inter-frame interpolation

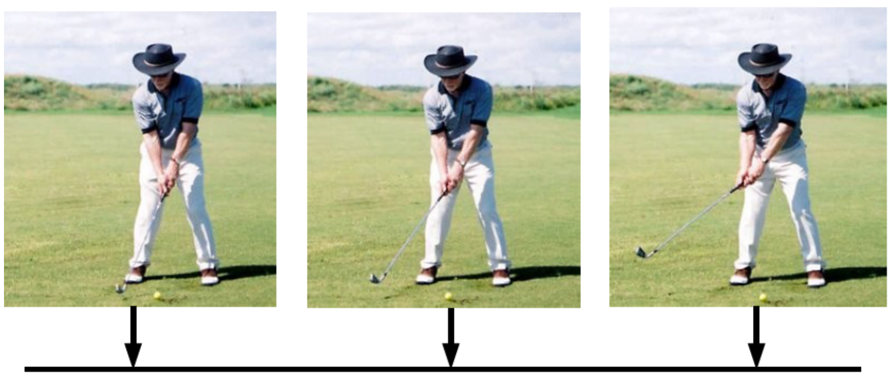
The world is not flat (cont)

'Linear' inter-frame interpolation



The world is not flat (cont.)

‘Nonlinear’ inter-frame interpolation



Manifold model: mobile robot navigation [2]

64×256 pixels: d = 16384

Robot localization θ = **(2D Coordinates, Orientation)** $\in \mathbb{R}^3$

$x = \varphi(\theta) \in \mathbb{R}^d$ - captured image at Robot localization θ

Appearance space $\mathbf{M} = \{x = \varphi(\theta),\ \theta \in \Theta\}$

consisting of images which may be captured

under **all possible localizations** $\theta \in \Theta \subset \mathbb{R}^3$ is 3D-surface (**Appearance manifold**) in $\mathbb{R}^d$

Skoltech

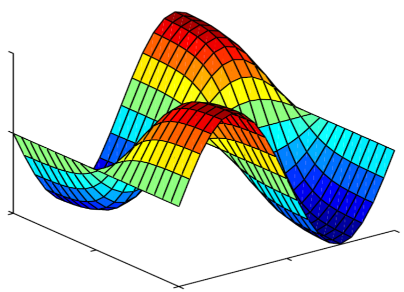
Skolkovo Institute of Science and Technology

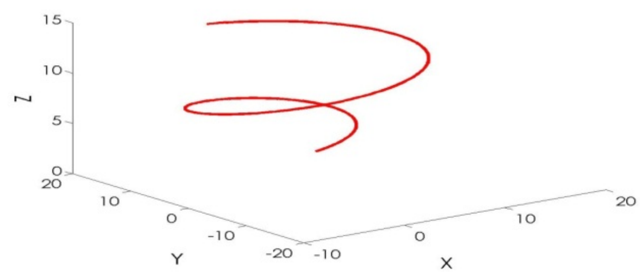
Manifold covered by single chart (surface in $\mathbb{R}^d$)

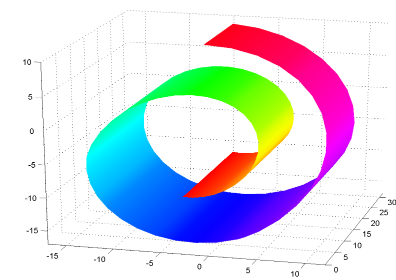
$$\mathbf{M} = \{x = g(z) \in \mathbb{R}^d : \; z \in \mathbf{Z} \subset \mathbb{R}^s\}$$

well-behaved **unknown** s -dimensional surface - **Data manifold**

covered by **single chart** g defined on **Coordinate space** $\mathbf{Z} \subset \mathbb{R}^s$
and embedded in ambient d-dimensional space, $s < d$







$h = g^{-1} : \mathbf{M} \rightarrow \mathbf{B}$ - inverse mapping – a parameterization $z = h(x)$ on the Data manifold

Statistical analysis of manifold valued data

Let μ be some **unknown** probability measure on **unknown** s-dimensional manifold $\mathbf{M} = \mathrm{supp}(\mu)$ with **unknown** value of s

Based on given sample of independent observations

$$D_n = \{x_1, x_2, \ldots, x_n\} \subset \mathbf{M}$$

solve various statistical problems such as:

- to estimate intrinsic dimension s
- to estimate low-dimensional parameterization h on the manifold $\mathbf{M}$
- to estimate the manifold $\mathbf{M}$
- to estimate tangent space $L(x)$ to the manifold $\mathbf{M}$ at point x
- to estimate a density $f(x) = \dfrac{d\mu}{dm}$, etc.

Skoltech
Skolkovo Institute of Science and Technology

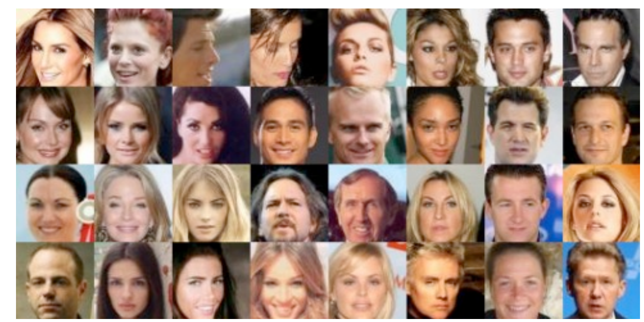
Manifold Learning via Deep Generative Model [3]

Probabilistic model for Data on manifolds

$z \sim p(z)$

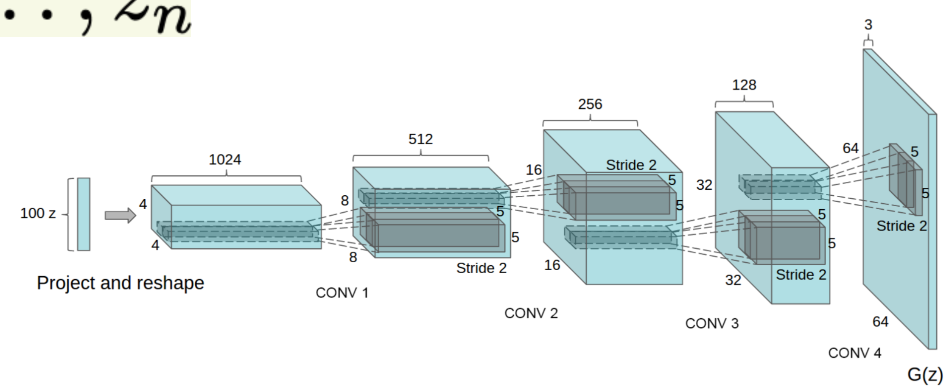
$x \sim p(x|g_\theta(z)) \cdot p(z)$

g_θ



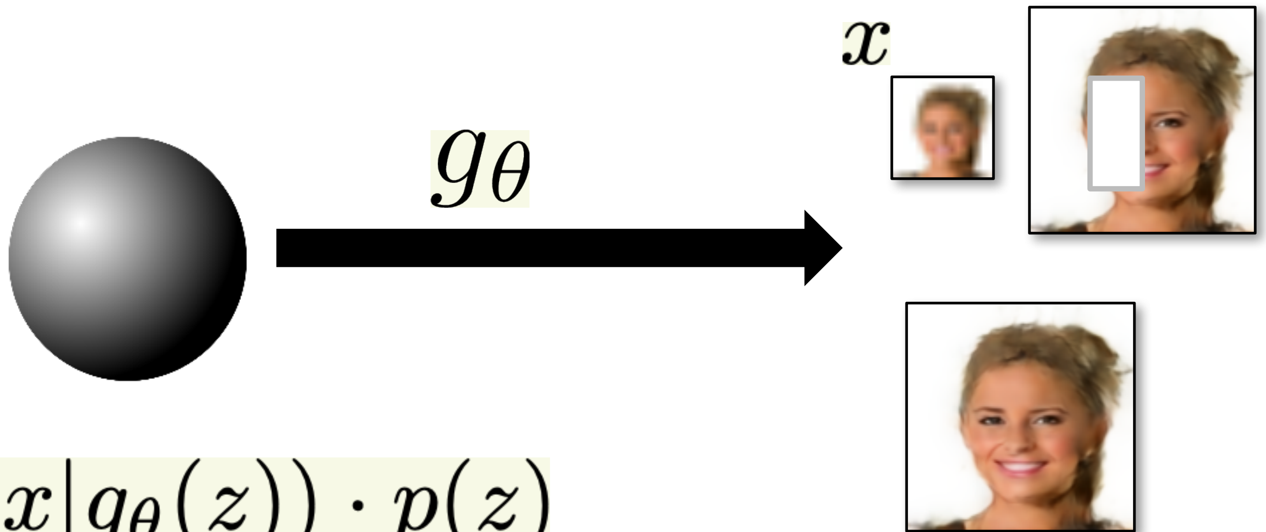
$z_1, z_2, \dots, z_n$

$x_1, x_2, \dots, x_n$



Why do I *really* need a latent model?

Answer: image restoration/editing/enhancement



$$x \sim p(x|g_\theta(z)) \cdot p(z)$$

$$\hat{z} = \arg\max_z \left[\log p(x|g_\theta(z)) + \log p(z)\right]$$

$$\hat{x} = g_\theta(\hat{z})$$

Skoltech
Skolkovo Institute of Science and Technology

Variational AutoEncoders [4]

Data likelihood: $p_\theta(x) = \int p_\theta(z) p_\theta(x|z) dz$

Variational AutoEncoders [4]

Data likelihood: $$p_\theta(x) = \int p_\theta(z) p_\theta(x|z) dz$$

Simple Gaussian prior

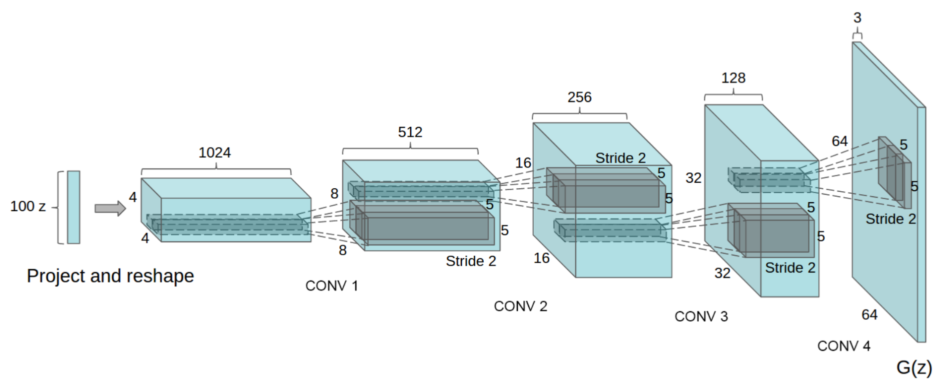
Variational AutoEncoders [4]

Data likelihood: $$p_\theta(x) = \int p_\theta(z) p_\theta(x|z) dz$$

Decoder neural network

Decoder neural network:

$$p_\theta(x|z) = \mathcal{N}(x|\mu_\theta(z), \sigma^2_\theta(z) \cdot \mathrm{I})$$



Variational AutoEncoders [4]

Posterior density is intractable:

$$p_\theta(z|x) = \overset{\checkmark}{p_\theta(x|z)}\,\overset{\checkmark}{p_\theta(z)}/\overset{\text{X}}{p_\theta(x)}$$

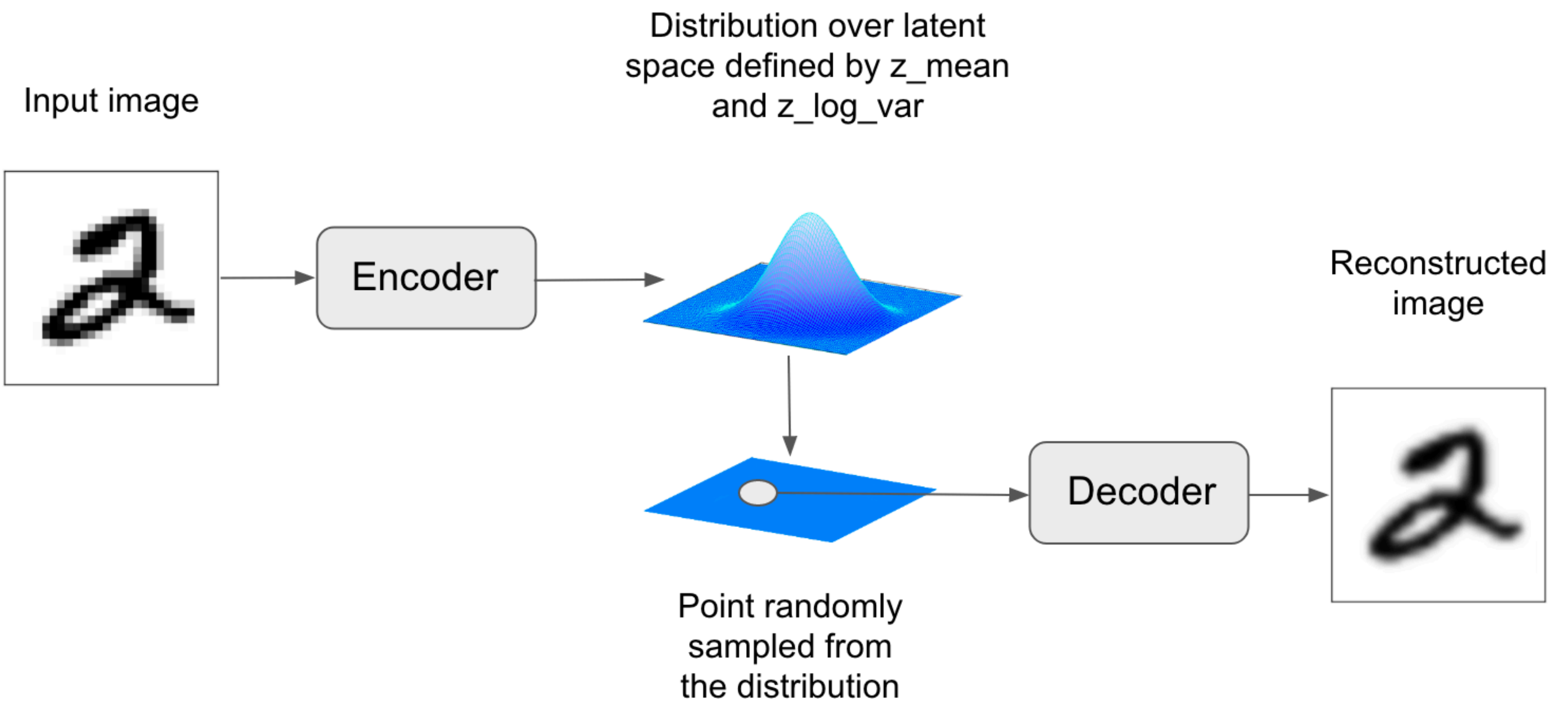
Solution: construct an encoder network $\boxed{q_\phi(z|x)}$

to approximate $p_\theta(z|x)$

Encoder neural network:

$$q_\phi(z|x) = \mathcal{N}(z|\mu_\phi(x), \sigma^2_\phi(x) \cdot \mathrm{I})$$

Variational AutoEncoders



Skoltech
Skolkovo Institute of Science and Technology

Variational AutoEncoders

Marginal log-likelihood:

$$\log p_\theta(x_i) =$$

$$= KL[q_\phi(z|x) || p_\theta(z|x)] + L(\theta, \phi, x_i)$$

always ≥ 0

Variational AutoEncoders

Marginal log-likelihood:

$$\log p_\theta(x_i) =$$

$$= KL[q_\phi(z|x)||p_\theta(z|x)] + L(\theta, \phi, x_i)$$

always ≥ 0

$$\geq L(\theta, \phi, x_i)$$

Variational lower-bound:

Reconstruction error

$$L(\theta, \phi, x_i) =$$

$$= -KL[q_\phi(z|x_i)||p(z)] + \mathbb{E}_{q_\phi(z|x_i)}[\log p_\theta(x_i|z)]$$

Regularization!

Latent representation

Variational AutoEncoders

ELBO:

$$\sum_{i=1}^{n} \log p_\theta(x_i) \geq$$

$$\geq \sum_{i=1}^{n} L(\theta, \phi, x_i) \to \max_{\theta,\phi}$$

Variational AutoEncoders

Empirical variational lower-bound:

$$p(z) = \mathcal{N}(z|0, \mathrm{I})$$

$$z_{i,l} \sim q_\phi(z|x_i)$$

Regularization!

$$\hat{L}(\theta, \phi, x_i) = \frac{1}{2}\sum_{j=1}^{d}[1 + \log \sigma_{j,\phi}^2(x_i) - \mu_{j,\phi}^2(x_i) - \sigma_{j,\phi}^2(x_i)]$$

$$+ \frac{1}{L}\sum_{l=1}^{L} \log p_\theta(x_i|z_{i,l}) \to \max_{\theta,\phi}$$

Reconstruction error!

Variational AutoEncoders

Use decoder network. Now sample z from prior!

Labeled Faces in the Wild

Degree of smile

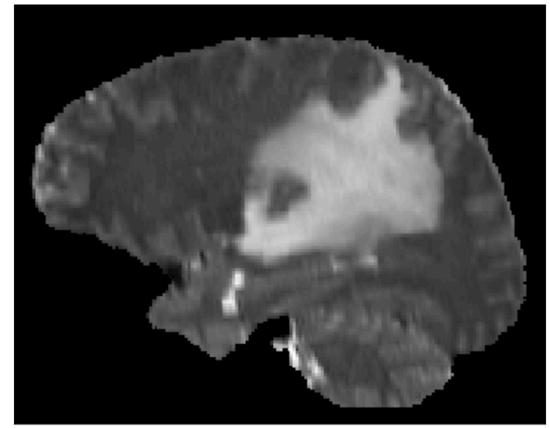
Vary $\mathbf{z_1}$

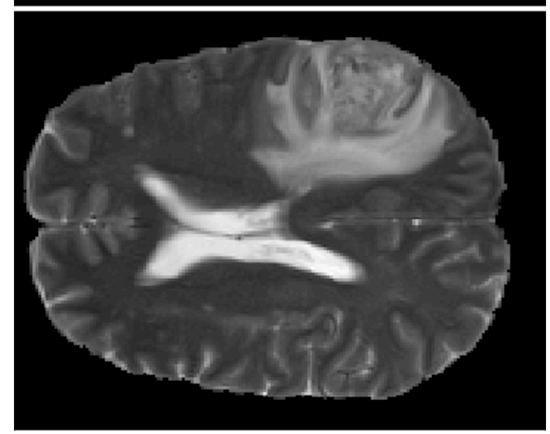
Vary $\mathbf{z_2}$

Head pose

Knowledge Transfer for Medical Imaging [5-7]

MRI – medical imaging technique used in radiology to form pictures of the body anatomy

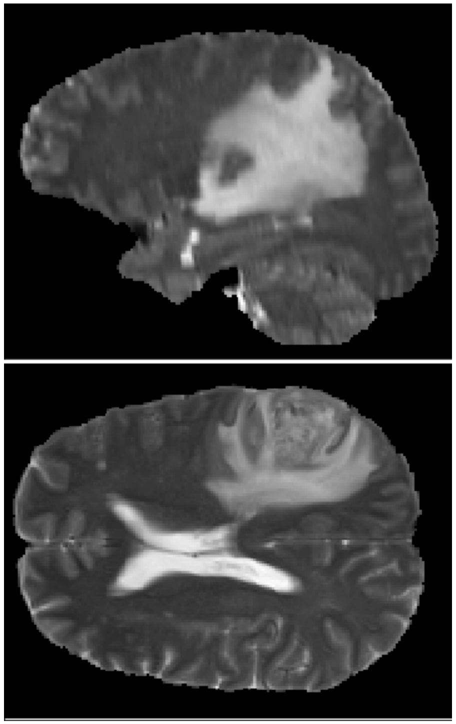


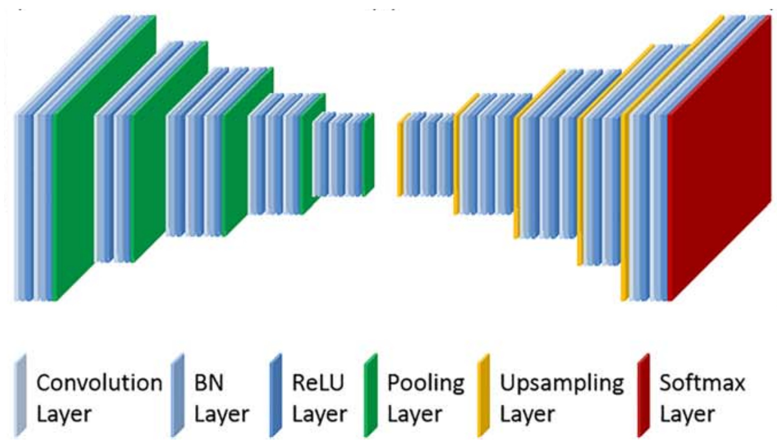


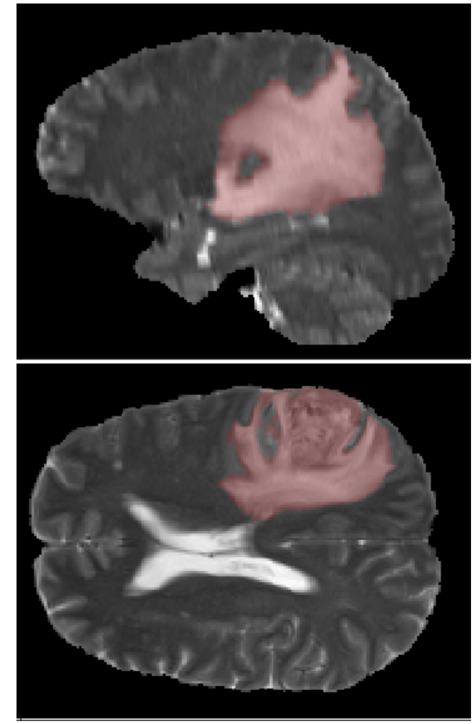
MRI semantic segmentation applications in medicine:

- Tumors (e.g. brain, liver) analysis and monitoring
- Multiple sclerosis plaques detection
- White matter hyperintesities detection

Challenges



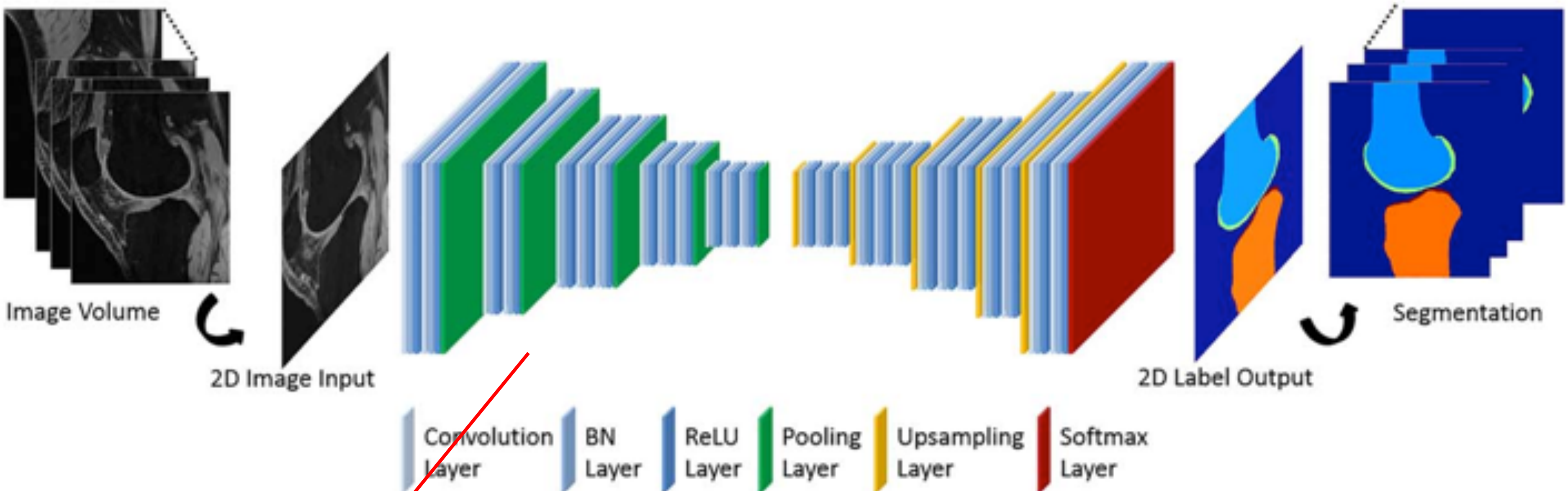


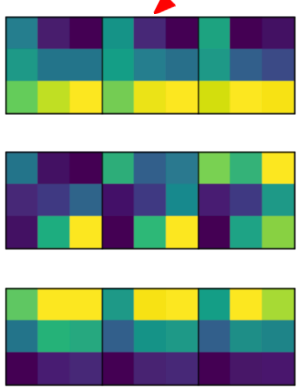


Scarse data:

- Expensive annotation
- Privacy concerns
- Bad performance of transfer learning due to disease specificity

U-Net model for Segmentation

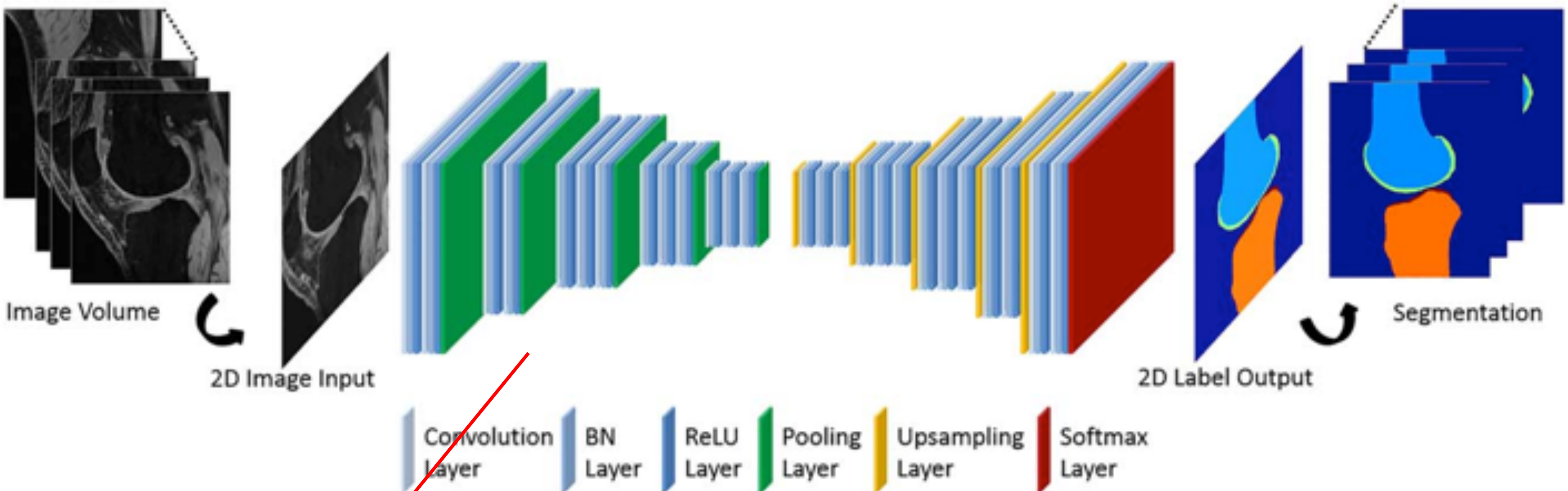


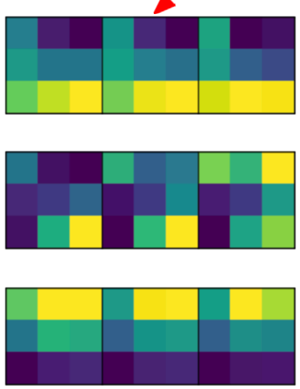


Manifold of Filters???

Skoltech
Skolkovo Institute of Science and Technology

U-Net model for Segmentation





Main Idea:

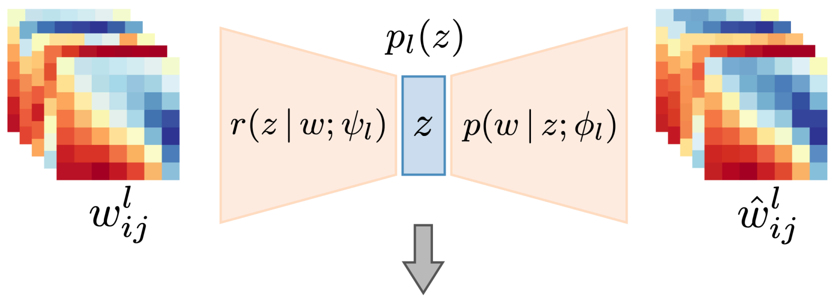
Use VAE model to learn implicit prior distribution over convolutional filters of each layer (DWP, [7])

Skoltech
Skolkovo Institute of Science and Technology

Deep Weight Prior [7]

Algorithm:

- ➢ Train network on the bootstrapped source dataset (*D₁*)
- ➢ Collect learned filters
- ➢ Train implicit prior distribution (VAE)



$$\hat{p}_l(w) = \int p(w \mid z; \phi_l) p_l(z)\, dz$$

- ➢ Use trained prior for variational inference on the target dataset (D_2)

$$\log p(y_i|x_i) \geq \mathbb{E}_{q_\theta(w)} \log p(y_i|x_i, w) - KL[q_\theta(w) \| \hat{p}(w)] \geq$$

$$\geq \mathbb{H}(q_\theta(w)) - \mathbb{E}_{q_\theta(w)}\{KL[r(z|w;\psi)\|p(z)] - \mathbb{E}_{r(z|w;\psi)} \log p(w|z;\phi)\}$$

$$\to \max_{\theta,\psi}$$

(from [7])

Experiments [5]

Datasets:

- 170 MRI of patients with multiple sclerosis (MS)
- 285 MRI of patients with brain tumor (BRATS18)

Task:

Binary semantic segmentation

Metrics:

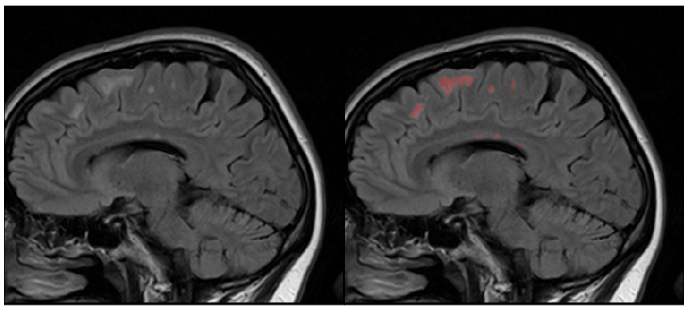
- Dice Similarity Coefficient

$$DSC = \frac{2TP}{2TP + FP + FN}$$

- Intersection Over Union

$$IOU = \frac{TP}{TP + FP + FN}$$

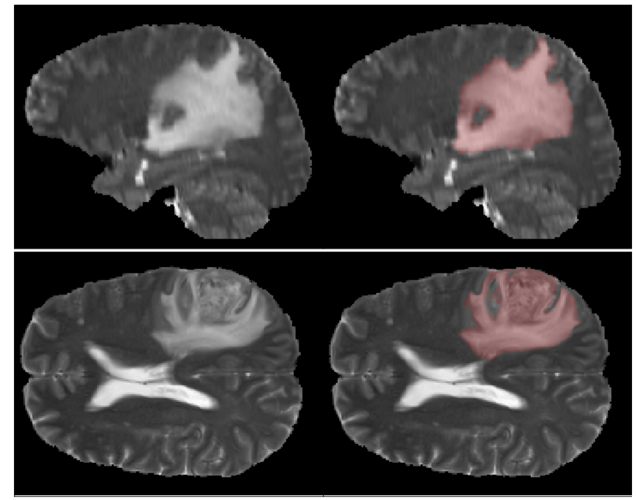
Experiments [5]



MS data: Multiple sclerosis

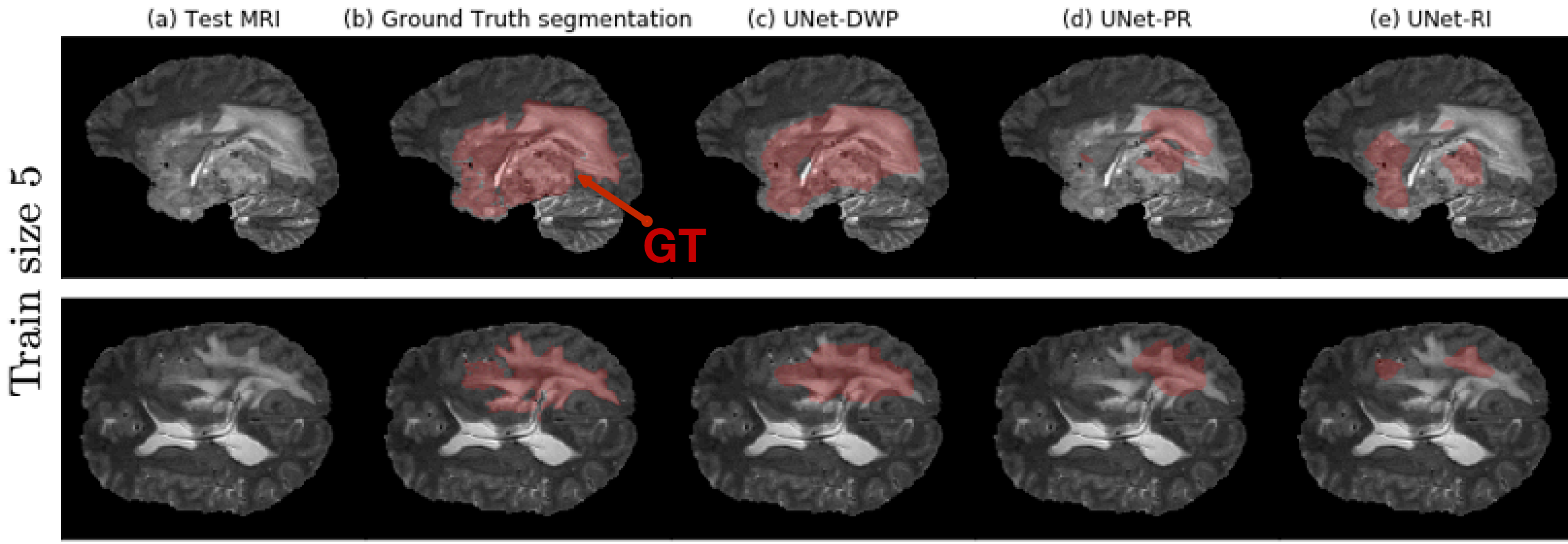
transfer

$\Rightarrow$



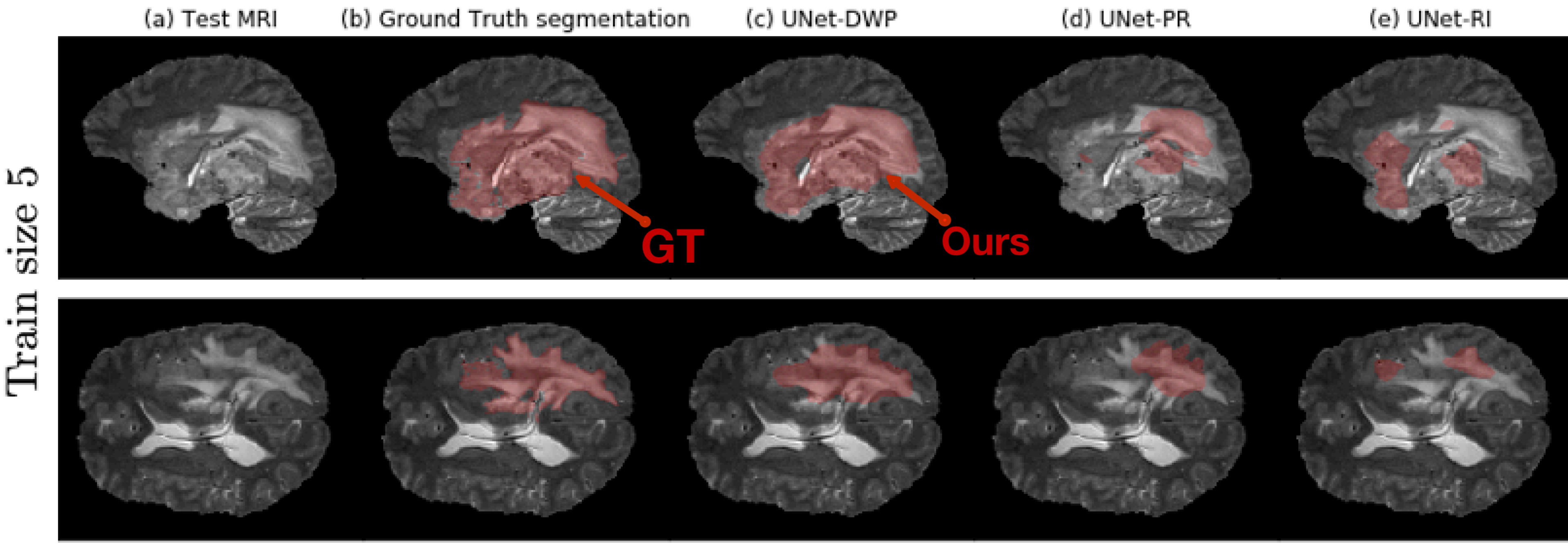
BRATS18 data: Brain tumors

Experiments [5]



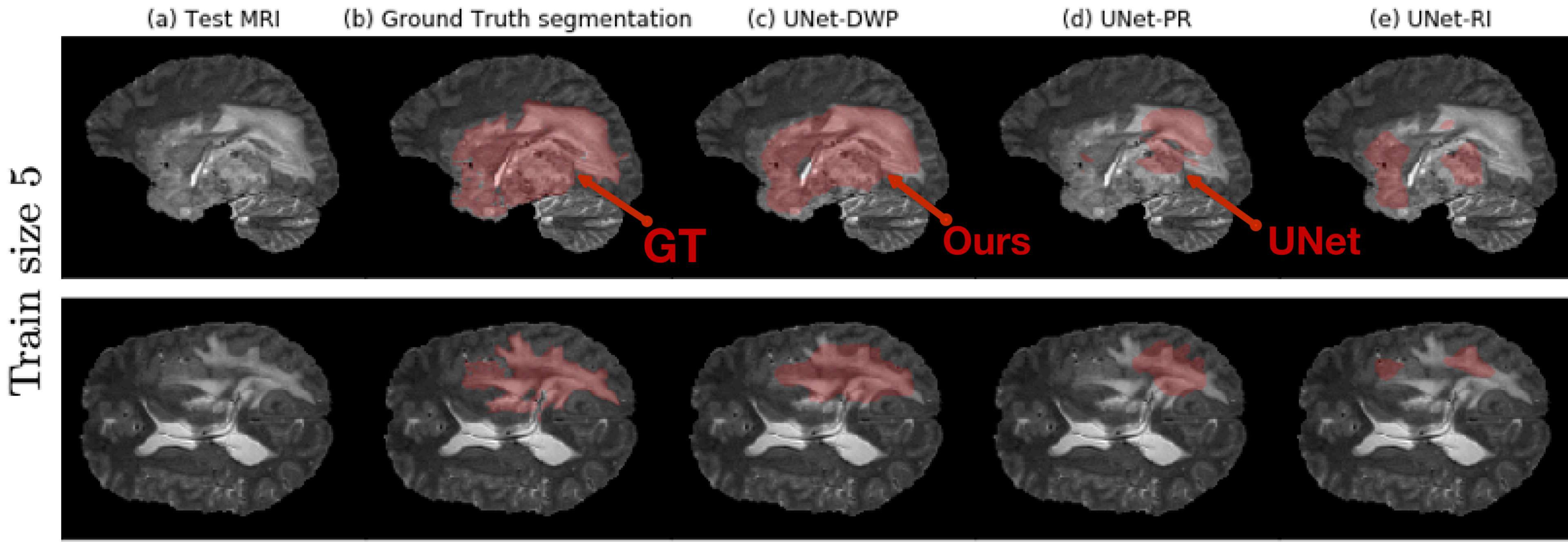
Train size: $N_1 = 170, \quad N_2 = 5$

Experiments [5]



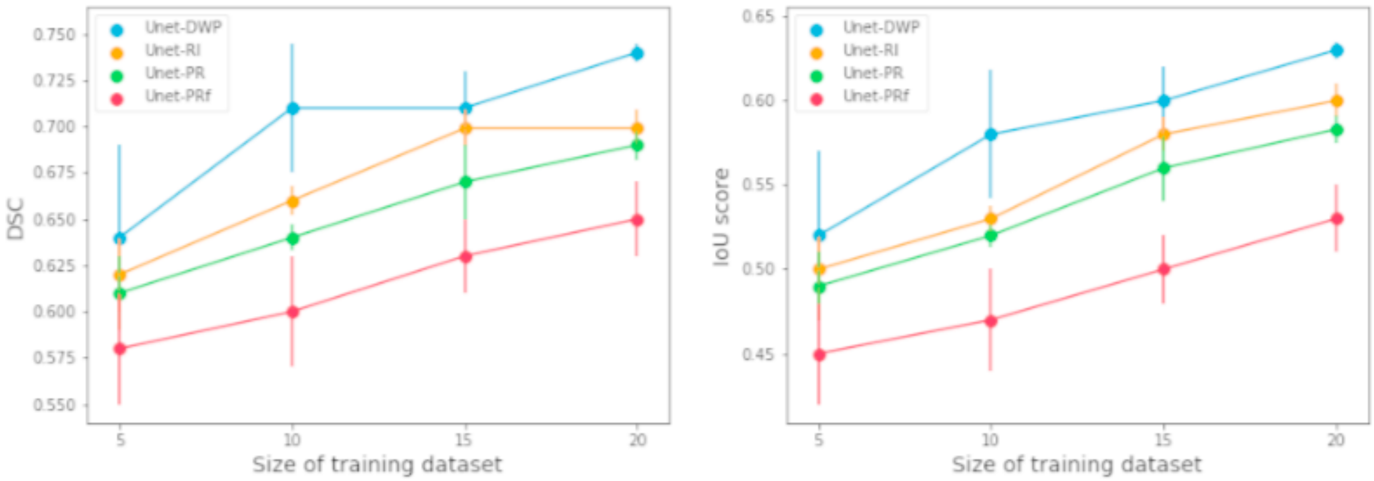
Train size: $N_1 = 170, \quad N_2 = 5$

Experiments [5]



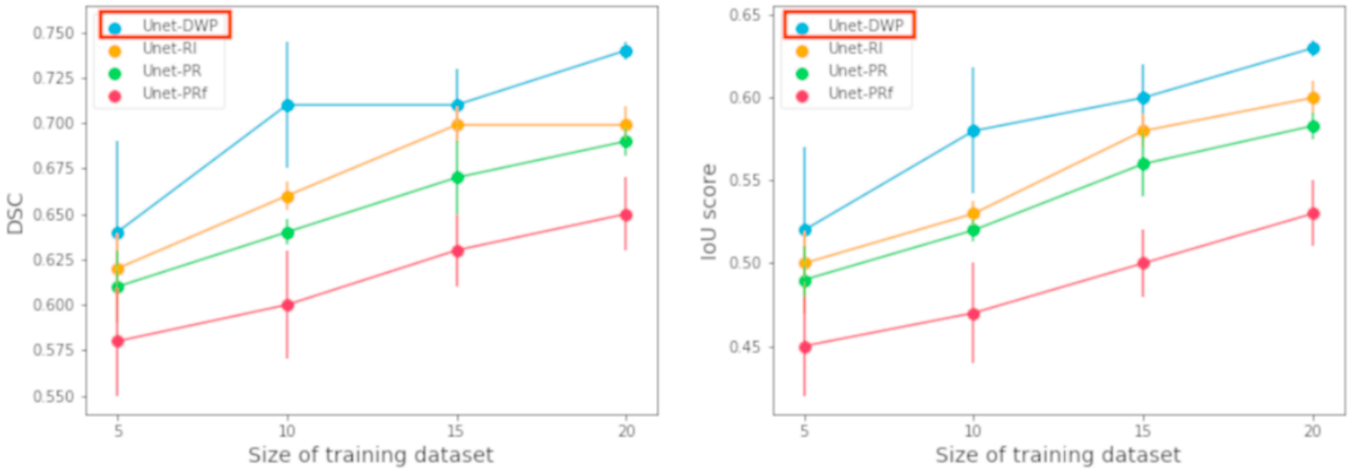
Train size: $N_1 = 170, \quad N_2 = 5$

Results



- **Unet-RI** (orange): without transfer learning
- **Unet-PR** (green): fine-tuning of the whole network
- **Unet-PRf** (red): fine-tuning of the input and output block

Results



- **Unet-RI** (orange): without transfer learning
- **Unet-PR** (green): fine-tuning of the whole network
- **Unet-PRf** (red): fine-tuning of the input and output block

Conclusion

Data Analysis under Manifold Assumption

- Manifold are everywhere
- VAEs and other Deep Generative Models are efficient for Manifold Parameterization and Estimation
- We can model a Prior distribution of parameters using Implicit Generative Models

Thanks for attention

Skoltech
Skolkovo Institute of Science and Technology

Some References

1. Yu. Ma, Yun. Fu. Manifold learning and applications, CRC Press 2011
2. Kuleshov A., Bernstein A., Burnaev E., Yanovich Yu. Machine Learning in Appearance-based Robot Self-localization, ICMLA, 2017
3. ShahRukh Athar, Evgeny Burnaev, Victor Lempitsky. Latent Convolutional Models. ICLR, 2019
4. Kingma and Welling, “Auto-Encoding Variational Bayes”, ICLR 2014
5. A. Kuzina, E. Egorov, E. Burnaev. Bayesian generative models for knowledge transfer in MRI semantic segmentation problems. Journal: Frontiers in Neuroscience, section Brain Imaging Methods, 2019
6. A. Kuzina, E. Egorov, E. Burnaev. BooVAE: A scalable framework for continual VAE learning under boosting approach arxiv.org/abs/1908.11853, 2019
7. Andrei Atanov, Arsenii Ashukha, Kirill Struminsky, Dmitry Vetrov, Max Welling, The Deep Weight Prior, ICLR, 2019
8. Kuleshov, Bernstein, Yanovich: Asymptotically optimal method in Manifold estimation, 2013
9. Bernstein, Kuleshov: Data-based Manifold Reconstruction via Tangent Bundle Manifold Learning, 2014
10. Alexander Kuleshov, Alexander Bernstein, Evgeny Burnaev. Conformal prediction in manifold learning. PMLR 91:234-253, 2018
11. A. Kuleshov, A. Bernstein and E. Burnaev. Kernel Regression on Manifold Valued Data, DSAA, 2018