ETH zürich

Using deep learning as priors in generative models for medical image computing

Ender Konukoglu

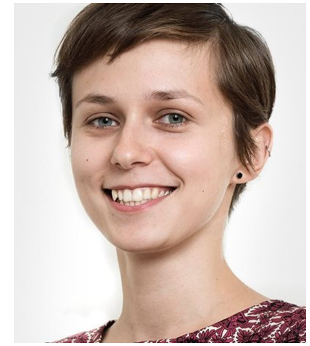
Computer Vision Laboratory, ETH Zürich

Acknowledgements

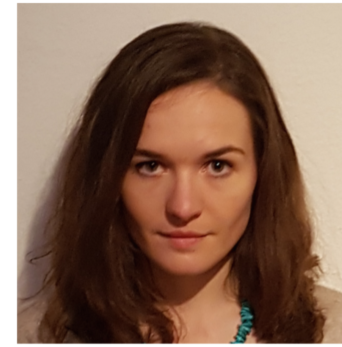
Kerem C. Tezcan

Christian F. Baumgartner

Xiaoran Chen



Katarina Tothova



Lisa M. Koch

FNSNF
SWISS NATIONAL SCIENCE FOUNDATION

SDSC

Platform for Advanced Scientific Computing

- Sebastian Kozerke
- Klaas Prüssmann

- Ben Glocker
- Nick Pawlowski

- Jean Augustinack
- David Salat
- Andre van der Kouwe
- Bruce Fischl

Outline

- Basics
- Examples without deep learning
- Unsupervised deep learning models
- Three ways to use DL models as priors

Generative modeling

$$\underbrace{p\left(\mathbf{y},\mathbf{m}\right)}_{\text{Joint distribution}} = \underbrace{p\left(\mathbf{y}|\mathbf{m}\right)}_{\text{Observation model}} \underbrace{p\left(\mathbf{m}\right)}_{\text{Prior}}$$

$\mathbf{y}$: Observation

$\mathbf{y} \in \mathbb{R}^N \quad \mathbf{y} \in \mathbb{C}^N$

$\mathbf{y} \in \{0, 1, \ldots, L\}^N$

- Image
- K-space data from MRI
- Initial segmentation
- Measurements

$\mathbf{m}$: Identity of interest

- Image
- Segmentation
- Class assignment
- Phenotypic information
- Demographic information

$p(\mathbf{m})$

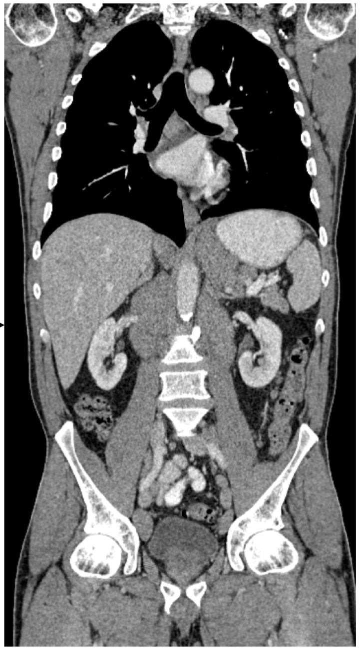
Initial belief what **m** can be

$p(\mathbf{y}|\mathbf{m})$

Generation process

Examples: Segmentation

$p(\mathbf{y}|\mathbf{m})$



$\mathbf{m}$: Segmentation

$p(\mathbf{m})$

$\mathbf{y}$: CT images

How do thorax and abdominal structures look like?

Generating an image from its segmentation, e.g. texture synthesis

Examples: Image enhancement

$p(\mathbf{y}|\mathbf{m})$

$\mathbf{m}$: High SNR MRI

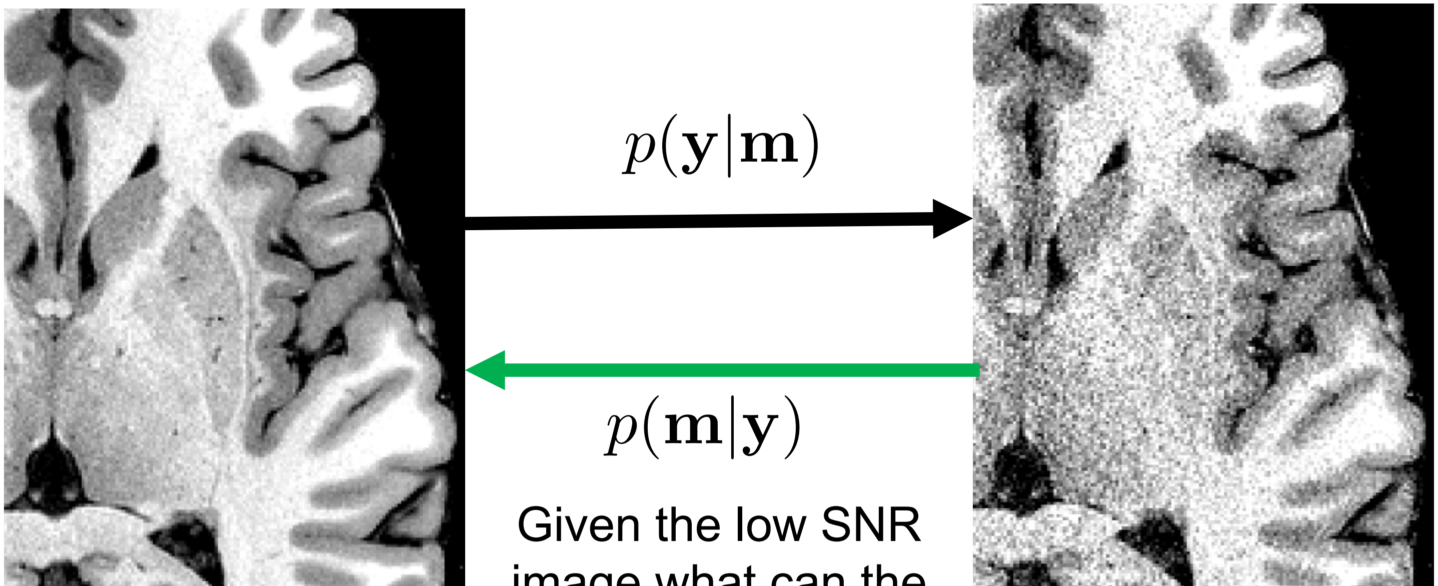
$p(\mathbf{m})$

How do high SNR MRI look like?

$\mathbf{y}$: Low SNR MRI

Generating a low SNR MRI from
a high high SNR one, e.g. adding noise

Posterior distribution



Given the low SNR image what can the high SNR image be?

Prediction is computed as posterior distribution in generative modeling

$$p\left(\mathbf{m}|\mathbf{y}\right) = \frac{p\left(\mathbf{y}|\mathbf{m}\right)p(\mathbf{m})}{p(\mathbf{y})}$$

$p(\mathbf{m}|\mathbf{y})$ Discriminative models directly estimate the mapping from observation to “label”. They can also estimate output probabilities

Maximum A Posteriori (MAP) estimation

$$p(\mathbf{m}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{m})\, p(\mathbf{m})}{p(\mathbf{y})}$$

Posterior is a distribution and can be difficult to compute for high dimensional **m**

Mode of the distribution is often easier to compute computationally

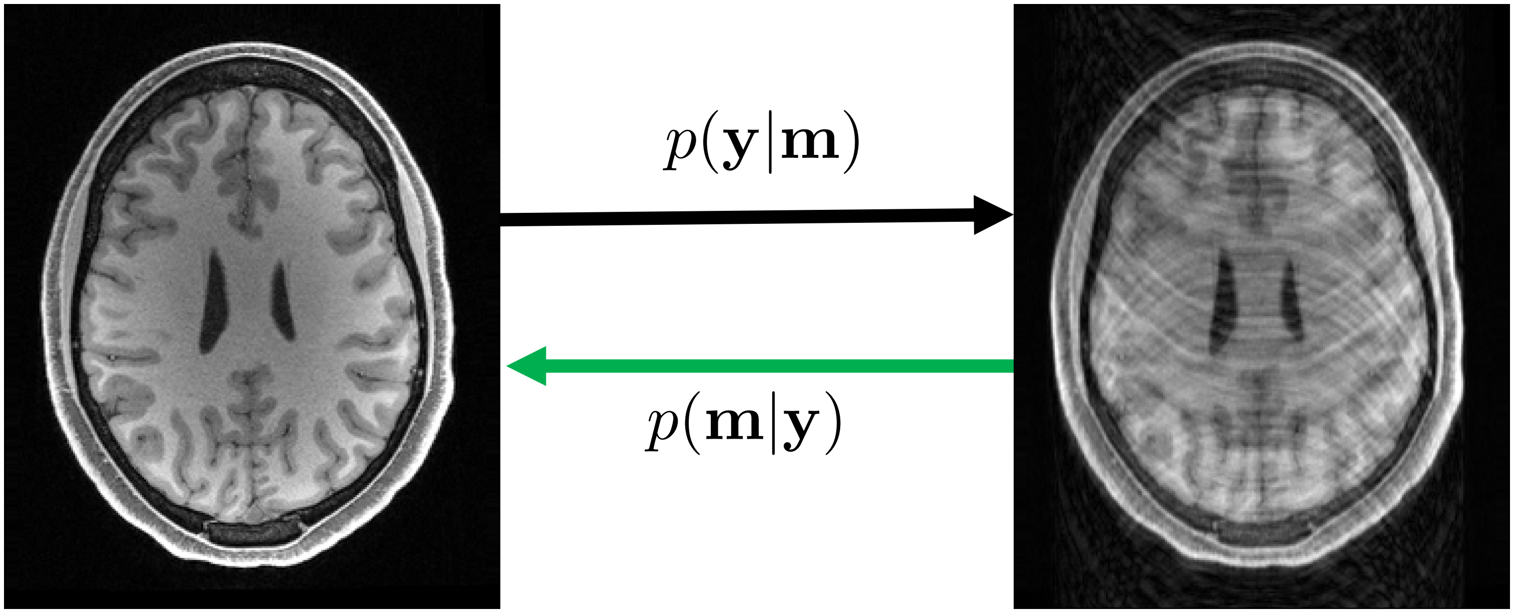
$$\begin{aligned}
\mathbf{m}^* &= \arg\max_{\mathbf{m}} p(\mathbf{m}|\mathbf{y}) \\
&= \arg\max_{\mathbf{m}} p(\mathbf{y}|\mathbf{m}) p(\mathbf{m}) \\
&= \arg\max_{\mathbf{m}} \ln p(\mathbf{y}|\mathbf{m}) + \ln p(\mathbf{m})
\end{aligned}$$

From the optimization perspective, prior is a regularization term

$$\mathbf{m}^* = \arg\max_{\mathbf{m}} \underbrace{\ln p(\mathbf{y}|\mathbf{m})}_{\text{Data term}} + \underbrace{\ln p(\mathbf{m})}_{\text{Regularization}}$$

$$\text{Ridge regression example: } \|\mathbf{X}\mathbf{w} - \mathbf{t}\|_2^2 + \|\mathbf{w}\|_2^2$$

An intuitive look at MAP

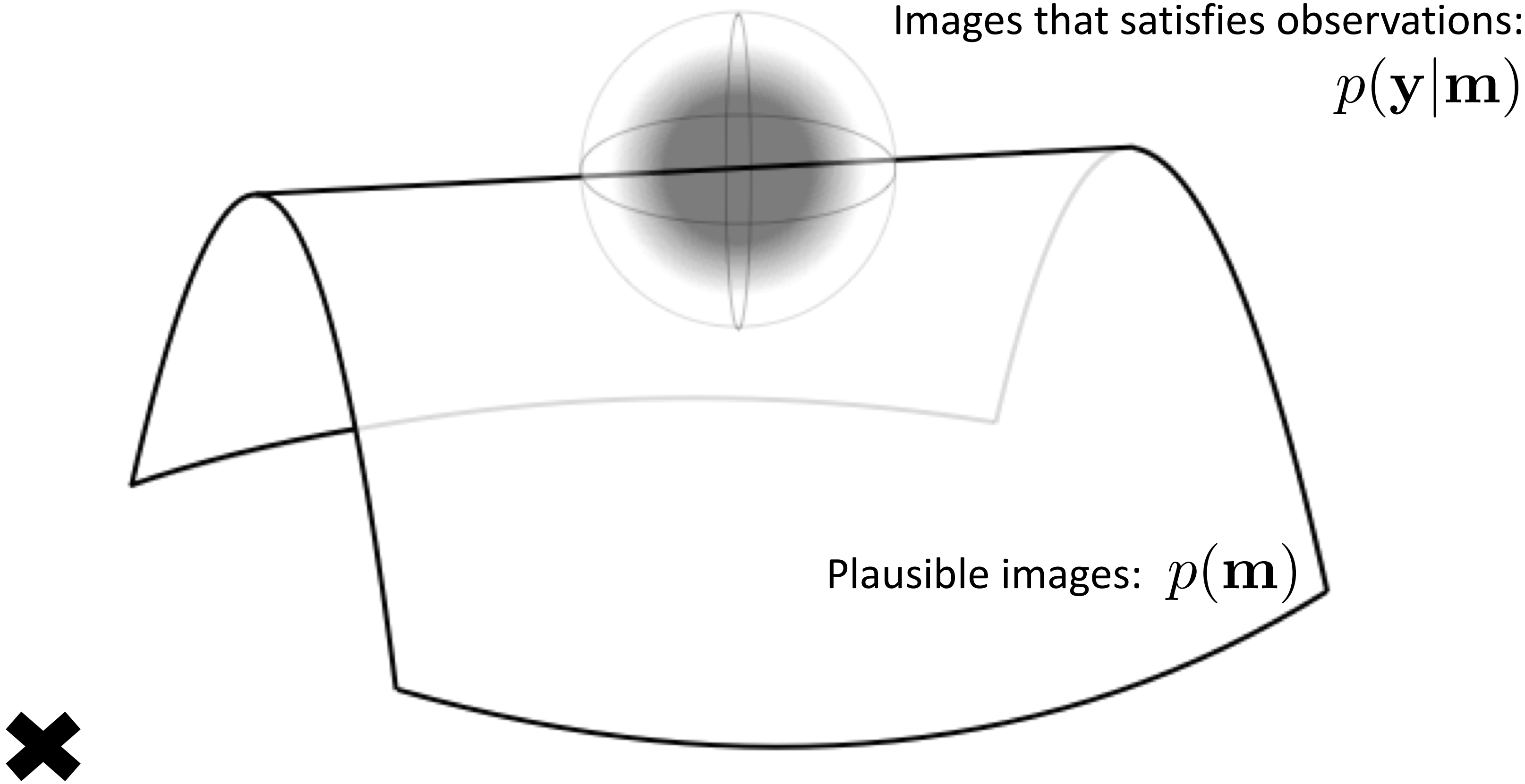


$$p\left(\mathbf{m}|\mathbf{y}\right) = \frac{p\left(\mathbf{y}|\mathbf{m}\right) p(\mathbf{m})}{p(\mathbf{y})}$$

$$\mathbf{m}^* = \arg\max_{\mathbf{m}} \ln p(\mathbf{y}|\mathbf{m}) + \ln p(\mathbf{m})$$

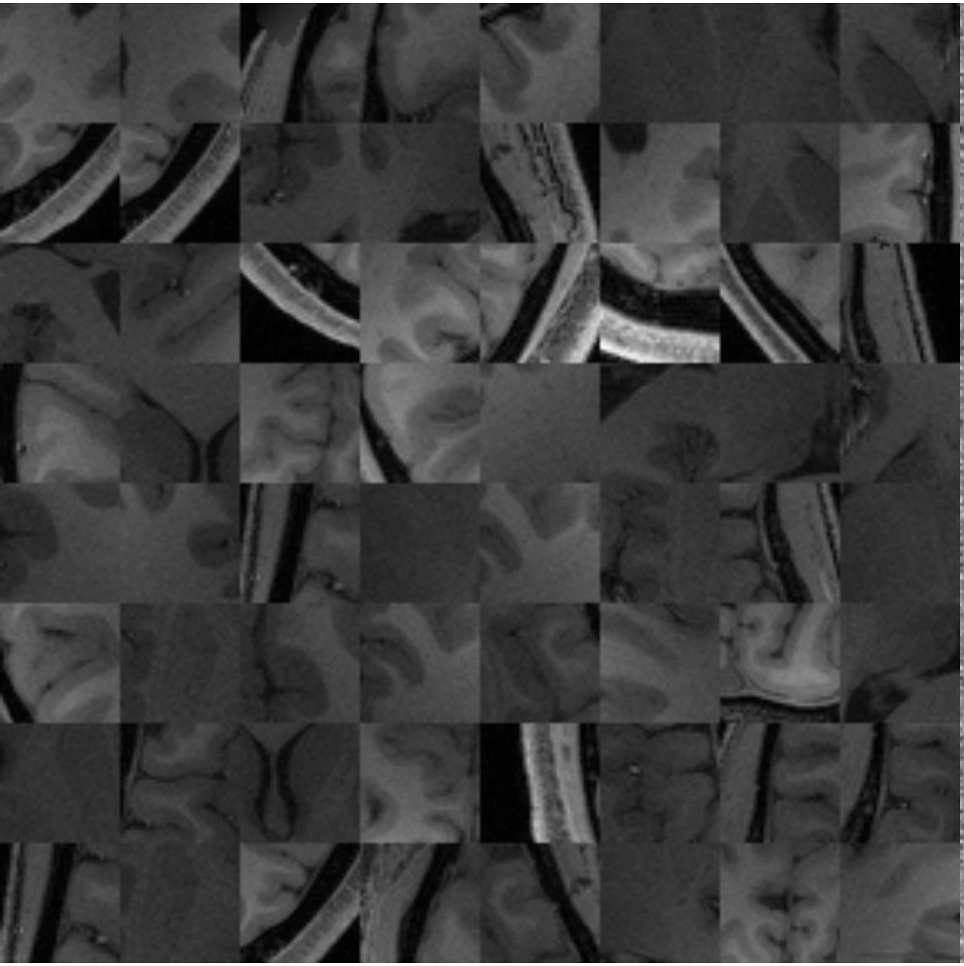
For simplicity assume **y** and **m** are image patches

Cartoon representation*

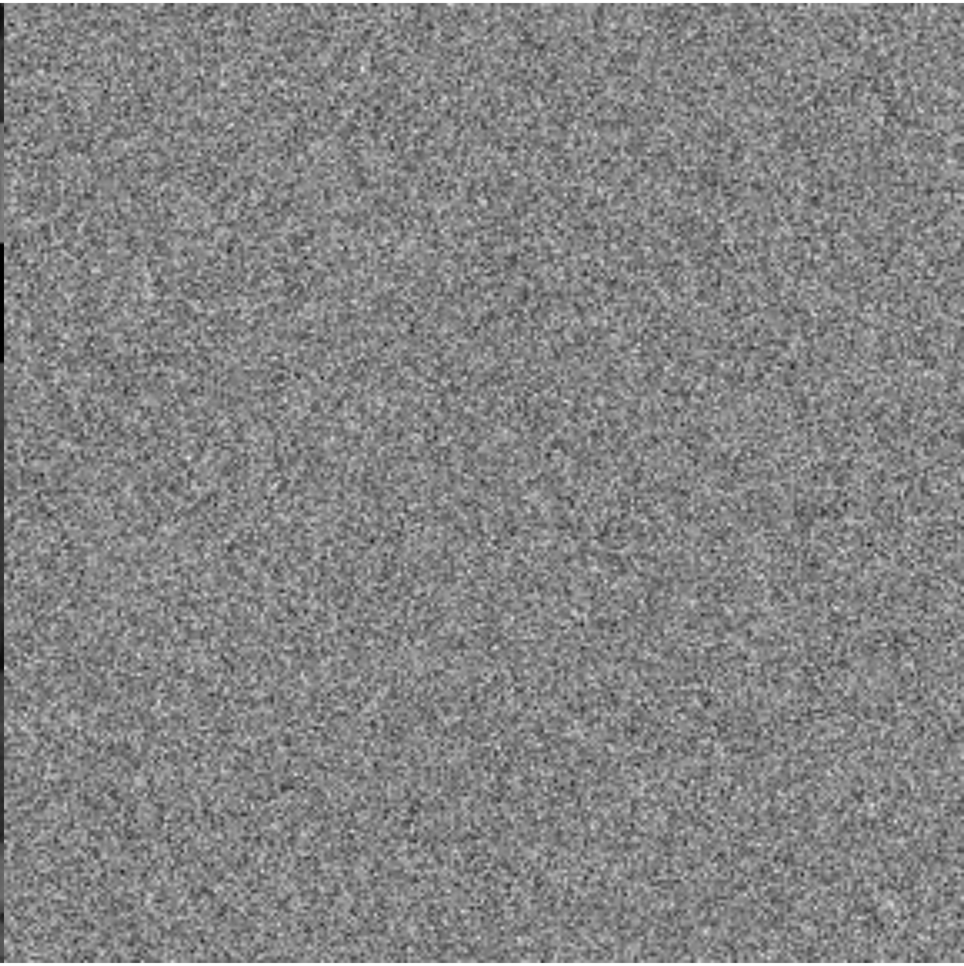


*Artistic illustration

Samples

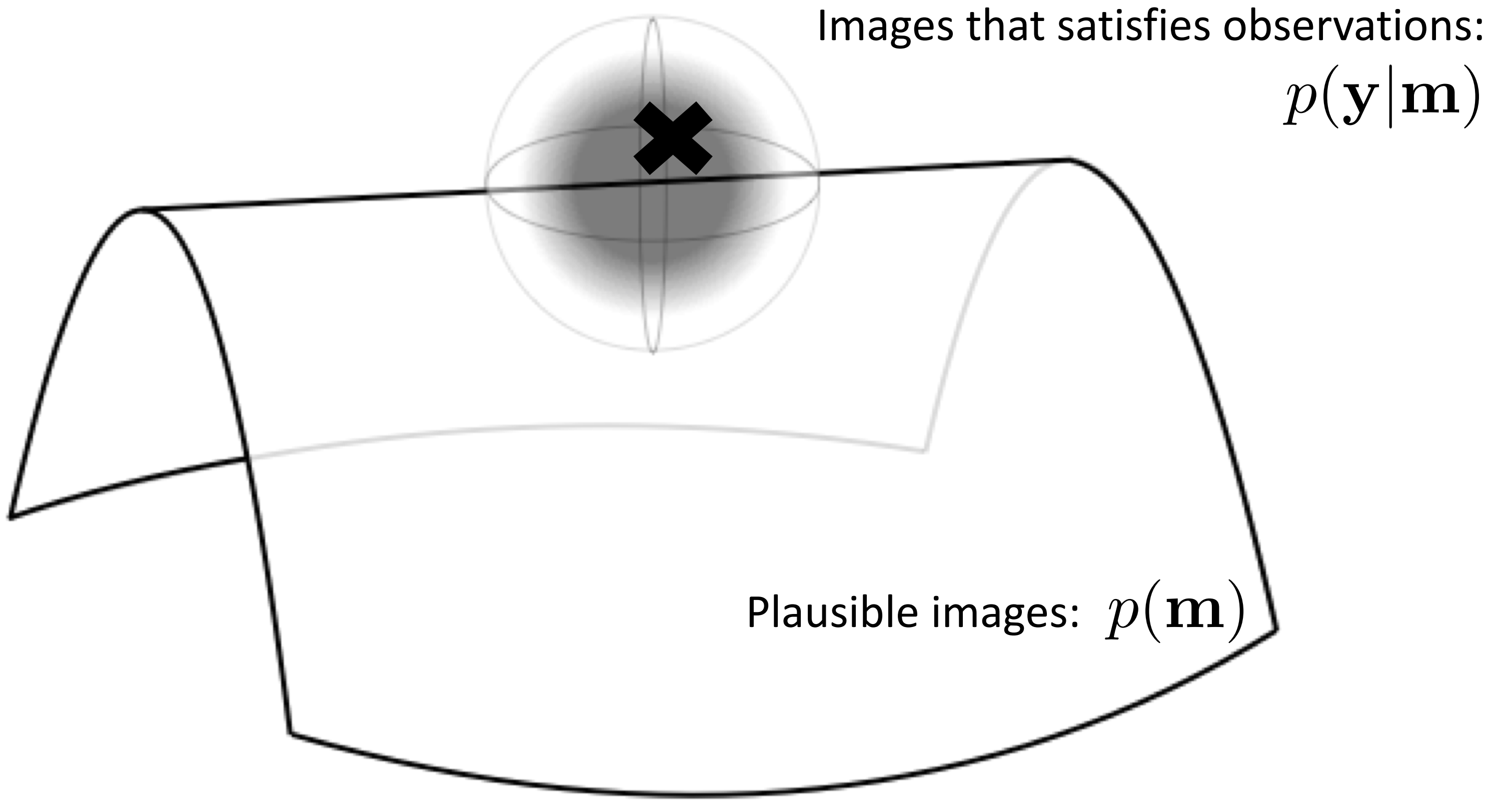


“Normal” Patches



Noise Patches

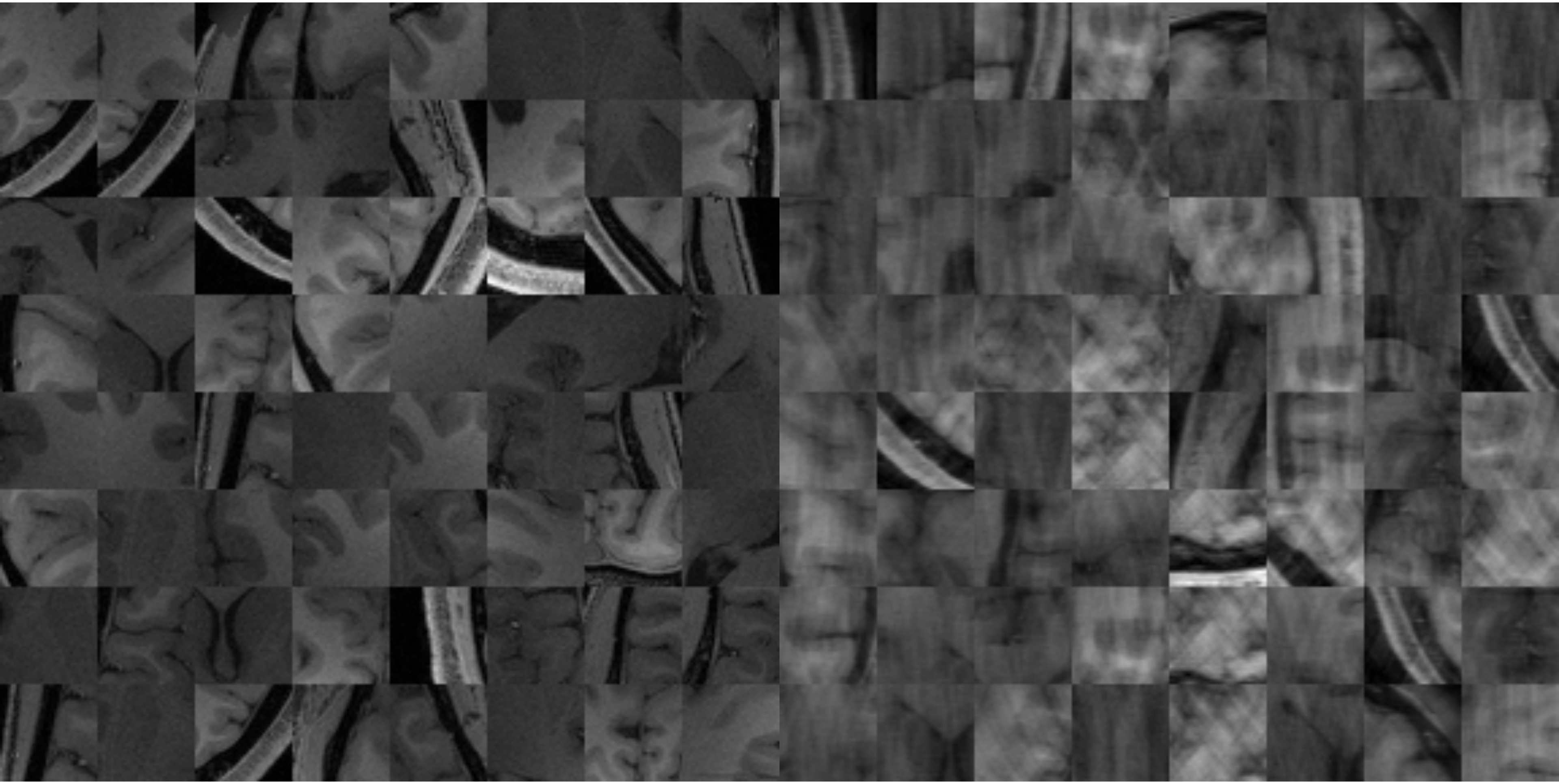
Cartoon representation*



*Artistic illustration

ETH zürich

Samples

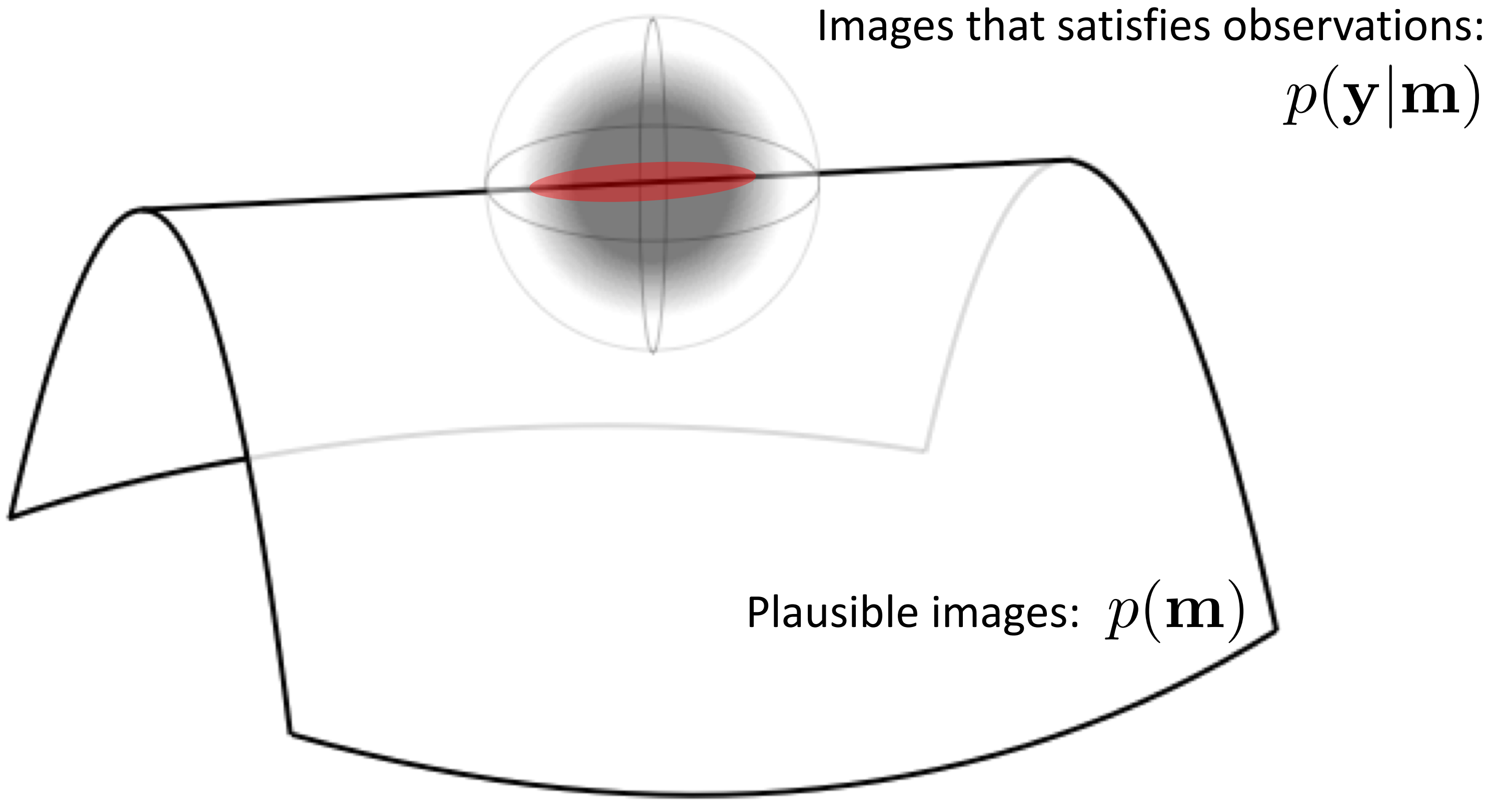


“Normal” Patches

Noisy* Patches

* Patches from an image under-sampled in k-space

Cartoon representation*

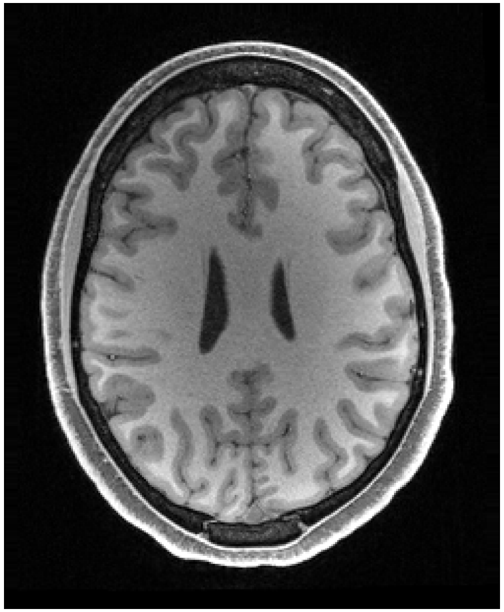


$p(\mathbf{m}|\mathbf{y})$ would lie in the area shown in red

$\mathbf{m}^*$ would be the point with the highest probability in the same area

*Artistic illustration

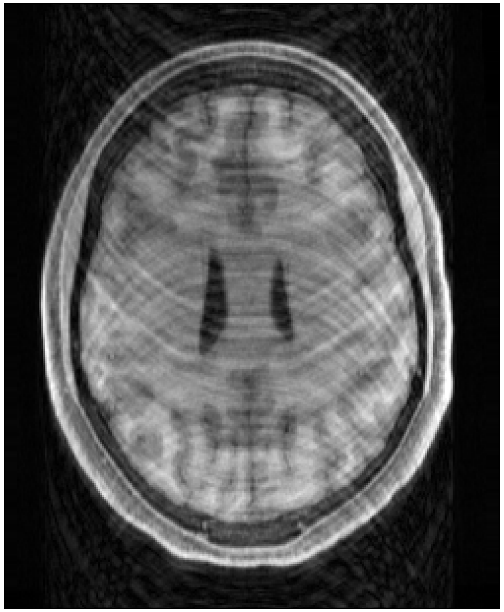
An intuitive look at MAP



Real image
Ground truth

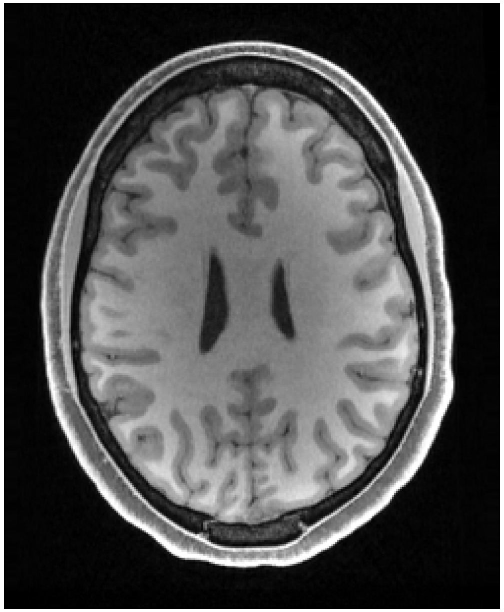
$p(\mathbf{m})$

A sample from the perfect prior



Observed Image

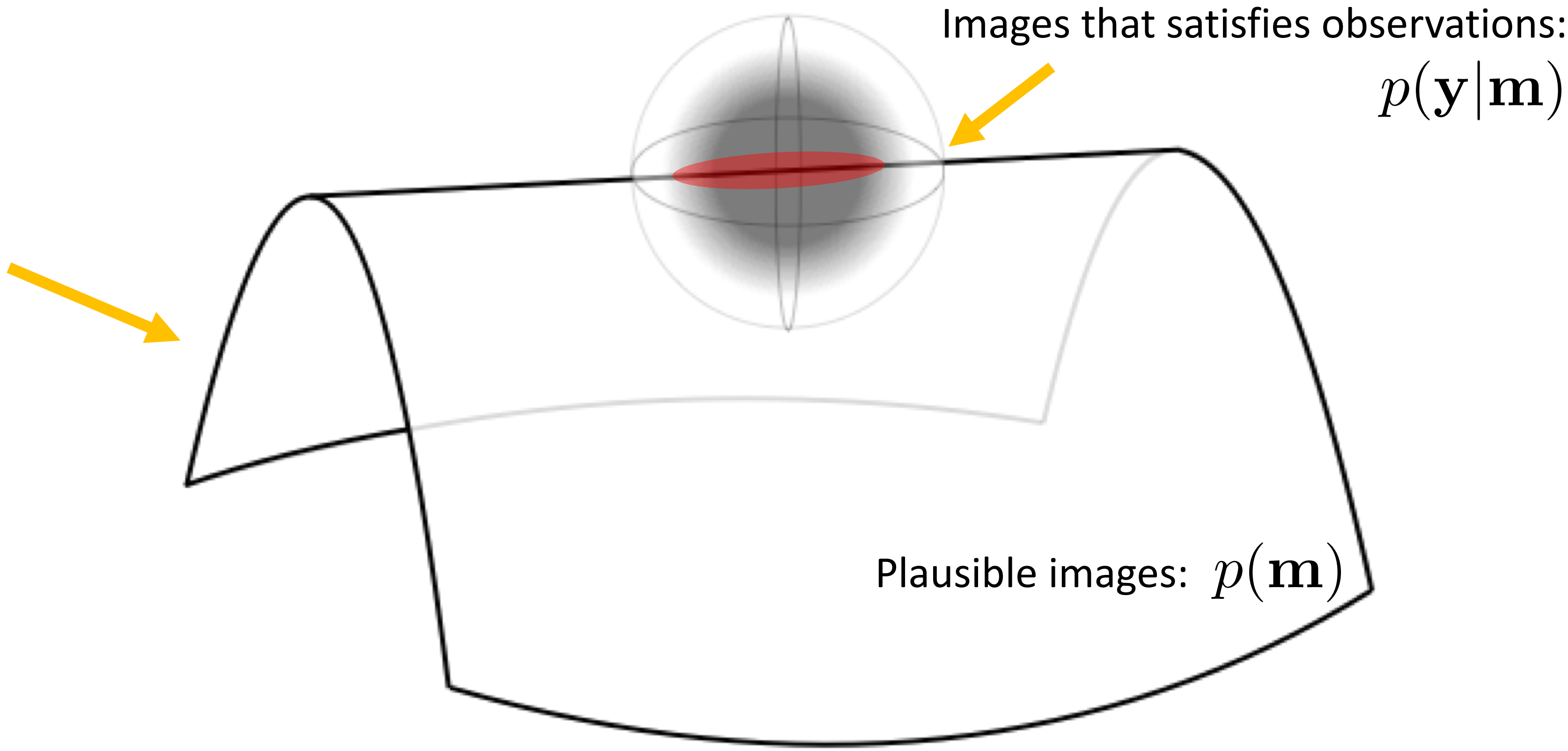
$\mathbf{y}$



Reconstructed

$\mathbf{m}^*$

Cartoon representation*



$p(\mathbf{m}|\mathbf{y})$ would lie in the area shown in red

$\mathbf{m}^*$ would be the point with the highest probability in the same area

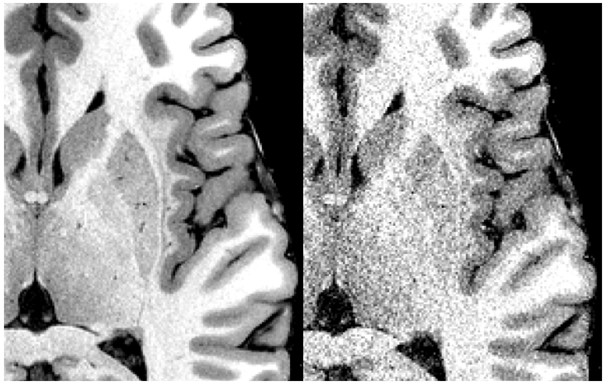
*Artistic illustration

Why use generative models for analyzing images?

$$\mathbf{m}^* = \arg\max_{\mathbf{m}} \underbrace{\ln p(\mathbf{y}|\mathbf{m})}_{\text{Data term}} + \underbrace{\ln p(\mathbf{m})}_{\text{Regularization}} \quad \text{vs} \quad \mathbf{m}^* = f_\theta(\mathbf{y})$$

- Improving robustness with prior knowledge

- General algorithms due to decoupling of prior and observation model
  Going beyond only labeled dataset

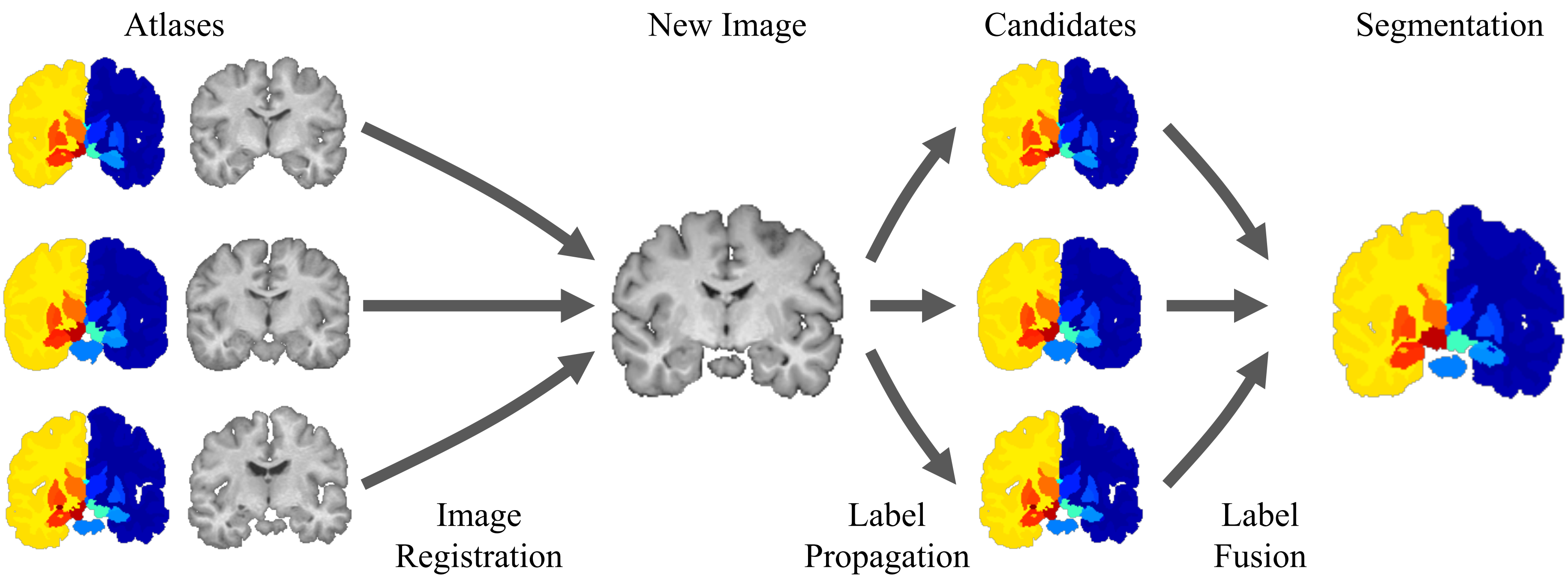


- Potentially low number of labeled examples
- Prior anatomical/physiological knowledge in medical image computing

Outline

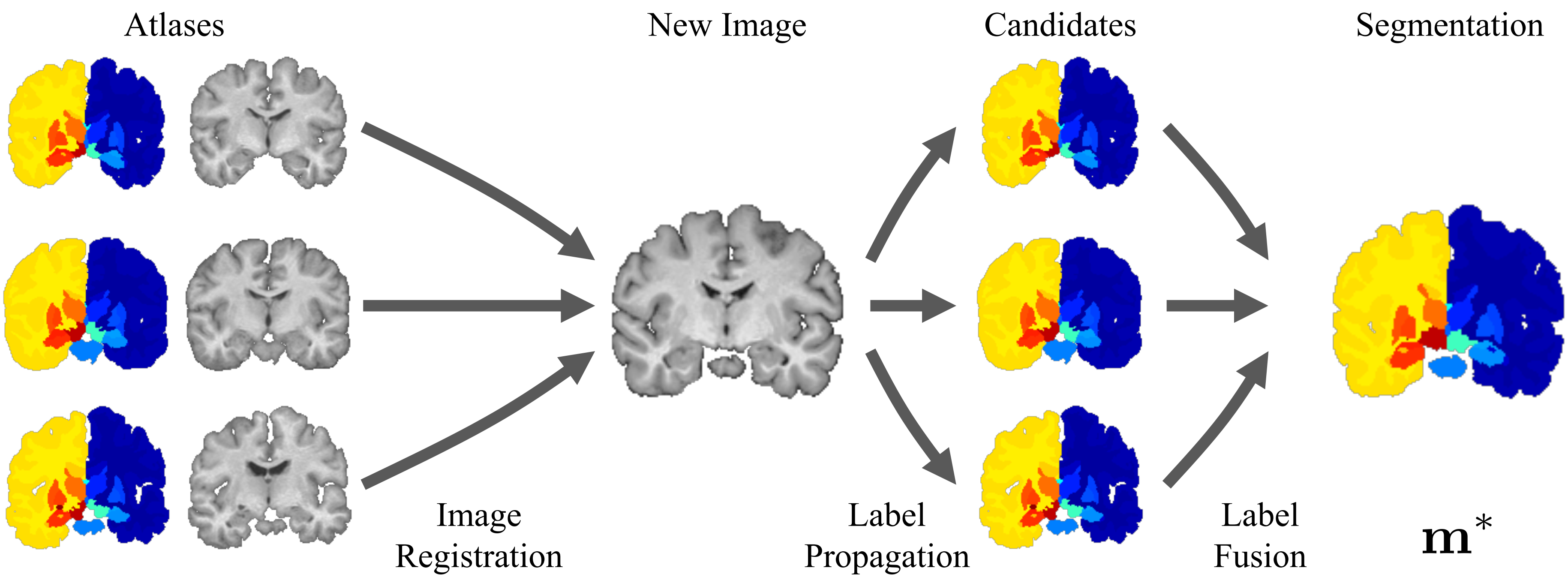
- Basics
- Examples without deep learning
  - Atlas-based segmentation
  - Shape models
  - Image enhancement
- Unsupervised deep learning models
- Three ways to use DL models as priors

Atlas-based segmentation



[Leemput et al., TMI 1999]
[Ashburner & Friston, Neuroimage 2005]
[Pohl et al. MedIA 2007]
[Leemput et al. Hippocampus 2009]
[Sabuncu et al. TMI 2010]
[Simpson et al. MICCAI 2013]
[Iglesias et al. MedIA 2015]
…..

Atlas-based segmentation – simplest model



$\mathbf{m}$: segmentation

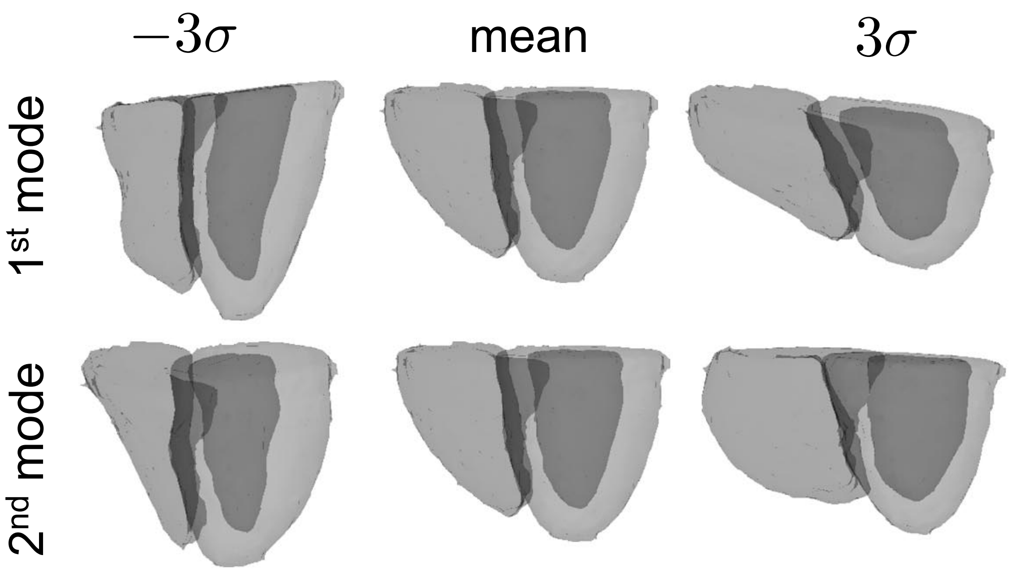
$\mathbf{y}$: (MRI) image

$p(\mathbf{m})$

Pixel-wise prior model based on aligned examples "atlases"

Shape models for segmentation and localization

Principal Component Analysis Model

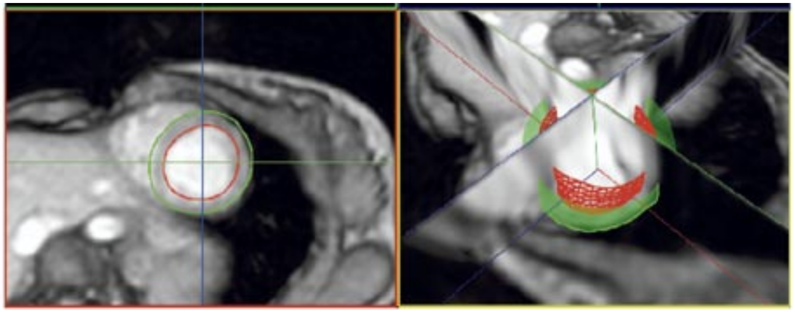


[Image from Frangi et al. TMI 2002]

$$\mathbf{m} : \text{Shapes as point distributions}$$
$$\mathbf{y} : \text{Observed ``noisy'' points}$$

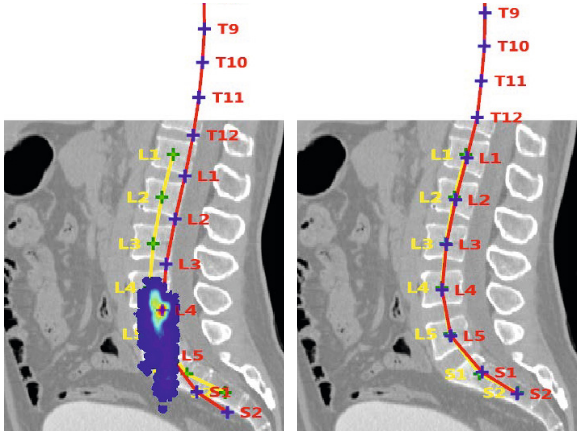
[Cootes et al., CVIU 1995]
[Frangi et al., TMI 2002]
[Mitchell et al. TMI 2002]
[Heimann et al. MedIA 2009]
…..

Matching to imaging data



[Image from Mitchell et al. TMI 2002]

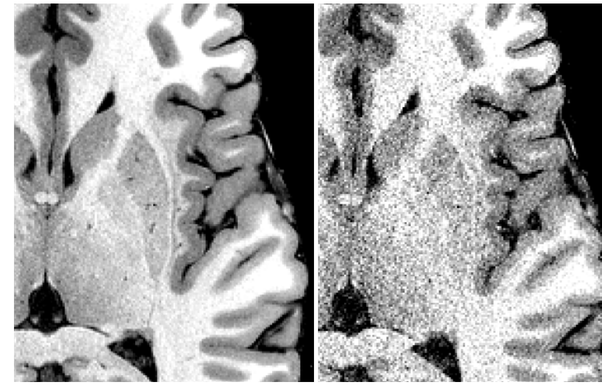
Hidden Markov Models



[Image from Glocker et al. Miccai 2012]

Image enhancement with theoretical priors

$$\mathbf{m}^* = \arg\max_{\mathbf{m}} \underbrace{\ln p(\mathbf{y}|\mathbf{m})}_{\text{Data term}} + \underbrace{\ln p(\mathbf{m})}_{\text{Regularization}}$$



$\mathbf{m}$ $\mathbf{y}$

$$\mathbf{m}^* = \arg\min \frac{\|E\mathbf{m} - \mathbf{y}\|_2^2}{\sigma^2} - \ln p(\mathbf{m})$$

E: any image encoding operation, e.g. subsampling

$$p(\mathbf{m}) \propto \exp\{-\|\Psi(\mathbf{m} - \mu)\|_p\}$$

$$\mu = 0, \Psi = \mathbb{I}, p = 2$$

Simple ridge regression

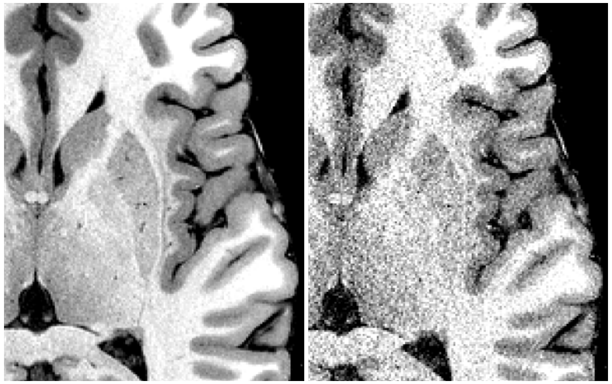
$$\mu = 0, \Psi = \nabla, p = 1$$

Total variation compressed sensing

[Rudin et al., Physica D 1992]
[Lustig et al. MRM 2007]
[Elad and Aharon IEEE TIP 2006]
[Ravishankar and Bressler TMI 2011]
[Babacan et al. TIP 2010]
…

Image enhancement with dictionaries

$$\mathbf{m}^* = \arg\max_{\mathbf{m}} \underbrace{\ln p(\mathbf{y}|\mathbf{m})}_{\text{Data term}} + \underbrace{\ln p(\mathbf{m})}_{\text{Regularization}}$$



$\mathbf{m}$ $\mathbf{y}$

Sparse dictionaries
[Elad and Aharon IEEE TIP 2006]

$$\alpha^* = \arg\min_{\alpha} \frac{\|ED\alpha - \mathbf{y}\|_2^2}{\sigma^2} + \|\alpha\|_1$$

$$\mathbf{m}^* = D\alpha^*$$

D: over-complete dictionary learned from examples
α: sparse code

Similar model with probabilistic PCA

$$\alpha^* = \arg\min_{\alpha} \frac{\|ED\alpha - \mathbf{y}\|_2^2}{\sigma^2} + \|\alpha\|_2$$

$$\mathbf{m}^* = D\alpha^*$$

D: PCA matrix
α: PCA code

[Rudin et al., Physica D 1992]
[Lustig et al. MRM 2007]
[Elad and Aharon IEEE TIP 2006]
[Ravishankar and Bressler TMI 2011]
[Babacan et al. TIP 2010]
…

Outline

- Basics
- Examples without deep learning
- Unsupervised deep learning models
  - Density networks
  - Variational auto-encoders
  - Generative adversarial networks
- Three ways to use DL models as priors

The prior $p(\mathbf{m})$

Dimensionality of **m** makes building a prior difficult
There was no good model for image priors

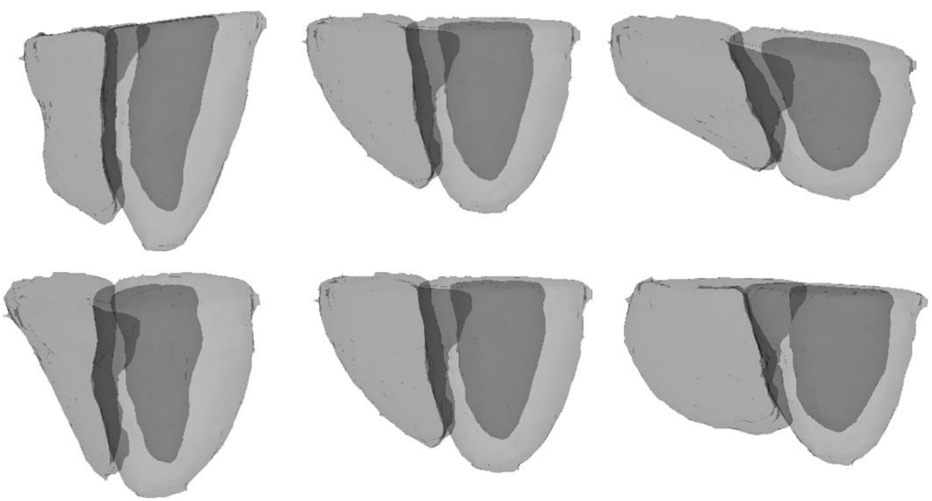
$$p(\mathbf{m}) \propto \exp\{-\|\Psi(\mathbf{m}-\mu)\|_p\}$$

Theoretical

-3σ mean 3σ

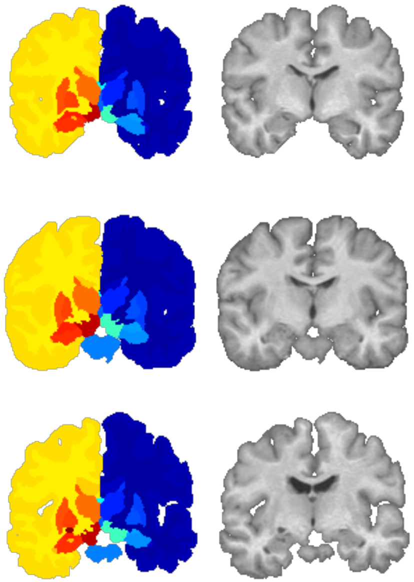
1st mode

2nd mode



Linear model with PCA

Atlases



$$\mathbf{m} = D\alpha$$
$$p(\alpha) \propto \prod \exp\{-|\alpha_i|\}$$

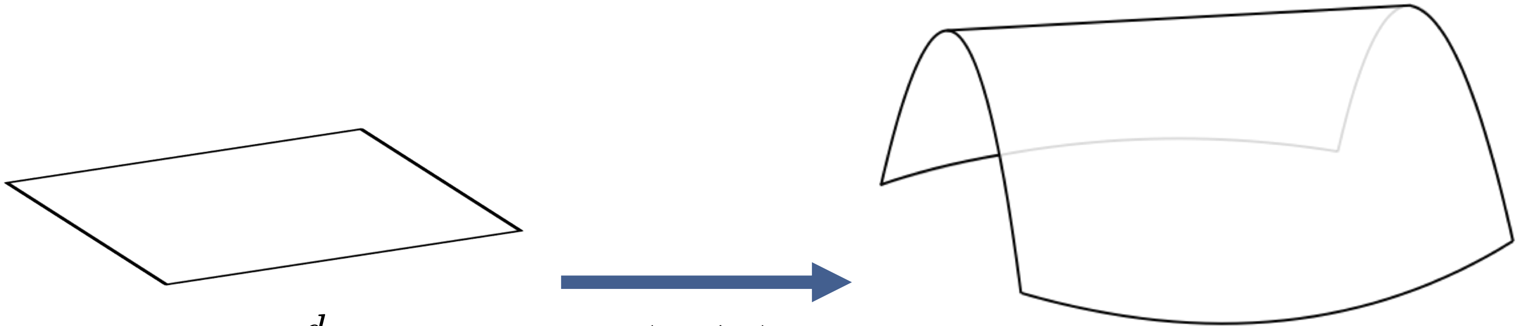
Sparse dictionary

Deep learning has alternatives

Latent variable models

$$p(\mathbf{m}) = \int p(\mathbf{m}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

$$\mathbf{z} \in \mathbb{R}^d \quad \mathbf{m} \in \mathbb{R}^D \quad d << D$$



$\mathbf{z} \in \mathbb{R}^d$ $\quad p(\mathbf{m}|\mathbf{z}) \quad$ $\mathbf{m} \in \mathcal{M} \subset \mathbb{R}^D$

latent space

image space

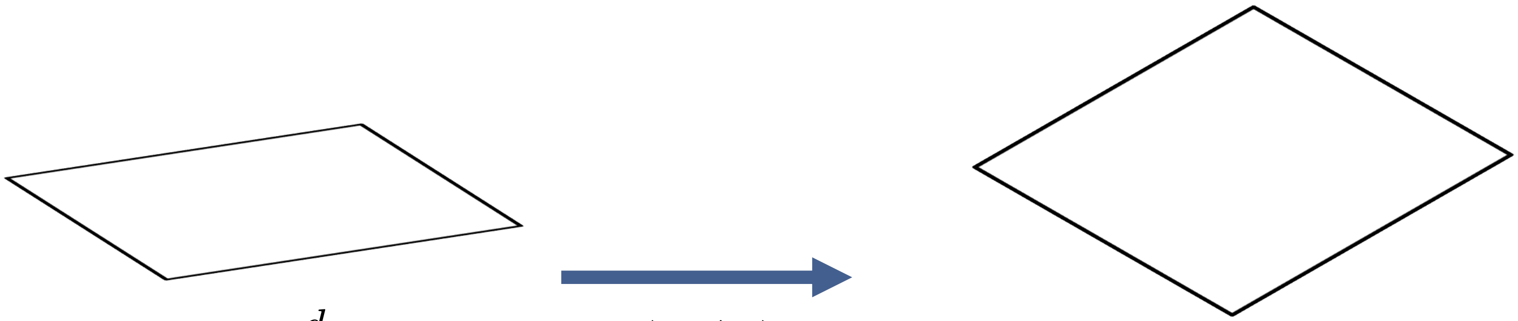
Principal component analysis

$$p(\mathbf{m}) = \int p(\mathbf{m}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

$$p(\mathbf{m}|\mathbf{z}) = \mathcal{N}\left(\mathbf{U}\mathbf{z}, \sigma^2\mathbf{I}\right)$$

$$\mathbf{z} \in \mathbb{R}^d \quad \mathbf{m} \in \mathbb{R}^D \quad d << D$$

$$p(\mathbf{z}) = \mathcal{N}\left(0, \mathbf{I}\right)$$



$\mathbf{z} \in \mathbb{R}^d$ $\quad p(\mathbf{m}|\mathbf{z})$ $\quad \mathbf{m} \in \mathcal{M} \subset \mathbb{R}^D$

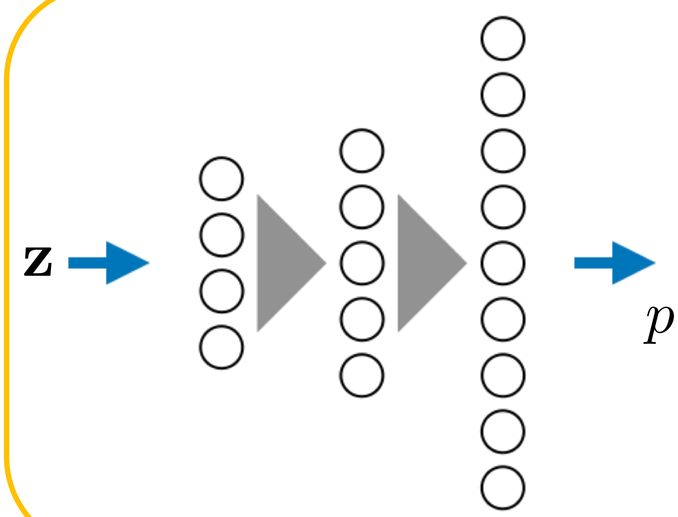
latent space

image space

Density networks
[MacKay, Nucl. Inst. Met. In Physics Research 1995]

$p(\mathbf{m}|\mathbf{z})$: Parameterize with a network with parameters **w**

$$p(\mathbf{m}|\mathbf{w}) = \int p(\mathbf{m}|\mathbf{z},\mathbf{w})p(\mathbf{z})d\mathbf{z}$$



Optimize with respect to **w**

$$\prod_n p(\mathbf{m}^{(n)}|\mathbf{w})$$

$$p(\mathbf{m}|\mathbf{z},\mathbf{w}) \quad = \quad \prod_n \int p(\mathbf{m}^{(n)}|\mathbf{z}^{(n)},\mathbf{w})p(\mathbf{z}^{(n)})d\mathbf{z}^{(n)}$$

using Monte Carlo sampling

$$\sum_n \ln \frac{1}{R} \sum_r p(\mathbf{m}^{(n)}|\mathbf{z}^{(r)},\mathbf{w}), \;\; \mathbf{z}^{(r)} \sim p(\mathbf{z})$$

Sampling was not efficient for large dim. problems, MacKay hinted imp. samp.

Variational auto-encoders

[Kingma and Welling 2013, Rezende et al. 2013]

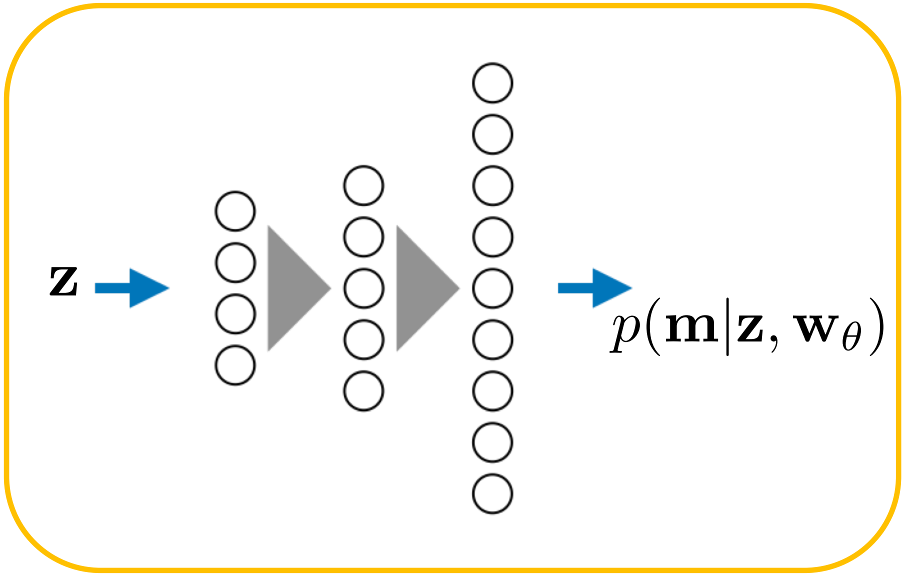
Builds on density networks concept but instead of Monte-Carlo uses variational inference with a network parameterized sampling (approximate) distribution

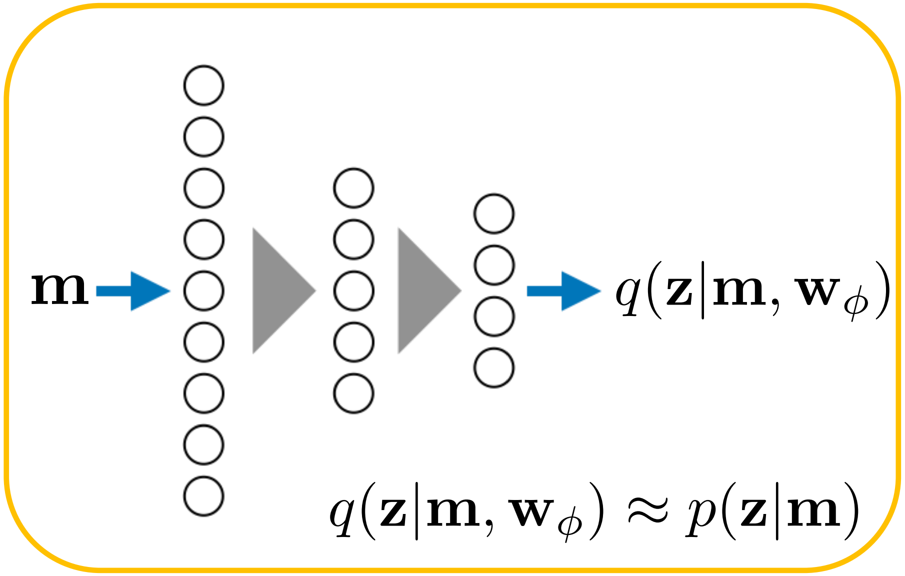
$$\ln p(\mathbf{m}|\mathbf{w}_\theta) = \ln \int p(\mathbf{m}|\mathbf{z}, \mathbf{w}_\theta) p(\mathbf{z}) d\mathbf{z}$$

$$\ln p(\mathbf{m}|\mathbf{w}_\theta) - \left(\mathbb{E}\left[\ln p(\mathbf{m}|\mathbf{z}, \mathbf{w}_\theta)\right] - D_{KL}\left[q(\mathbf{z}|\mathbf{x}, \mathbf{w}_\phi) \| p(\mathbf{z})\right]\right) = D_{KL}\left[q(\mathbf{z}|\mathbf{x}, \mathbf{w}_\phi) \| p(\mathbf{z}|\mathbf{x})\right]$$

Variational auto-encoders

[Kingma and Welling 2013, Rezende et al. 2013]





Optimize jointly with backpropagation and reparameterization trick for Gaussian distributions:

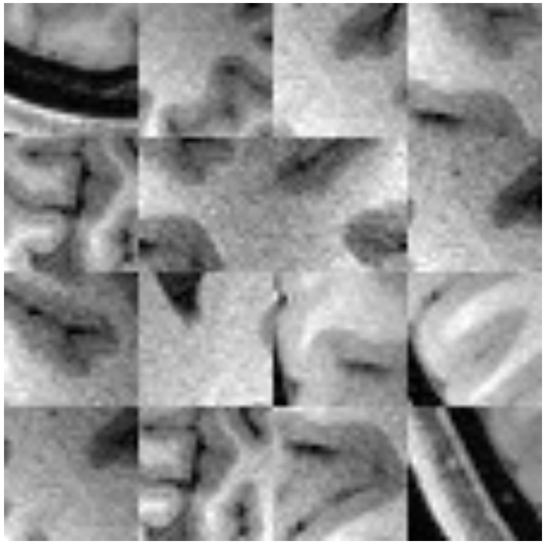
$$q(\mathbf{z}|\mathbf{m},\mathbf{w}_\phi) = \mathcal{N}(\mu_\mathbf{z}, \Sigma_\mathbf{z}) \qquad p(\mathbf{m}|\mathbf{z},\mathbf{w}_\theta) = \mathcal{N}(\mu_\mathbf{x}, \Sigma_\mathbf{x}) \qquad p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$$

$$\arg\max_{\mathbf{w}_\theta,\mathbf{w}_\phi} \frac{1}{R}\sum_r \ln p(\mathbf{m}|\mathbf{z}^r,\mathbf{w}_\theta) - D_{KL}\left[q(\mathbf{z}|\mathbf{m},\mathbf{w}_\phi)\|p(\mathbf{z})\right]$$

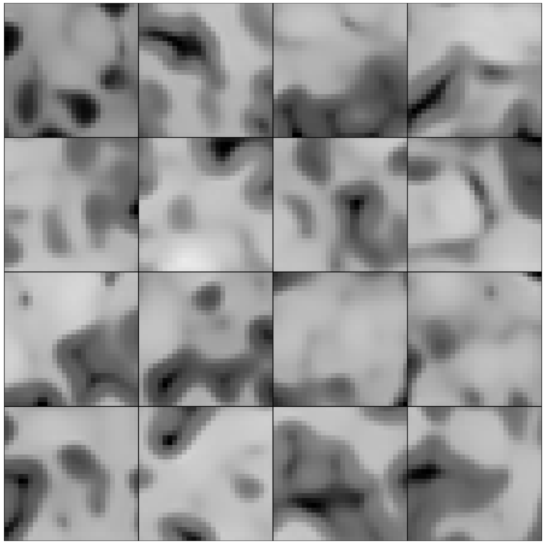
$$\mathbf{z}^r = \mu_\mathbf{z} + \Sigma_\mathbf{z}^{1/2}\epsilon^r, \;\; \epsilon^r \sim \mathcal{N}(0,\mathbf{I})$$

Samples from learned distributions via VAE and PCA

Real patches of 28x28

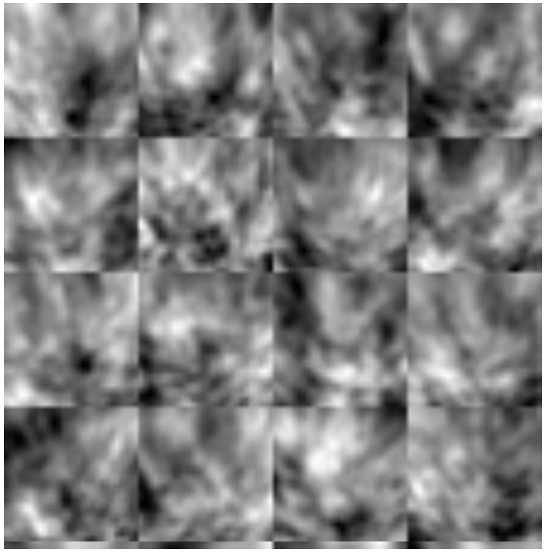


VAE Generated



60 components

PCA



250 components
60 components

Generative adversarial networks

[Goodfellow et al. NIPS 2014]

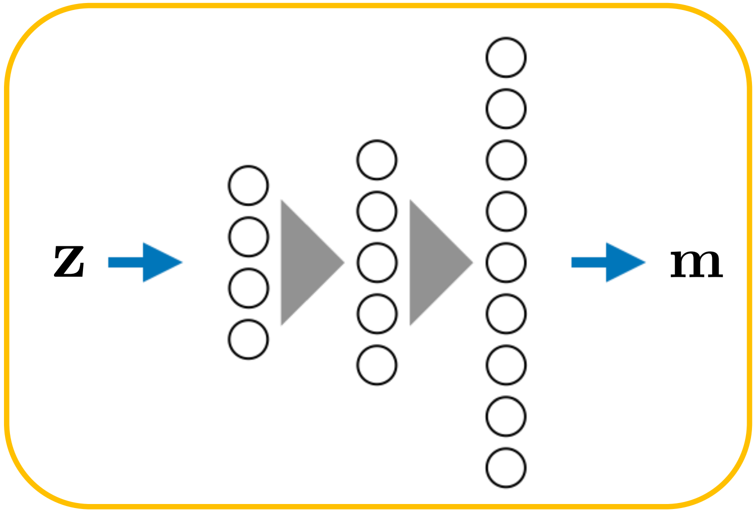
Instead of an explicit probabilistic model, a GAN is a sampling tool that generates samples from the data distribution

Optimize the network weights with a two player game

$$\min_{\mathbf{w}_\theta} \sup_{\mathbf{w}_\phi} \mathbb{E}_{\mathbf{m}\sim p_d(\mathbf{m})} \left[\ln D\left(\mathbf{m}|\mathbf{w}_\phi\right)\right] + \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \left[\ln(1 - D\left(G\left(\mathbf{m}|\mathbf{z}, \mathbf{w}_\theta\right)|\mathbf{w}_\phi\right))\right]$$

$$\min_{\mathbf{w}_\theta} \sup_{\mathbf{w}_\phi} \mathbb{E}_{\mathbf{m}\sim p_d(\mathbf{m})} \left[D\left(\mathbf{m}|\mathbf{w}_\phi\right)\right] - \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})} \left[D\left(G\left(\mathbf{m}|\mathbf{z}, \mathbf{w}_\theta\right)|\mathbf{w}_\phi\right)\right]$$

[Arjovsky et al. 2017]



- A generator G and a discriminator D competes and G is the resulting sampling tool
- Notice that in conventional GANs does not have an auto-encoder scheme, i.e. there is no network mapping **m** to a corresponding **z**.
- Extensions exists

Outline

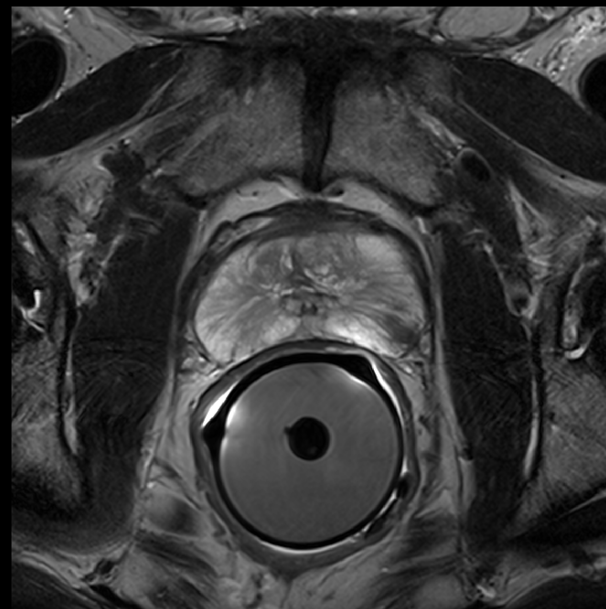
- Basics
- Examples without deep learning
- Unsupervised deep learning models
- Two ways to use DL models as priors
  - Optimization with DL priors
  - Prediction in the latent space
  - Related: predicting latent space of a linear PCA model

Image analysis with DL priors

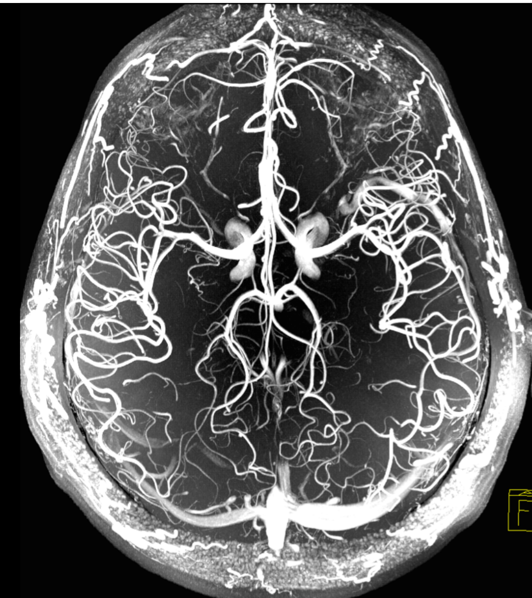
- MR image reconstruction using deep density priors
  [K. Tezcan, C. Baumgartner and E. Konukoglu
  arXiv: 1711.11386]
- Solving Linear Inverse Problems Using GAN Priors: An Algorithm with Provable Guarantees
  [V. Shah and Chinmay Hedge, arXiv: 1802.08406]

Magnetic resonance imaging reconstruction

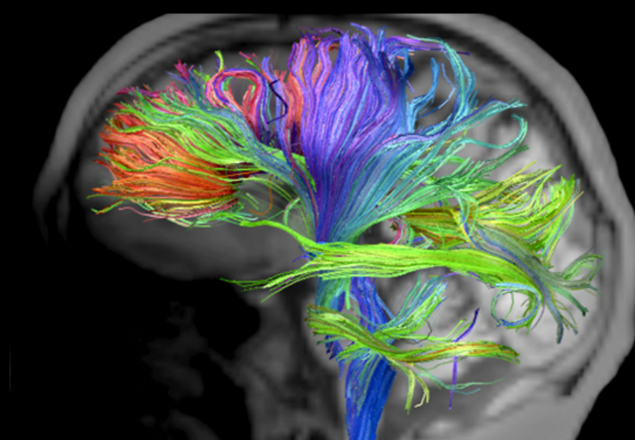
[Visceral challenge]



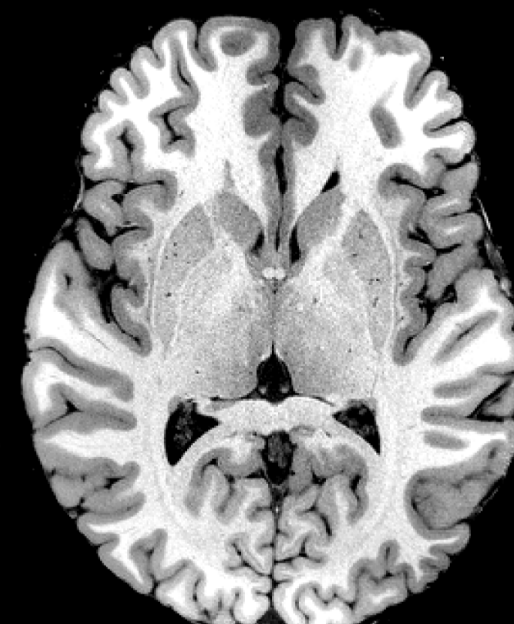
[Olivio Donati / USZ]



[Jon Polimeni/MGH]

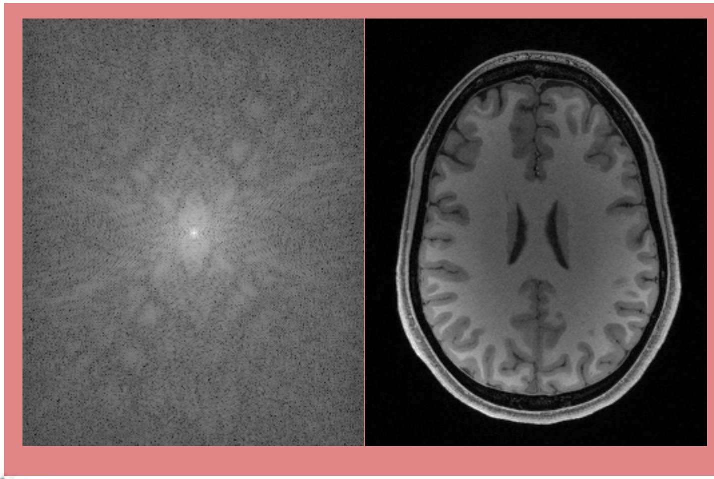


[Siemens Healthcare USA]



[Andre van der Kouwe /MGH]

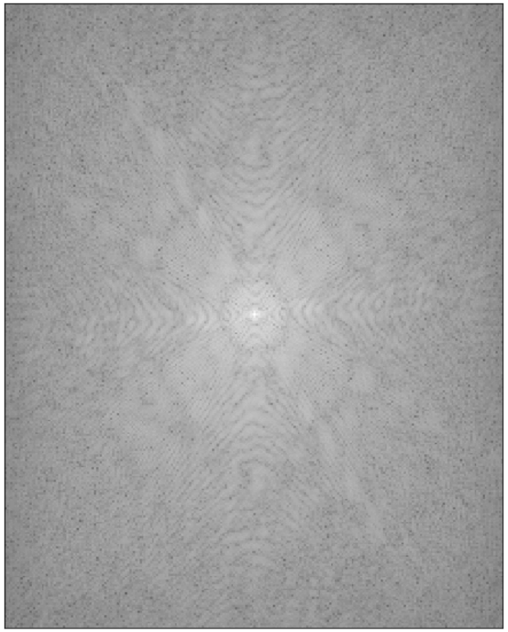
MRI acquisition

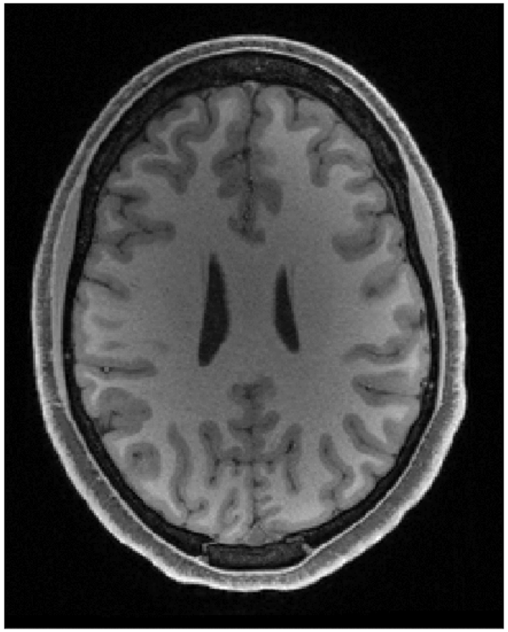


MRI reconstruction – simplistic view

Measurement space (K-space)
Magnitude

Image
Magnitude





Inverse Fourier Transform

MRI reconstruction – simplistic view

Measurement space (K-space)
Magnitude

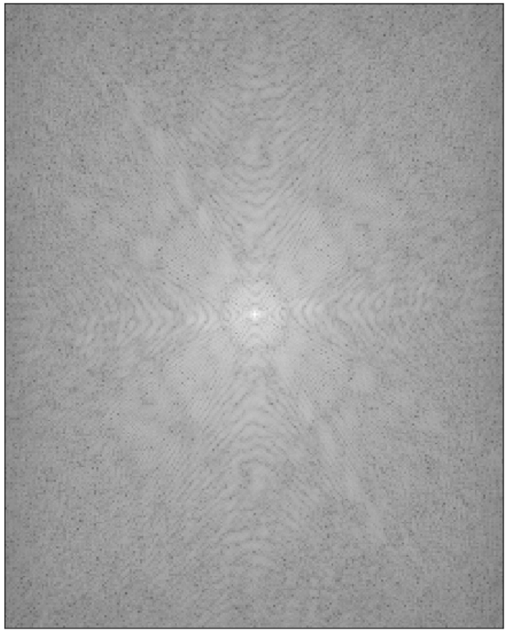
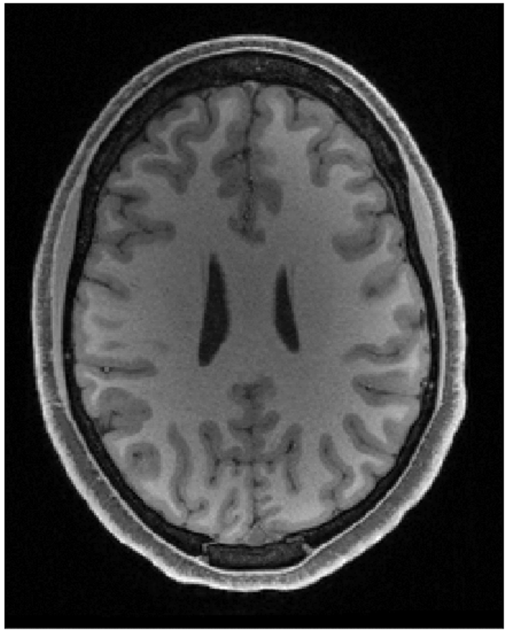
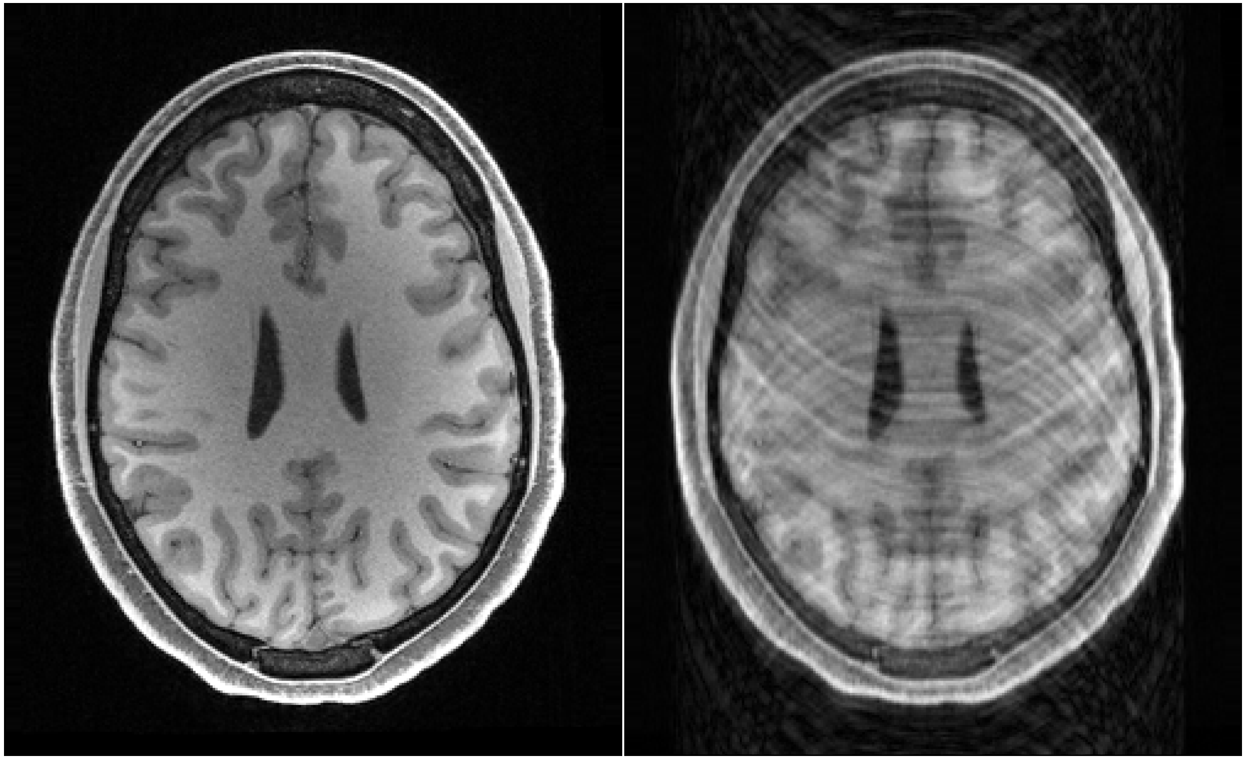


Image
Magnitude



Inverse Fourier Transform

Fewer measurements = less time



Fully sampled image

Under-sampled image
1/3rd the measurements

Semantic information from under-sampled image

- Reconstruction
- Segmentation
- Quantification
- Prediction
- Abnormality detection
- Adaptive scanning
- …

Bayesian model for image reconstruction

$$\mathbf{y} = E\mathbf{m} + \eta, \ \mathbf{m} \in \mathbb{C}^N, \ E \in \mathbb{C}^{M \times N}, \ \mathbf{y} \in \mathbb{C}^M, \ M < N$$

k-space observations = Encoding Operator X **true image** + noise

$$p(\mathbf{y}|\mathbf{m}) = \mathcal{N}(\mathbf{y}|E\mathbf{m}, \sigma_n)$$

Observation model as a Gaussian distribution – thermal noise etc.

$$p(\mathbf{m}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{m})p(\mathbf{m})}{p(\mathbf{y})}$$

Posterior distribution – My initial belief updated by my observations

$$\mathbf{m}^* = \arg_{\mathbf{m}} \max p(\mathbf{m}|\mathbf{y}) = \arg_{\mathbf{m}} \max p(\mathbf{y}|\mathbf{m})p(\mathbf{m})$$

Image that maximizes my current belief after observations

$$\mathbf{m}^* = \arg_{\mathbf{m}} \max \{\log p(\mathbf{y}|\mathbf{m}) + \log p(\mathbf{m})\}$$

Data term Regularization

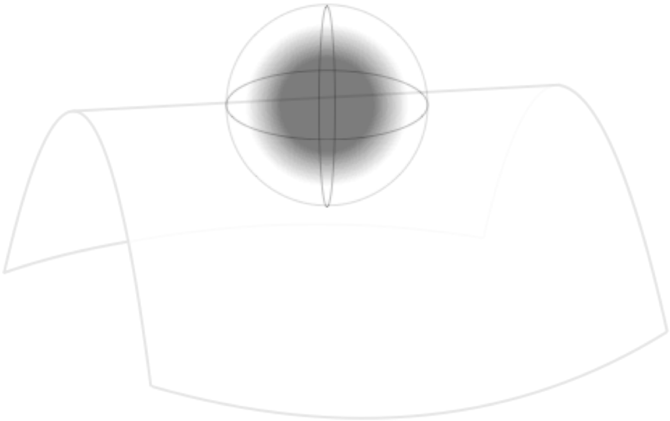
Optimization problem

$$\mathbf{m}^* = \arg_{\mathbf{m}} \max p(\mathbf{m}|\mathbf{y}) = \arg_{\mathbf{m}} \max p(\mathbf{y}|\mathbf{m})p(\mathbf{m})$$

Image that maximizes my current belief after observations

$$p(\mathbf{y}|\mathbf{m}) = \mathcal{N}(\mathbf{y}|E\mathbf{m}, \sigma_n)$$

We have a rough idea about the observation model



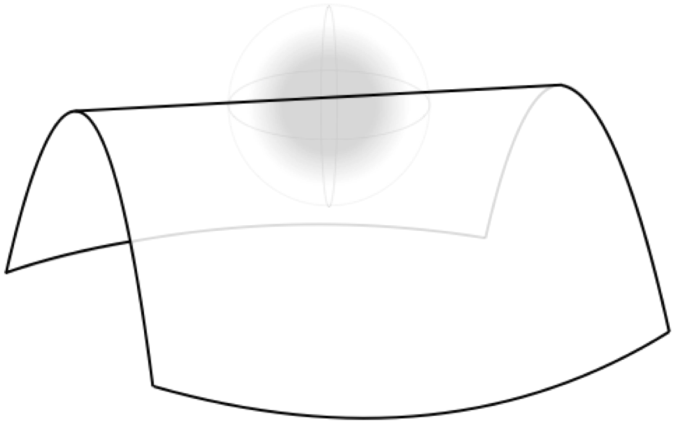
Optimization problem

$$\mathbf{m}^* = \arg_{\mathbf{m}} \max p(\mathbf{m}|\mathbf{y}) = \arg_{\mathbf{m}} \max p(\mathbf{y}|\mathbf{m})p(\mathbf{m})$$

Image that maximizes my current belief after observations

$$p(\mathbf{m})$$

Prior model



- Should have small number of edges / high-frequency regions

$$p(\mathbf{m}) \propto \exp(-\frac{\|\nabla \mathbf{m}\|_1}{b})$$

- Basic non-parametric estimation from examples

  Kernel density estimation

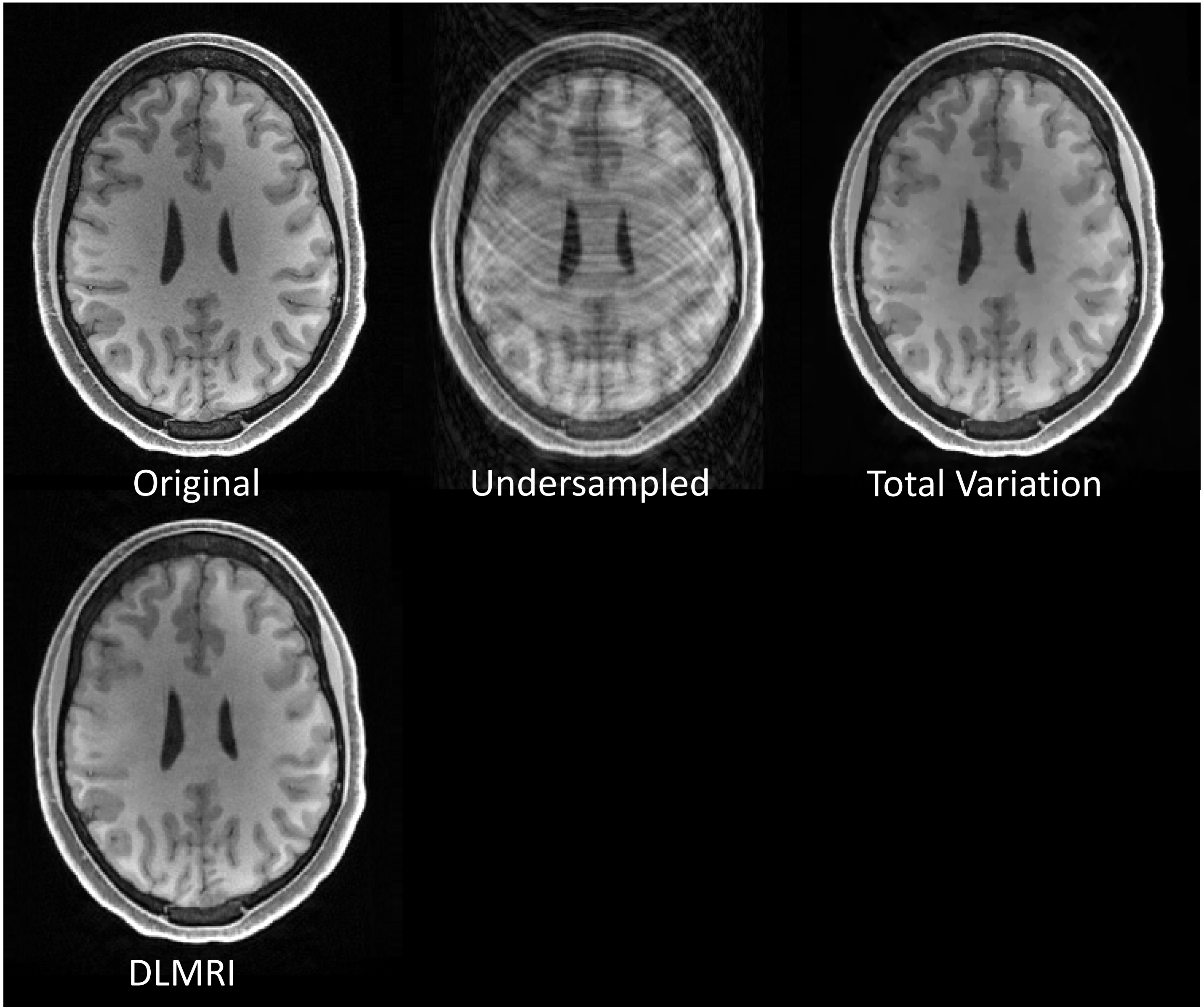
- Live on a hyperplane and form a Gaussian

  Principal Component Analysis

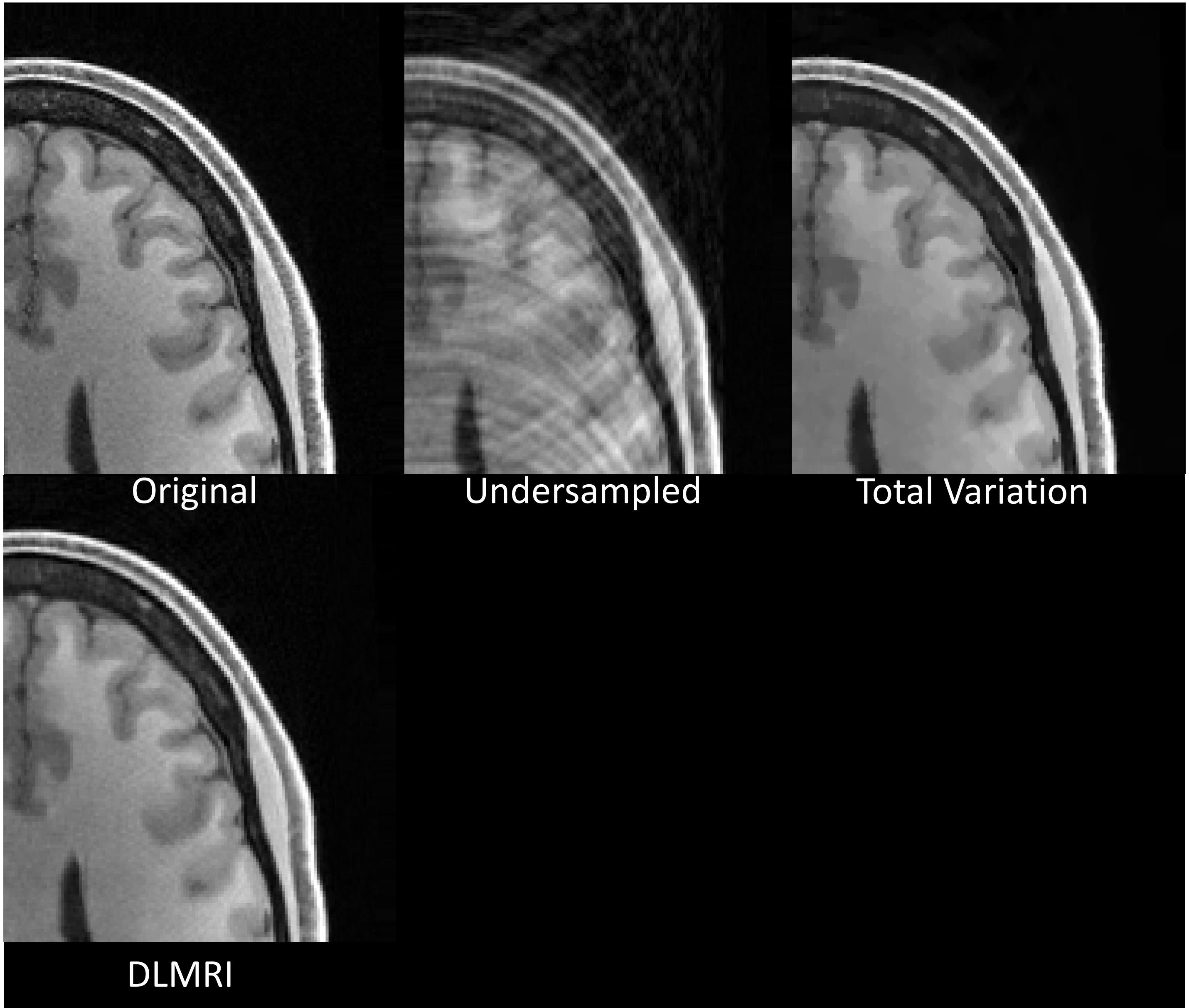
- Should be explained as a sparse combination of some atomic elements

  Sparse dictionary learning,…

Reconstructions with 1/3rd measurements



Reconstructions with 1/3rd measurements

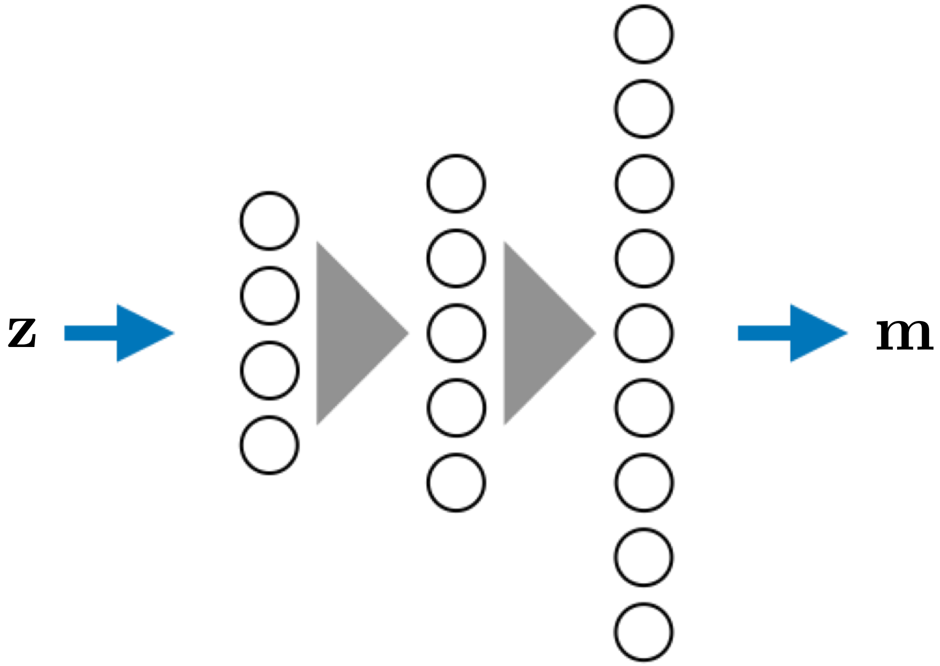


Prior with variational autoencoder model

[Kingma and Welling 2013, Rezende et al. 2013]

$$p(\mathbf{m}) = \int p(\mathbf{m}|\mathbf{z})p(\mathbf{z})d\mathbf{z}, \;\; p(\mathbf{z}) = \mathcal{N}(0, \mathbf{I})$$

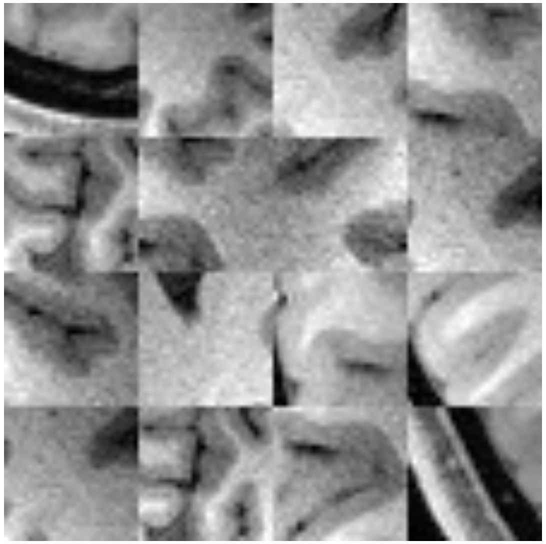
Similar to any latent variable model, e.g. PCA when conditional is linear



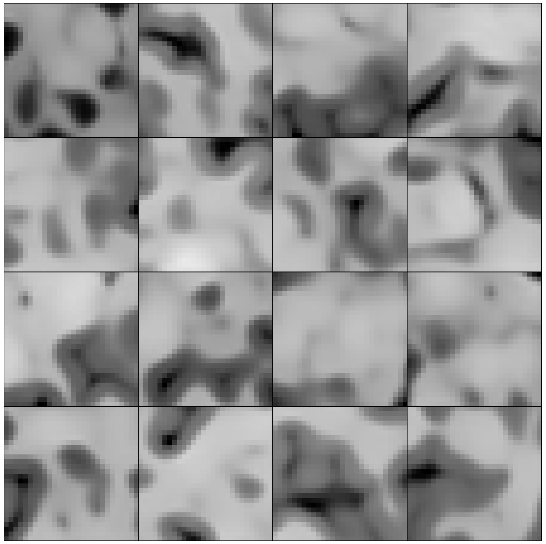
- Now we know about it
- We train the network with fully sampled high quality images
- Using Gaussian distributions for both likelihood and approximate sampling
- Prior distribution of healthy image patches

Samples from learned distributions of fully-sampled image patches

Real patches of 28x28

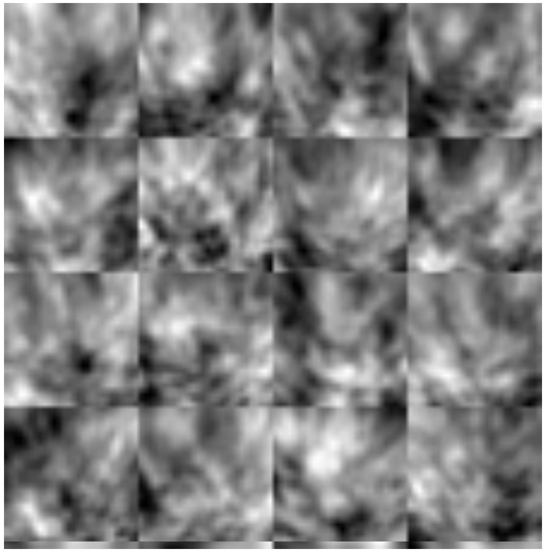


VAE Generated



60 components

PCA



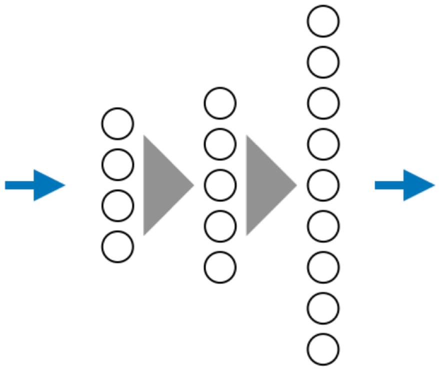
250 components

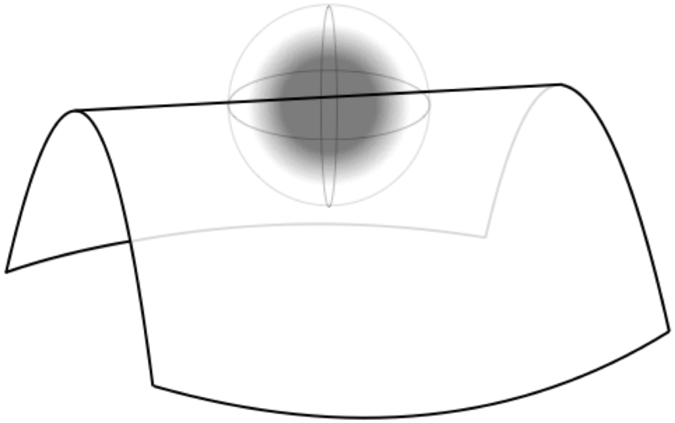
Back to the optimization problem

$$\mathbf{m}^* = \arg_{\mathbf{m}} \max p(\mathbf{m}|\mathbf{y}) = \arg_{\mathbf{m}} \max p(\mathbf{y}|\mathbf{m}) p(\mathbf{m})$$

Image that maximizes my current belief after observations

$$p(\mathbf{y}|\mathbf{m}) = \mathcal{N}(\mathbf{y}|E\mathbf{m}, \sigma_n)$$

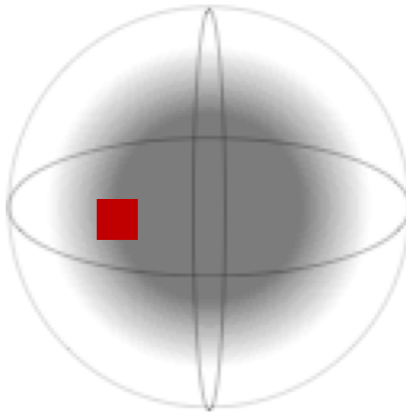




Prior is approximate

but differentiable

Optimization

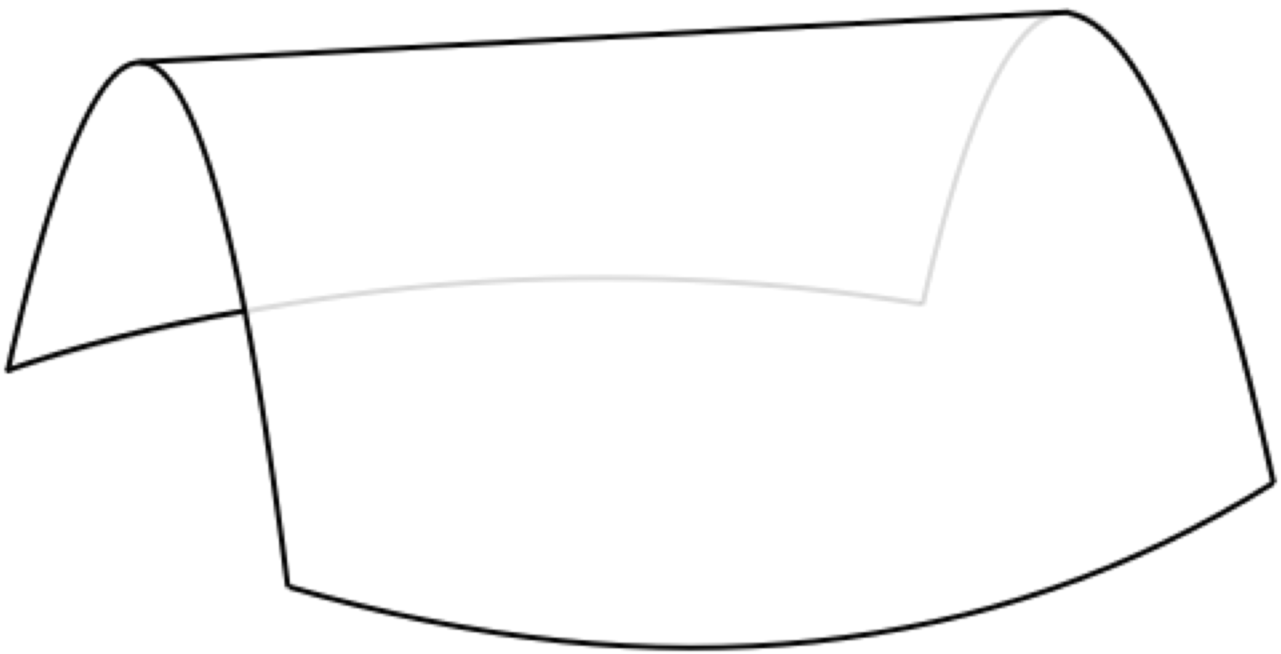


Approximate to step towards high prior area

$$\frac{d}{d\mathbf{m}} \ln p(\mathbf{m})$$

Project back to satisfying observations

$$\mathbf{m}^{t+1} = \mathbf{m}^t - E^H(E\mathbf{m}^t - \mathbf{y})$$



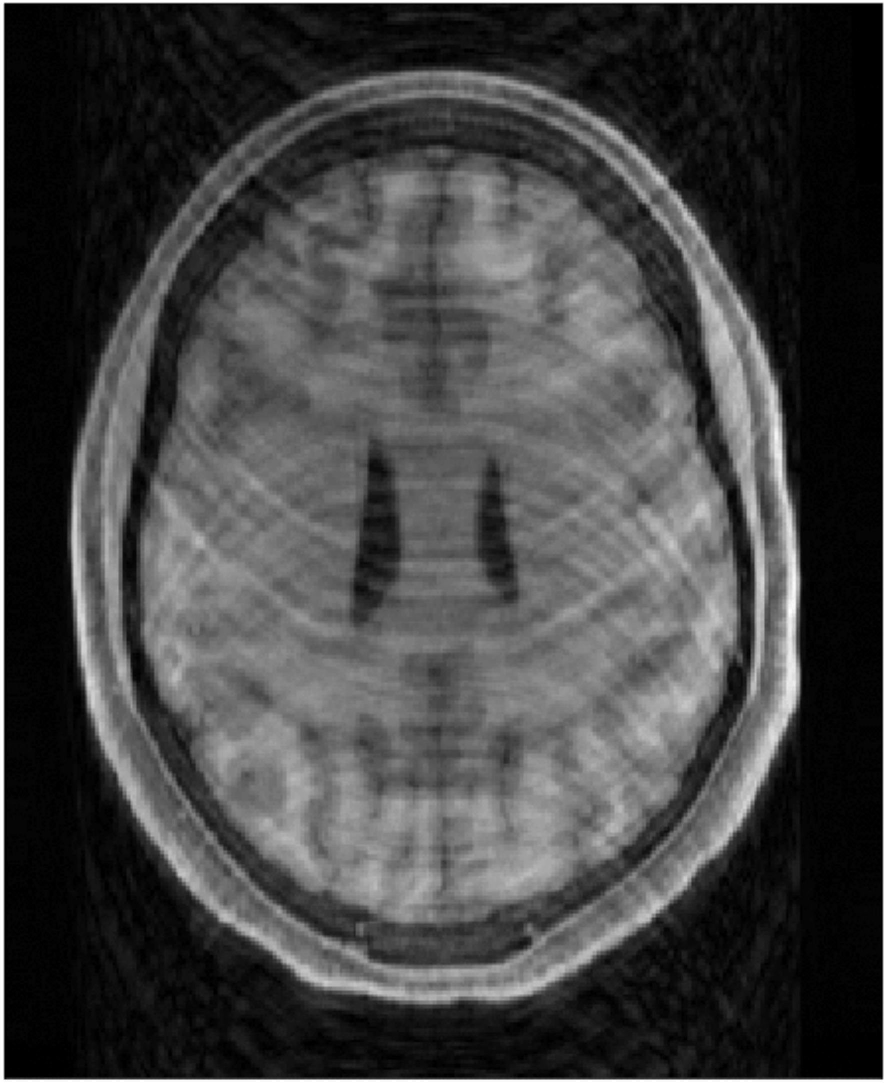
But we don’t have access to p(m)…

- We’ll take the gradient of the (evidence) lower bound

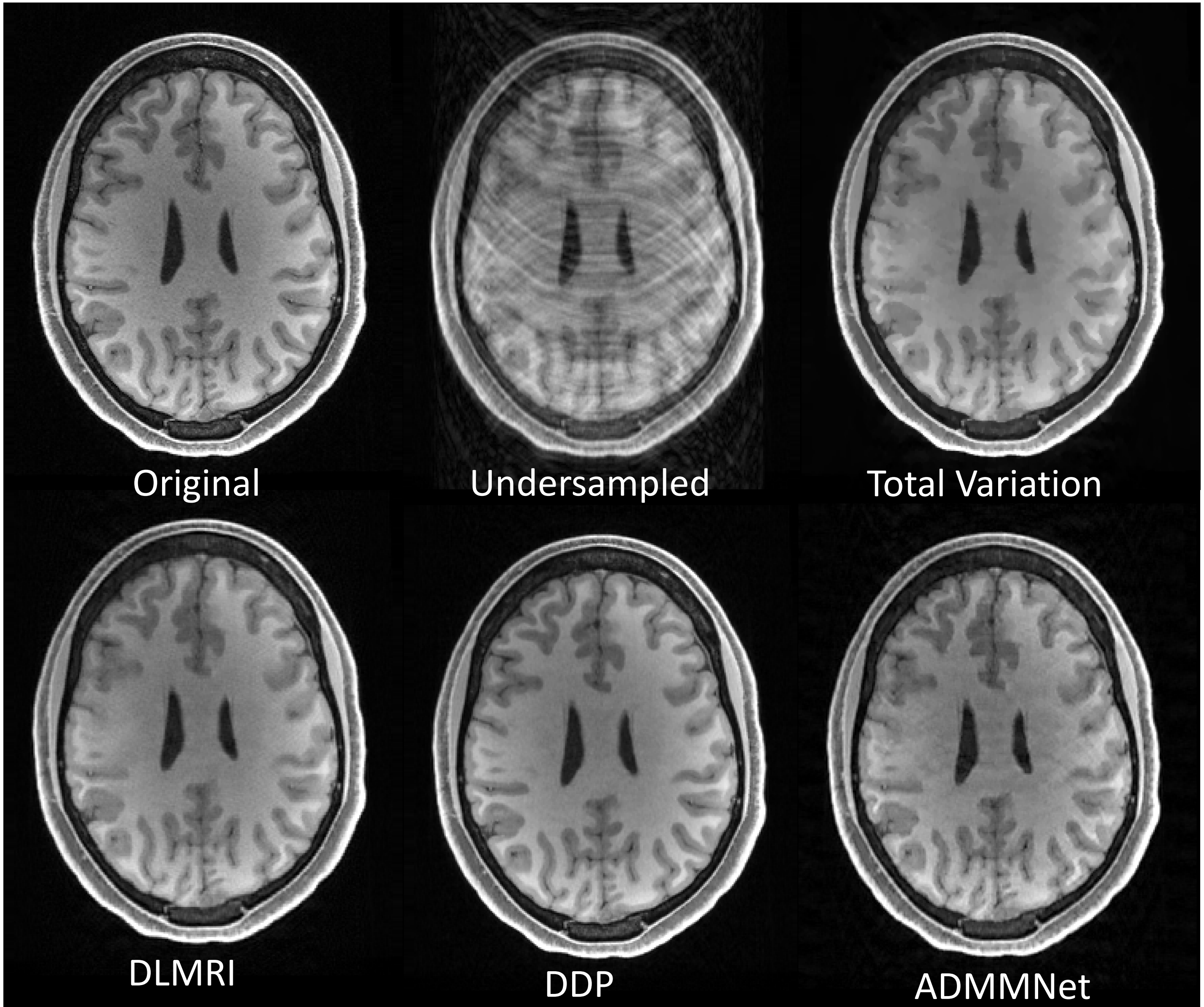
$$\ln p(\mathbf{m}) \geq \mathbb{E}_{\mathbf{z}\sim q}\left[\ln p(\mathbf{m}|\mathbf{z}, \mathbf{w}_\theta)\right] - D_{KL}\left[q(\mathbf{z}|\mathbf{m}, \mathbf{w}_\phi) \| p(\mathbf{z})\right]$$

$$\frac{d}{d\mathbf{m}} \ln p(\mathbf{m}) \approx \frac{d}{d\mathbf{m}} \left(\mathbb{E}_{\mathbf{z}\sim q}\left[\ln p(\mathbf{m}|\mathbf{z}, \mathbf{w}_\theta)\right] - D_{KL}\left[q(\mathbf{z}|\mathbf{m}, \mathbf{w}_\phi) \| p(\mathbf{z})\right]\right)$$

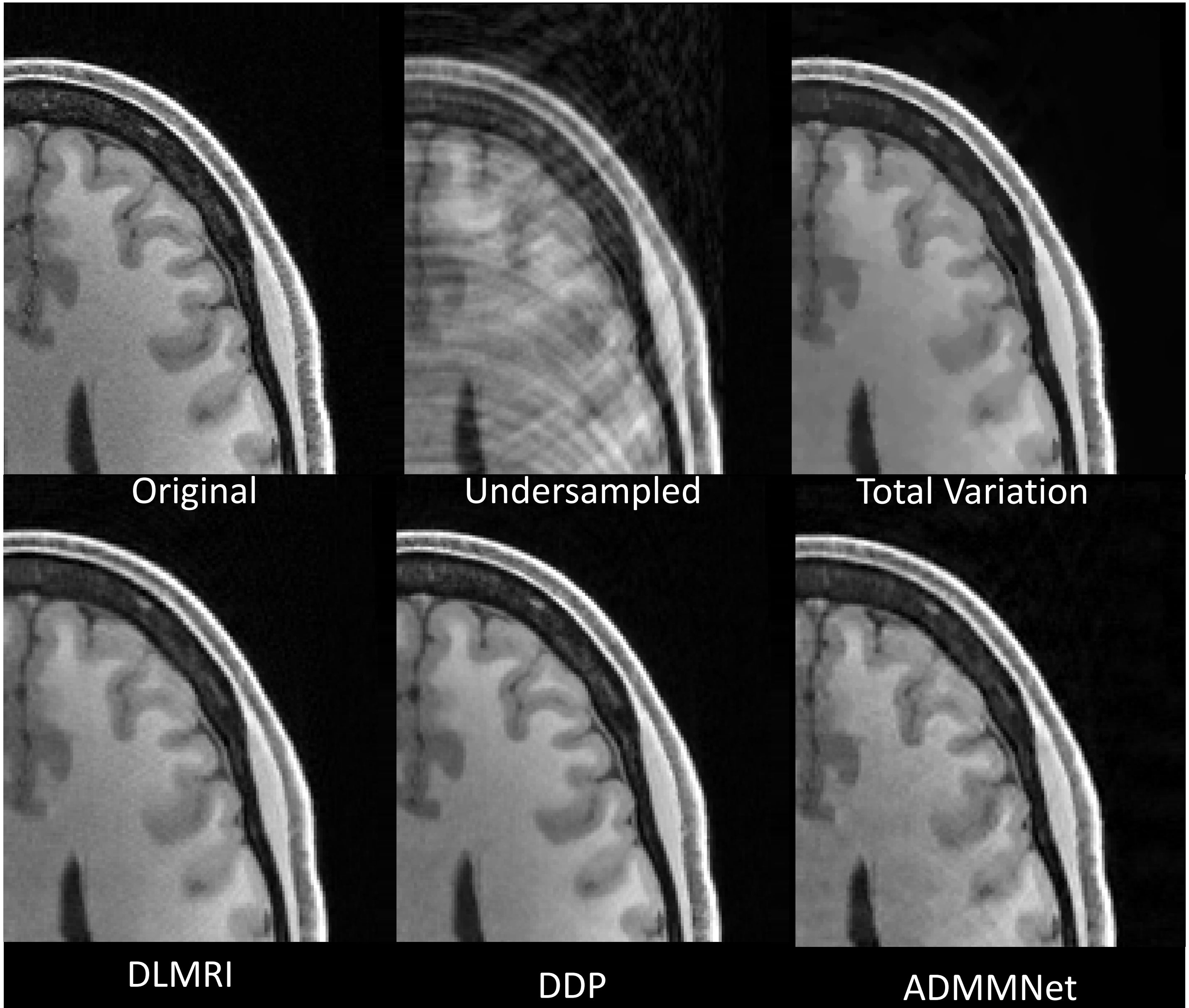
- Note that **z** also depends on **m**, it needs differentiation



Reconstructions with 1/3rd measurements



Reconstructions with 1/3rd measurements



Numerically

	R = 2 – ½ measurements		R=3 – 1/3rd measurements	
	RMSE	CNR	RMSE	CNR
Fully Sampled	-	0.43 (0.09)	-	0.43 (0.09)
Zero-filled	13.40 (0.84)	0.36 (0.08)	21.40 (1.12)	0.30 (0.07)
Total Variation	4.00 (0.51)	0.41 (0.10)	7.57 (0.71)	0.35 (0.01)
Dictionary Learning	4.65 (0.54)	0.41 (0.10)	7.41 (0.83)	0.36 (0.01)
ADMMNet	3.74 (0.40)	0.43 (0.10)	7.49 (0.47)	0.39 (0.01)
DDP	**2.77 (0.39)**	**0.43 (0.10)**	**4.29 (0.51)**	**0.43 (0.01)**

Average values (std) over images from 17 test subjects

Data not similar to training data used for prior

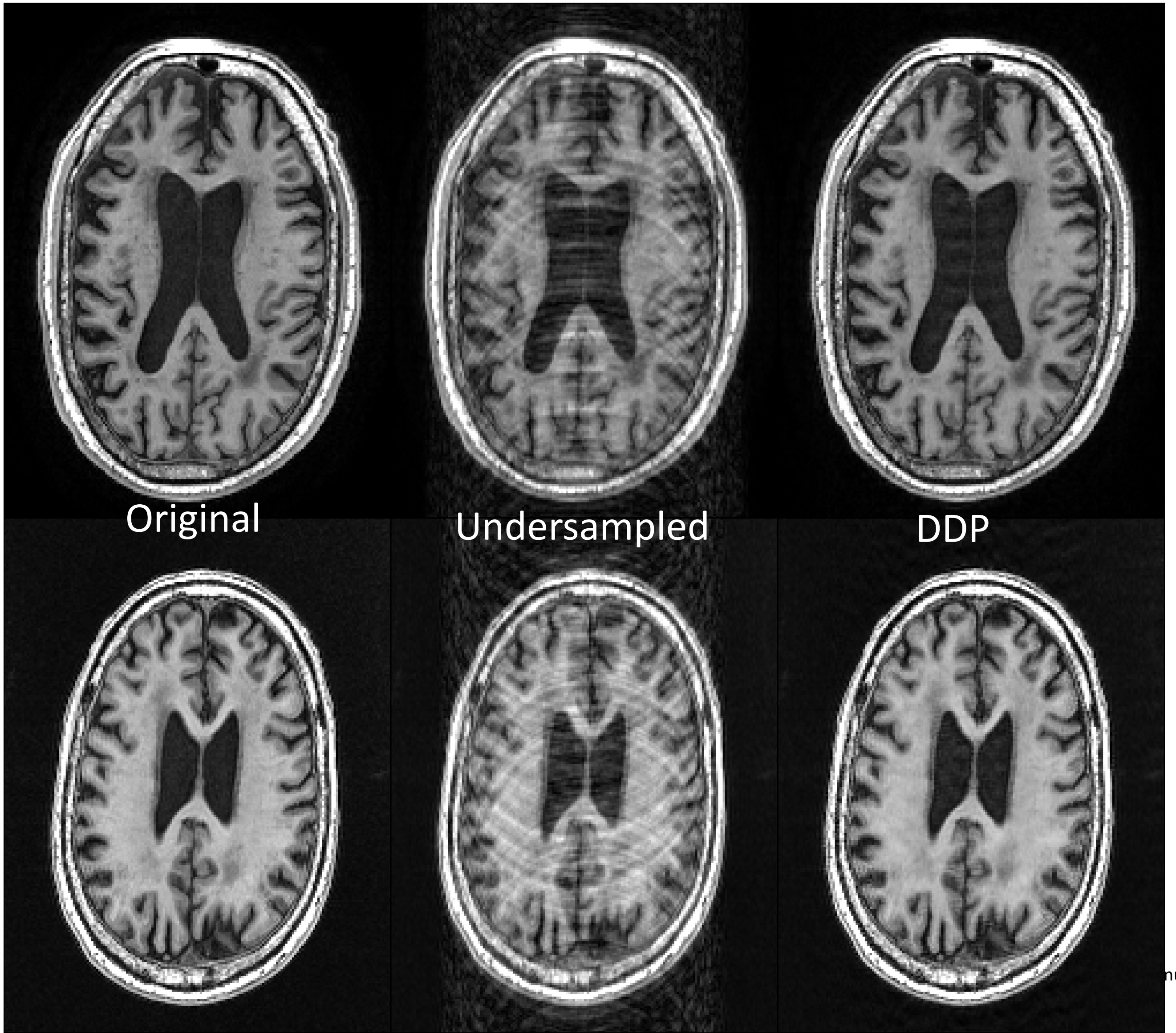


Image analysis with DL priors

- MR image reconstruction using deep density priors
  [K. Tezcan, C. Baumgartner and E. Konukoglu
  arXiv: 1711.11386]
- Solving Linear Inverse Problems Using GAN Priors: An Algorithm with Provable Guarantees
  [V. Shah and Chinmay Hedge, arXiv: 1802.08406]

Alternative with GAN

[V. Shah and C. Hedge, arXiv: 1802.08406]

Moving away from the probabilistic interpretation and seeing it as a constrained optimization problem

$$\mathbf{m}^* = \arg \min_{\mathbf{m} \in \mathcal{M}} \|\mathbf{y} - E\mathbf{m}\|_2^2$$

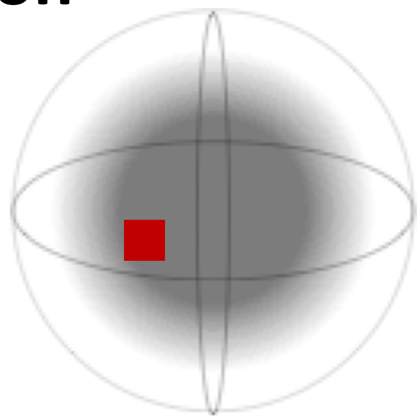
$\mathcal{M}$: space of plausible images

Iterative optimization with two steps:

$$\hat{\mathbf{m}}^t \leftarrow \mathbf{m}^t - \eta \frac{\partial}{\partial \mathbf{m}} \|\mathbf{y} - E\mathbf{m}^t\|_2^2$$: Gradient step towards the observation

$$\mathbf{m}^{t+1} \leftarrow G\left(\arg\min_{\mathbf{z}} \|\hat{\mathbf{m}}^t - G(\mathbf{z})\|\right)$$: Projection to data manifold
Best latent variable that matches the image at that iteration

Optimization

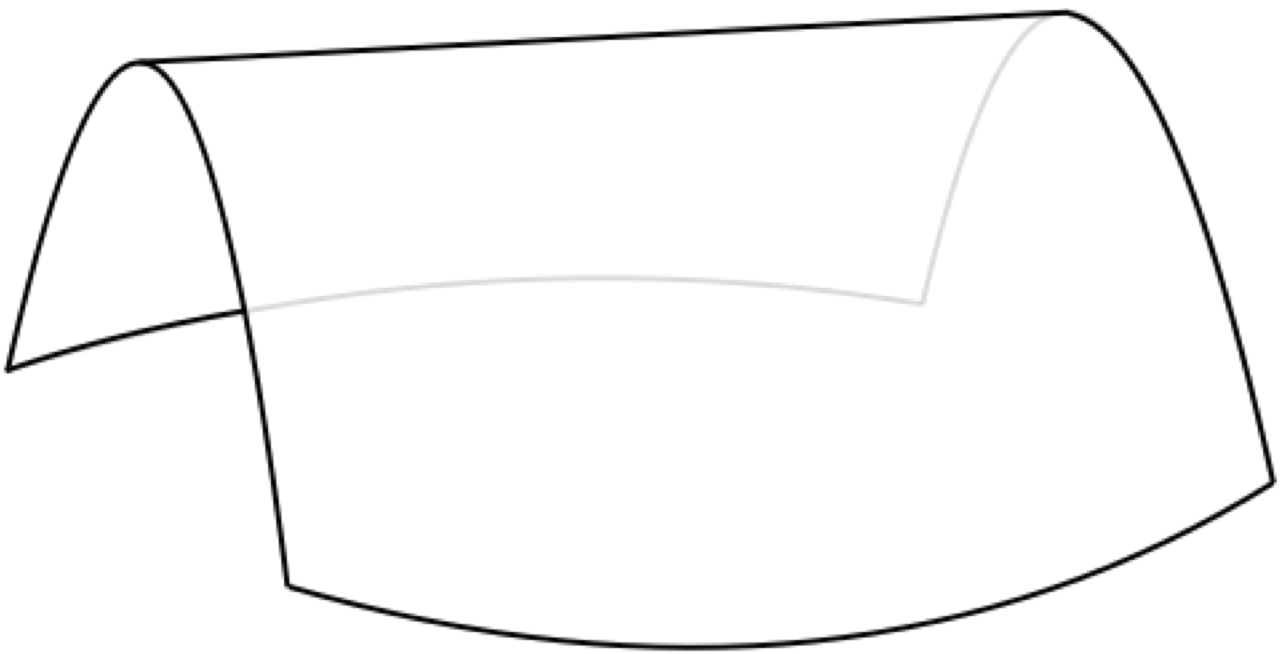


Project to data manifold with its separate optimization

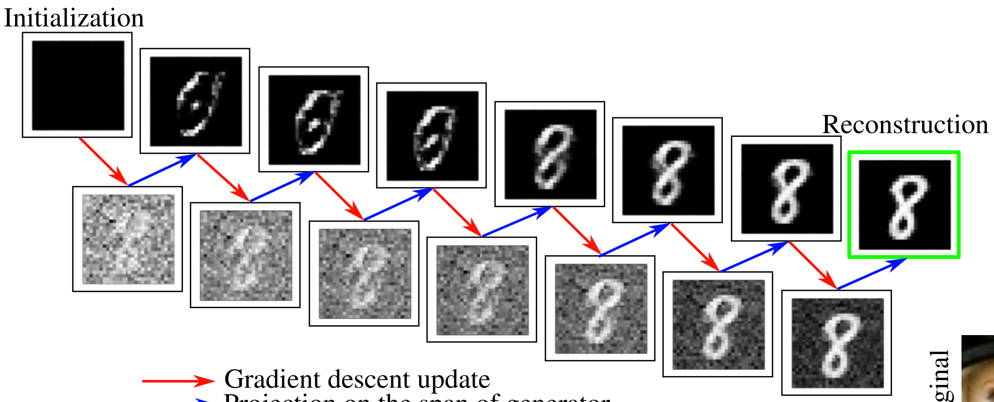
$$\mathbf{m}^{t+1} \leftarrow G\left(\arg\min_{\mathbf{z}} \|\hat{\mathbf{m}}^t - G(\mathbf{z})\|\right)$$

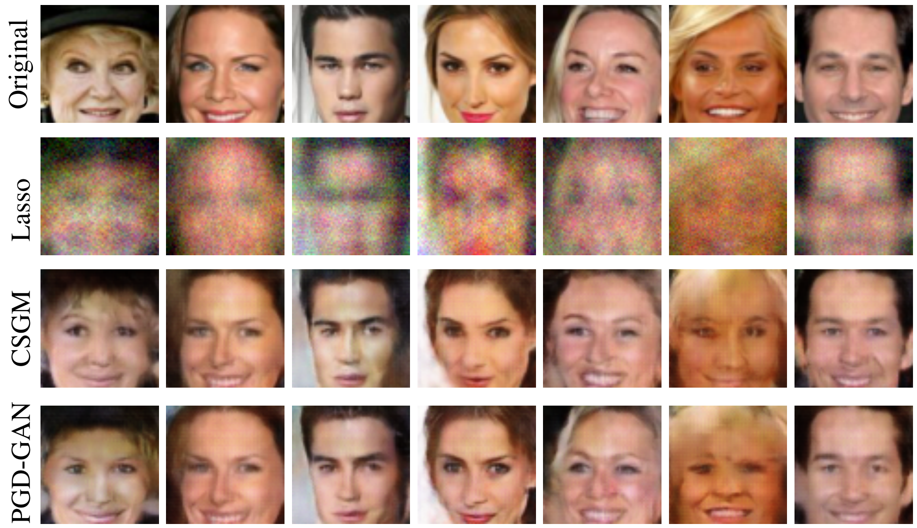
Gradient step towards observation

$$\hat{\mathbf{m}}^t \leftarrow \mathbf{m}^t - \eta \frac{\partial}{\partial \mathbf{m}} \|\mathbf{y} - E\mathbf{m}^t\|_2^2$$



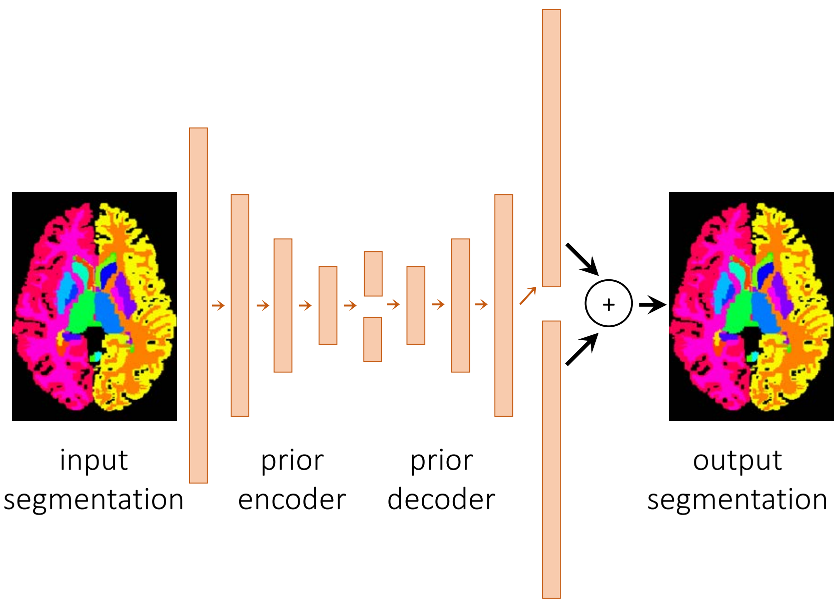
Examples from [V. Shah and C. Hedge, arXiv: 1802.08406]



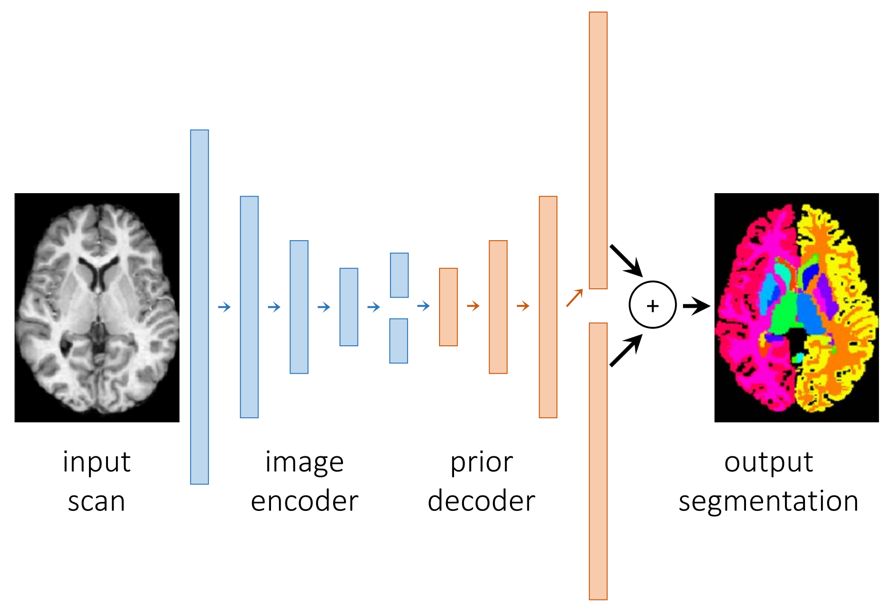


Prediction in latent space

- Anatomical Priors in Convolutional Networks for Unsupervised Biomedical Segmentation [A. Dalca, J. Guttag and M. Sabuncu, CVPR 2018]
- Prior model is used in a feed-forward fashion without data consistency



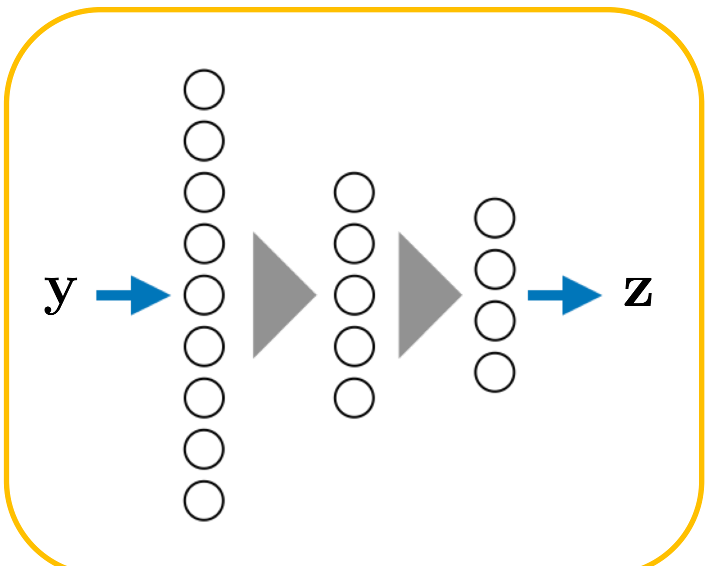
VAE prior model for brain segmentation



Directly mapping image to the latent space

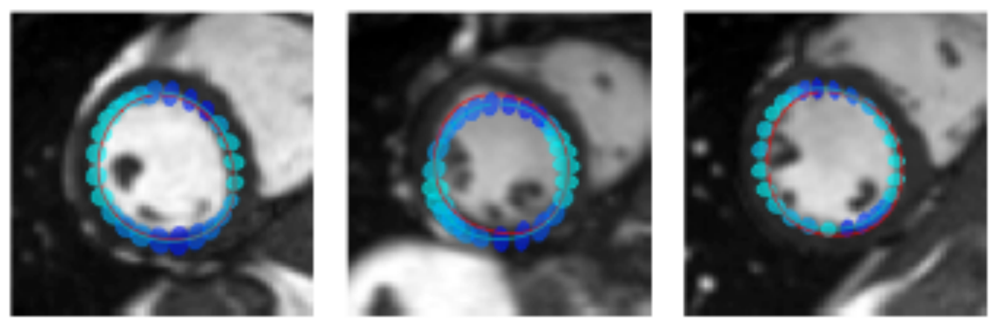
Related: Predicting latent space of a linear PCA model

- Integrating statistical prior knowledge into convolutional neural networks [Milletari et al., MICCAI 2017]
- Using a linear prior model instead of a non-linear DL prior
- Predicting in the latent space directly with a network
- Add uncertainties to it [Tothova et al. arXiv:1807.11272]



PCA model learned from the training set

$$\mathbf{m} = \mathbf{U}\mathbf{z} + \mu_{\mathbf{m}} + \mathbf{s}$$

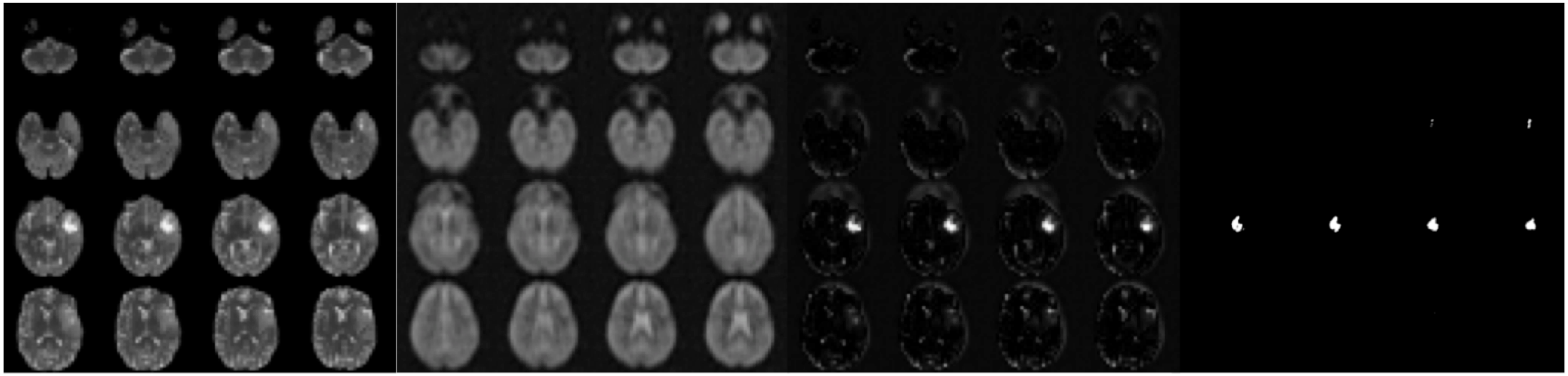


[Image from Tothova et al. arXiv:1807.11272]

Also related: Outlier detection in medical images

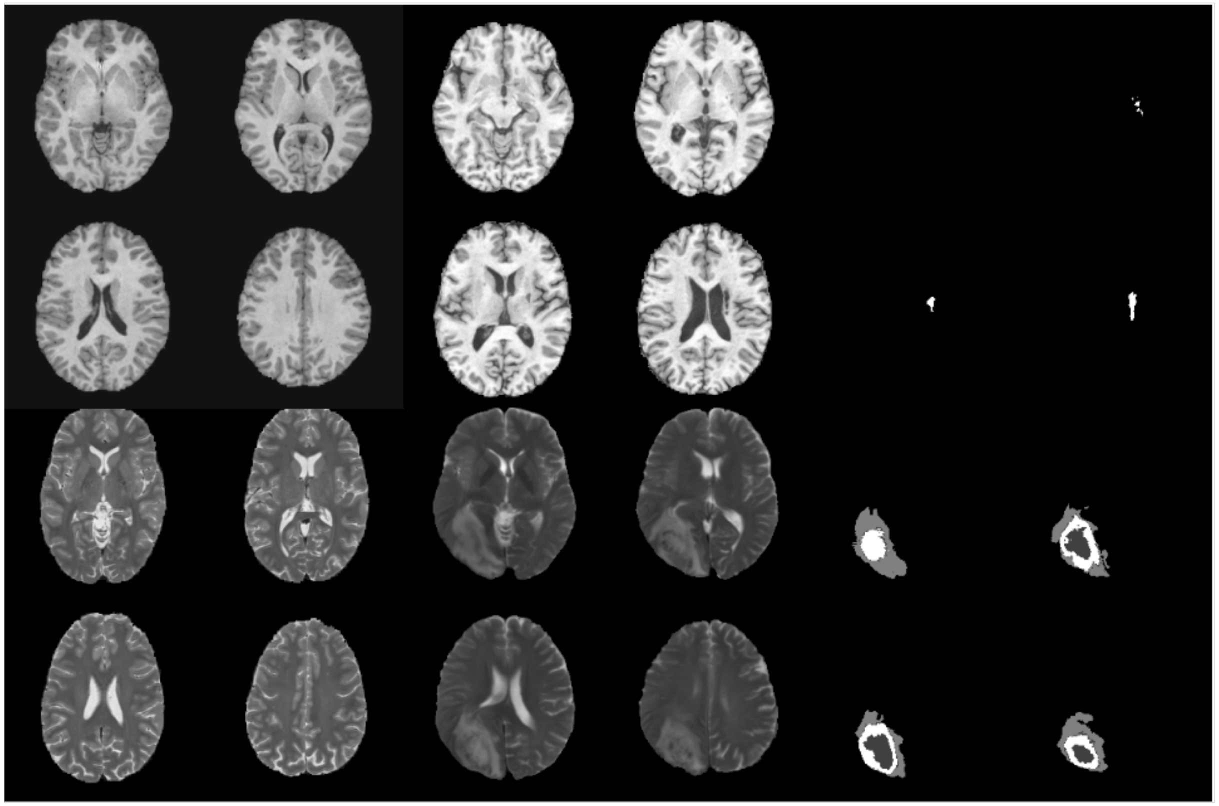
- Anomaly detection with GANs [T. Schlegl et al. IPMI 2017]
- Anomaly detection with Auto-encoder methods
  [X. Chen and E. Konukoglu, MIDL 2018]

Image | Reconstruction with prior | Reconstruction error | Ground truth segmentation



Really difficult problem for DL

- Real life problem that DL methods are not good at yet [X. Chen et al. arXiv:1806.05452]

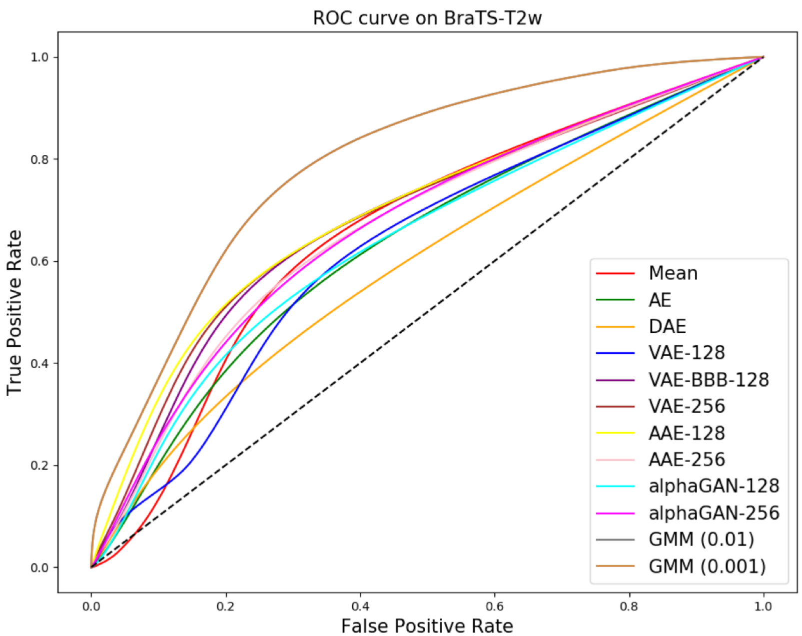


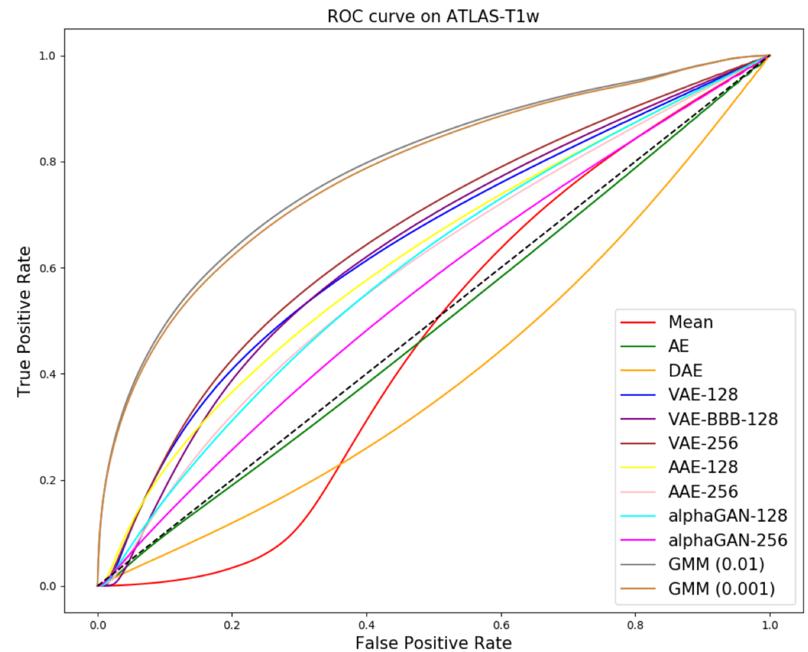
- Variety of lesions and their appearance
- Domain shift – part of the real problem

Evaluated

- AE
- Denoising AE
- VAE
- Adversarial AE
- α- GAN
- Atlas-based detection

Guess what performed the best?





Atlas-based outlier detection linear registration and E-M based GMM

ETH zürich

Thank you

Questions?

ender.konukoglu@vision.ee.ethz.ch