Imperial College London

TOTAL

Stochastic Simulation with Generative Adversarial Networks

Lukas Mosser, Olivier Dubrule, Martin J. Blunt

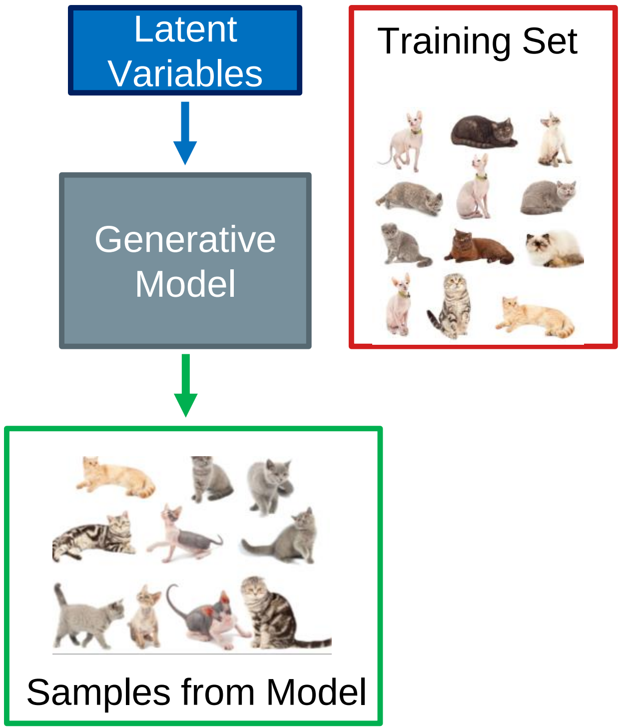
lukas.mosser15@imperial.ac.uk, o.dubrule@imperial.ac.uk, m.blunt@imperial.ac.uk

(Deep) Generative Methods

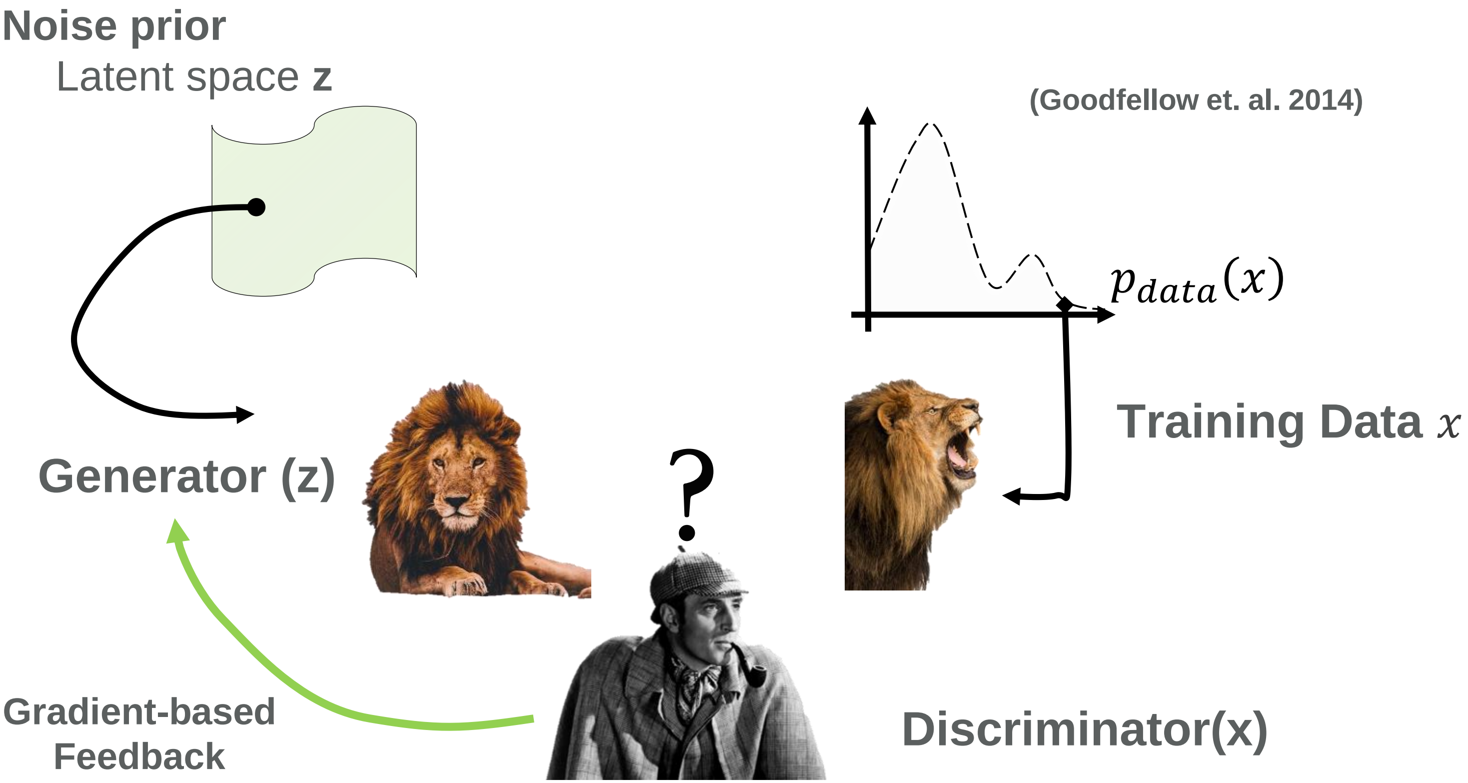
- Task: Draw (new) samples from unknown density given a set of samples

Main Problem: How to find the generative model?

- Generative Adversarial Networks (GAN)
  - Two competing Neural Networks
- Variational Autoencoders (VAE)
  - Bayesian Graphical Model of data distribution
- Autoregression (Pixel-CNN)
  - Conditional Distribution on every sample
- Many More ...



Generative Adversarial Networks – Toy Example



Generative Adversarial Networks – Training

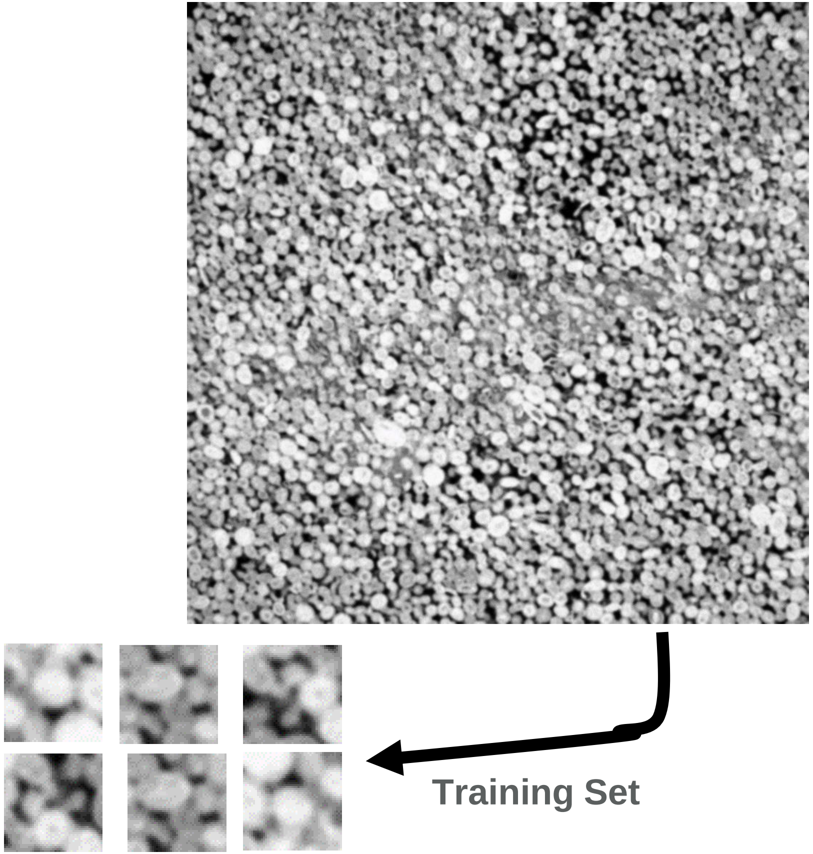
- **Requirements:**
  - **Training Set of data**
  - **Generator – creates samples G(z)** $\mathbf{z} \sim \mathcal{N}(0,1)^{d\times1\times1\times1} \quad G_\theta : \mathbf{z} \rightarrow \mathbb{R}^{1\times64\times64\times64}$
  - **Discriminator – evaluates samples** $D_\omega : \mathbb{R}^{1\times64\times64\times64} \rightarrow [0,1]$
  - **Cost function:** $\min_\theta \max_\omega \{\mathbb{E}_{\mathbf{x}\sim p_{data}}[log\ D_\omega(\mathbf{x})] + \mathbb{E}_{\mathbf{x}\sim p_{\mathbf{z}}}[log\ 1 - D_\omega(G_\theta(\mathbf{z}))]\}$

- **GAN training – two step procedure in supervised way**
  - **Discriminator training step – Generator fixed**
    - **Train on real data samples**
    - **Train on fake samples**
  - **Generator training step – Discriminator fixed**
    - **Push generator towards “real” images**

Ketton Limestone Dataset and Preprocessing

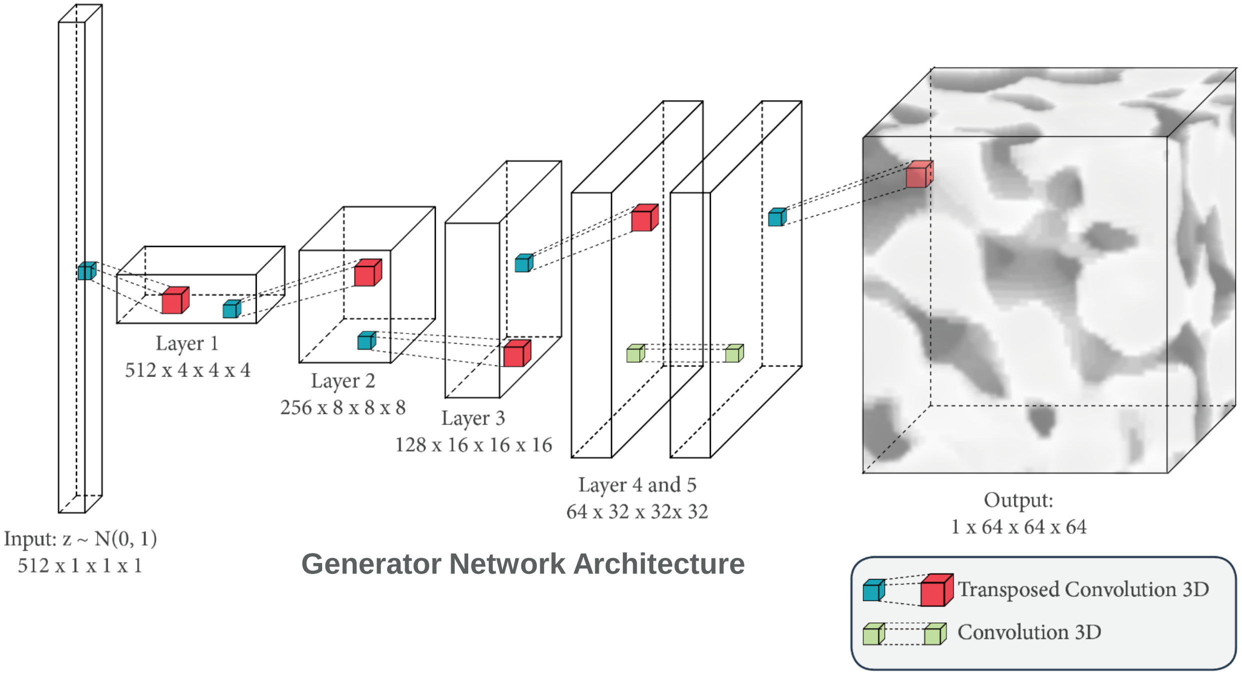
- **Oolitic Limestone**
- **Intergranular pores**
- **Intragranular Micro-Porosity**
- **Ellipsoidal grains**
- **99% Calcite**
- **Image Size:**
  - $\mathbf{900^\wedge 3}$ voxels @ 26.7 μm

Extract Non-Overlapping Training Images ($\boldsymbol{64^3\ voxels}$)



Network Architecture - 3D Convolutional Network

Represent G(z) and D(x) as deep neural networks:



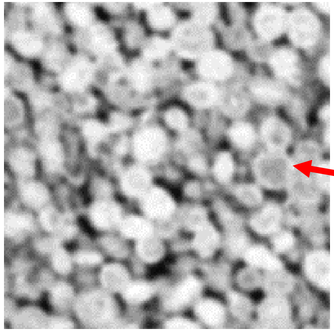
Generator Network Architecture

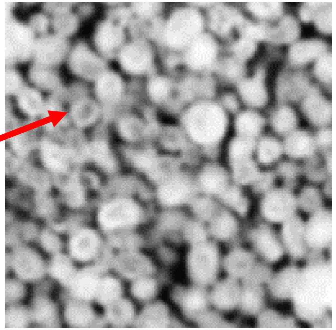
Discriminator: Binary Classification Network -> Real / Fake

Reconstruction Quality – Unconditional Simulation

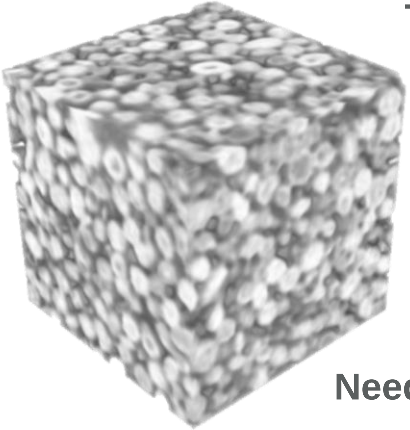
Ketton Training Image

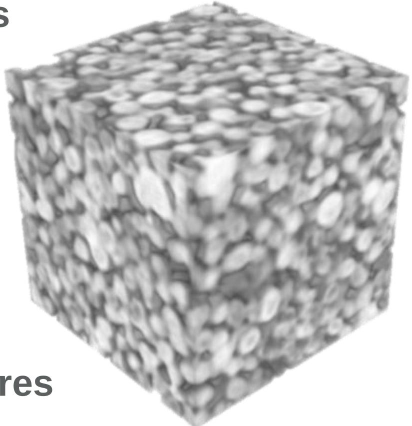
GAN generated sample





Intergranular Porosity
Moldic Features
Micro-Porosity





Training Time: 8 hours

Generation: 5 sec.

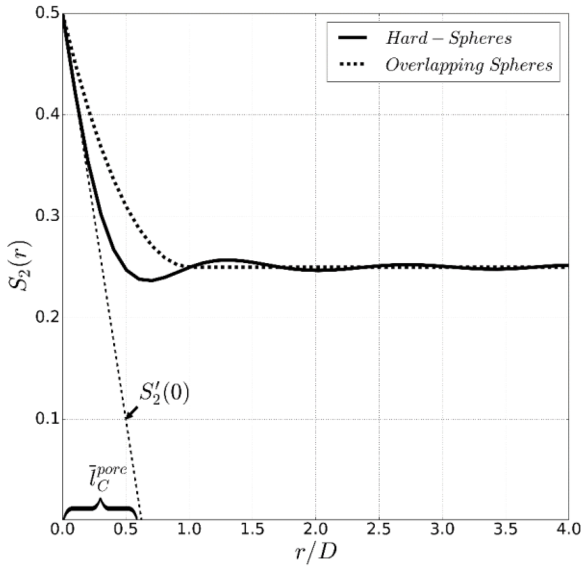
High visual quality
Needs quantitative measures

Reconstruction Quality Criteria

Statistical Properties

- **Two-Point Probability Function $S_2(r)$**
  - » **Radial Average / Directional**

$$S_2(\mathbf{r}) = \mathbf{P}(\mathbf{x} \in P, \mathbf{x} + \mathbf{r} \in P) \;\; for \;\; \mathbf{x}, \mathbf{r} \in \mathbb{R}^3$$



Minkowski Functionals

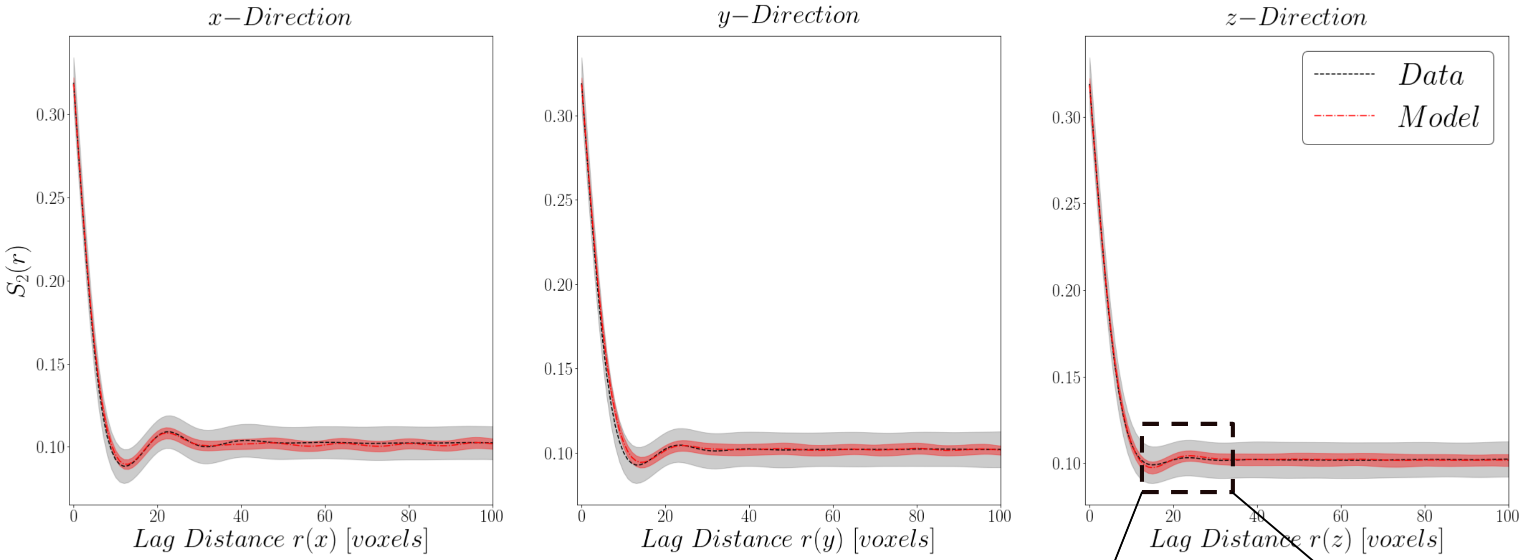
- **Porosity ϕ**
- **Specific Surface Area S_v**
- **Integral of Mean Curvature**
- **Specific Euler Characteristic χ_v**
- **Compute as function of image gray-level => Characteristic Curves**

Flow Properties: Solve Stokes flow in pore domain

- **Permeability + Velocity Distributions**

$$\nabla \cdot \mathbf{v} = 0$$
$$\mu \nabla^2 \mathbf{v} = \nabla p$$

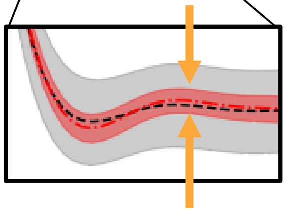
Ketton Comparison – Directional $S_2(r)$



Isotropic Covariance

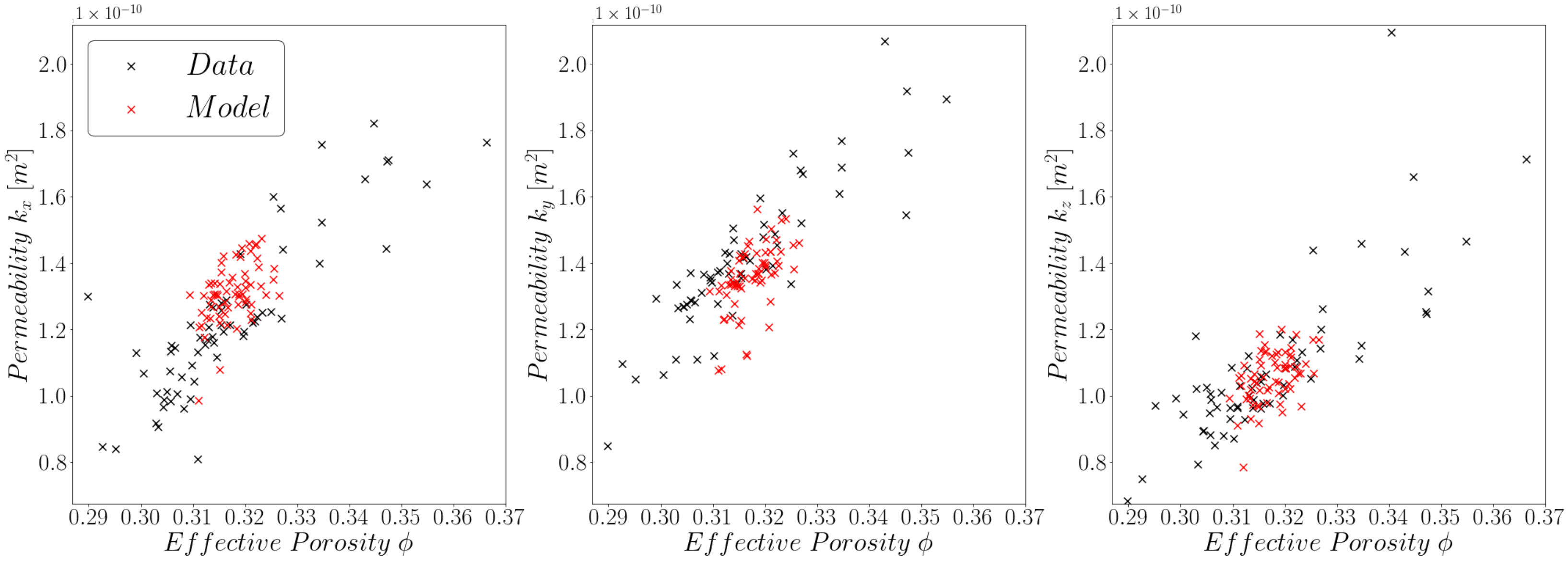
Pronounced Oscillations -> “Hole-Effect”

- **Captured by GAN model**



Smaller Variance of GAN model

Ketton Comparison – Permeability



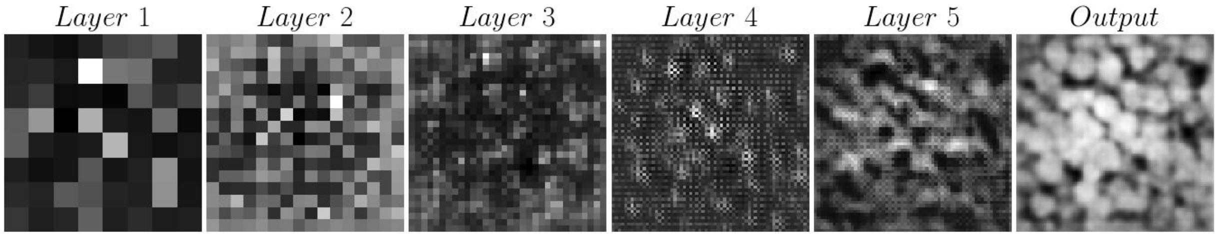
Isotropic Permeability

Range of effective (flowing) porosity: Data (0.29- 0.37) GAN (0.31-0.33)

Same order of magnitude and $\bar{\bar{k}} - \phi$ relationship

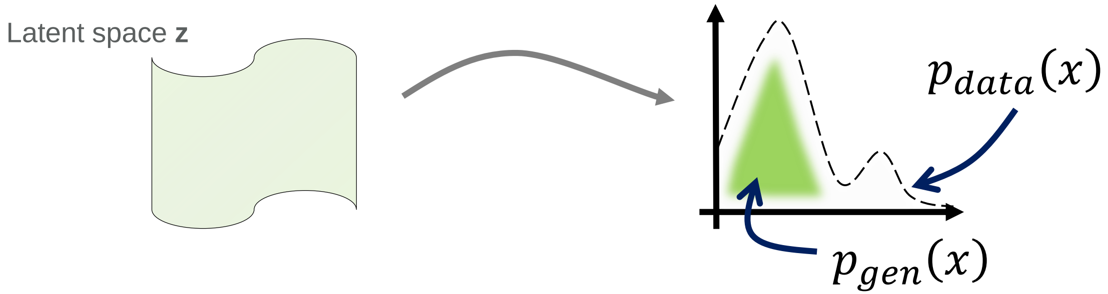
Opening the GAN black box

What does the Generator learn?



Multi-scale Representation of pore space

Smaller Variance in GAN generated samples: Why?

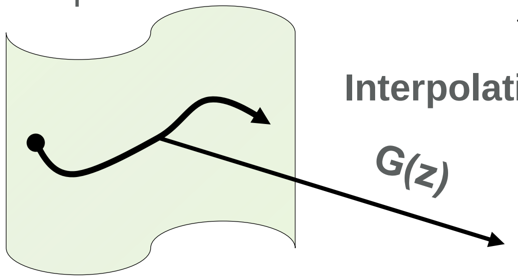


Generator can miss modes of the data distribution -> Mode-Collapse

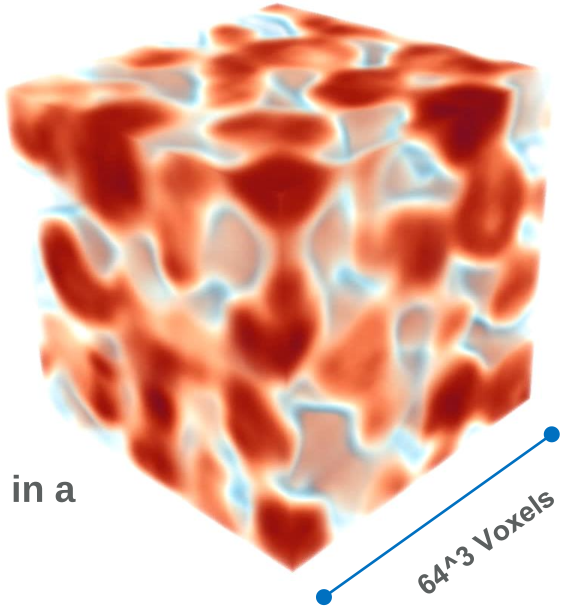
Latent Space Interpolation

Latent space **z**

$$z^* = \beta\, z_{start} + (1-\beta)\, z_{end}, \beta \in [0,1]$$



Interpolation path visualization

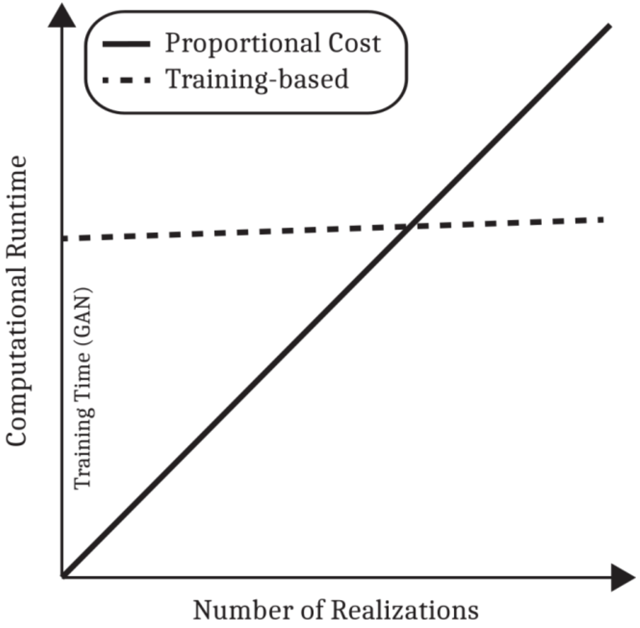


Interpolation in latent space:

Shows that generator has learned a meaningful representation in a lower dimensional space!

Computational Effort

Authors	Method	Size [voxels^3]	Run time ($\times$ 1) (h)
Computational run time comparison			
Pant (2016)	Simulated annealing	300^3	22–47
Tahmasebi et al. (2017)	Patch-based	$1000^2 \times 300$	0.1
Okabe and Blunt (2004)	MPS	150^3	12
Current work	GAN	450^3	8



Main Computation cost training:
Amortizes with number of samples due to low per sample cost / runtime

Image Inpainting (Yeh et al. 2016)

Task: Restore missing details given a corrupted / masked image $M \cdot \tilde{x}$

Use a generative model G(z) to find missing details, conditional to given information.

Contextual Loss: $L_{content} = \lambda \left|\left| M \cdot G(z) - M \cdot \tilde{x} \right|\right|_2$ → Corresponds to likelihood

Perceptual Loss: $L_{perc} = log(1 - D(G(z)))$ → Regularization for prior
Stay close to "real" images

Optimize loss by gradient descent on latent vector z

$M \cdot \tilde{x}$ | Human Artist | L_2 $Loss$ | $L_{content} + L_{perc}$

Credit: Kyle Kastner

Conditioning – Pore Scale Example

Two-dimensional data at pore-scale more abundant e.g. thin-sections
Combine 3D generative model G(z) with 2D conditioning data

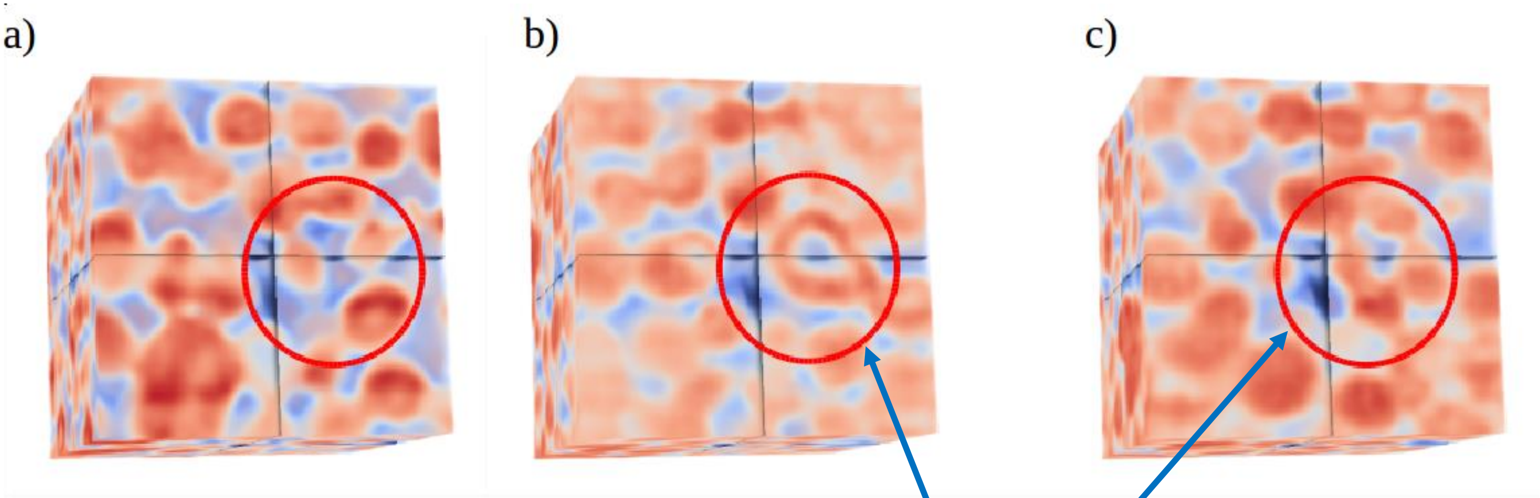
Generative Model: Ketton Limestone GAN (Part 1)
Mask: Three orthogonal cross-sections, honor 2D data in a 3D image

Contextual Loss: $L_{content} = \lambda \left|\left| M \cdot G(z) - M \cdot \tilde{x} \right|\right|_2$ **on orthogonal cross-sections**
Perceptual Loss: $L_{perc} = \log(1 - D(G(z)))$ **on whole volumetric generated image G(z)**

$$L_{Total} = \lambda\, L_{content} + L_{perceptual}$$

Optimize Total Loss, by modifying latent vector (GAN parameters fixed)
-> Many local minima at error threshold -> stochastic volumes that honor 2D data

Conditioning – Pore Scale Example



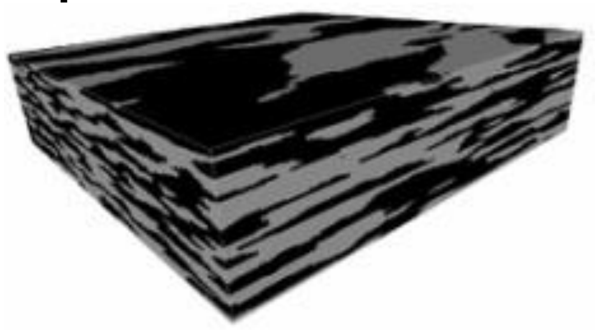
Conditioning Data
Ground Truth Volume

Stochastic Sample 1
Conditioned to Data

Stochastic Sample 2
Conditioned to Data

Same 2D conditioning data leads to varied realizations in 3D

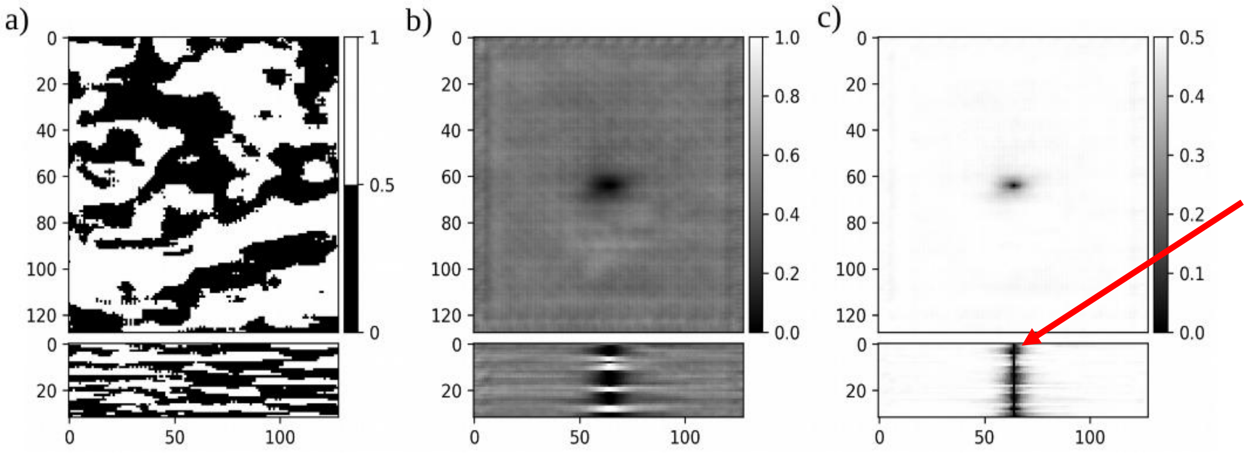
Conditioning – Reservoir Scale Example



Maules Creek Training Image (Credit G. Mariethoz)

Pre-trained 3D-Generative Adversarial Network

Condition to single well (1D conditioning) from ground truth data:



No Variance at Well

Single Realization **Mean (N=1000)** **Standard Dev. (N=1000)**

Conclusions:

Generative Adversarial Networks are:

- **Parametric – Latent Vector**
- **Differentiable – Allow for optimization**
- **Learned from training examples**

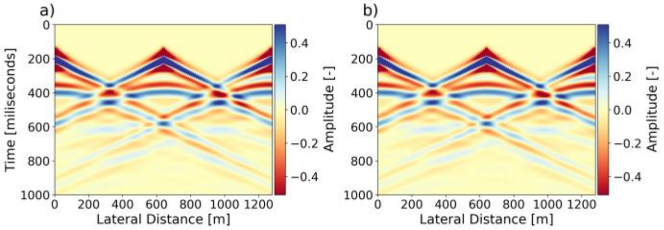
That allow continuous reparametrizations of geological models.

- **Can be conditioned to existing grid-block scale data.**

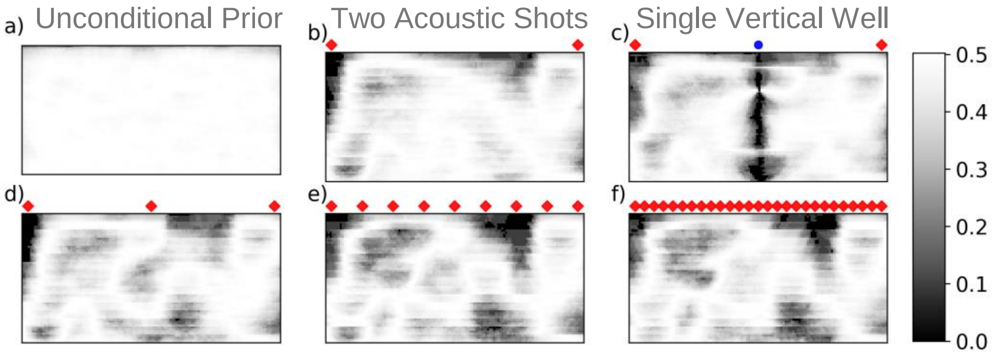
Possibly very useful for solving stochastic inverse problems

Main Idea: Represent prior with a (deep) generative model

arXiv preprint arXiv:1806.03720



Ground Truth Posterior Sample



Imperial College London

TOTAL

Thank you!

References

Reconstruction of three-dimensional porous media using generative adversarial neural networks. ***Physical Review E*****, *96*(4), 043309**,
Mosser, L., Dubrule, O., & Blunt, M. J. (2017).

Stochastic reconstruction of an oolitic limestone by generative adversarial networks. ***Transport in Porous Media*****, 1-23,**
Mosser, L., Dubrule, O., & Blunt, M. J. (2017).

Conditioning of Generative Adversarial Networks for Pore and Reservoir Scale Models, **80th EAGE Conference**,
Mosser L., Dubrule, O., & Blunt, M. J. (2018).

Stochastic seismic waveform inversion using generative adversarial networks as a geological prior. ***arXiv preprint arXiv:1806.03720***,
Mosser, L., Dubrule, O., & Blunt, M. J. (2018).