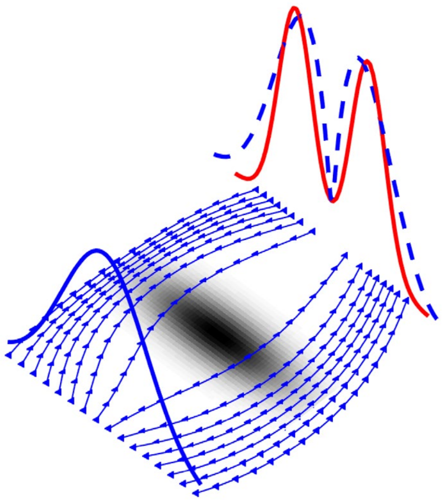
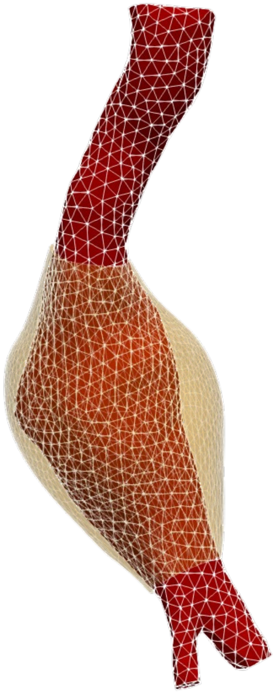


Geometric Deep Learning

From Learning ODE Dynamics towards Graph Neural Diffusion



Christoph Brune
Applied Analysis, SACS, Applied Mathematics
University of Twente, NL
c.brune@utwente.nl

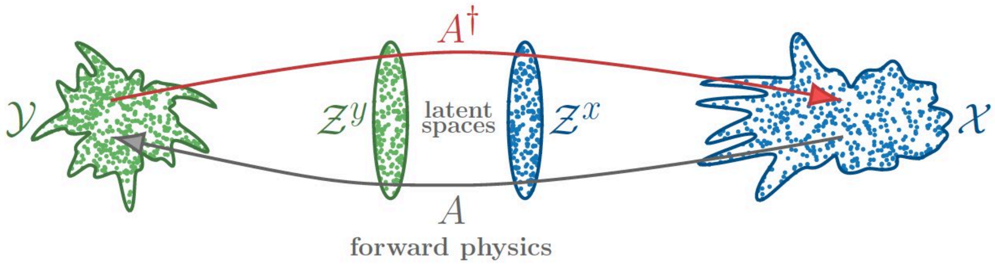
LMS-ICMS Workshop, Bath, July 26-30, 2021

Contributors: **Yoeri Boink, Leonie Zeune**, Srirang Manohar (UT)
Manu Kalia, Hil Meijer (UT), Nathan Kutz, Steven Brunton (Washington)
Julian Suk, Jelmer Wolterink (UT), Pim de Haan, Max Welling (Amsterdam)

OUTLINE

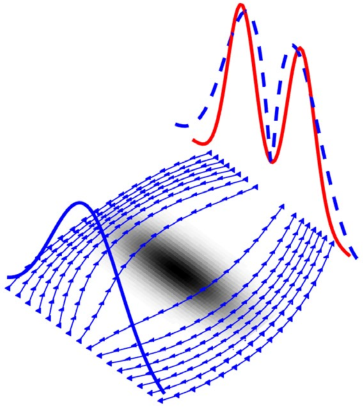
Inverse Problems and Deep Learning

+ Learned SVD via hybrid autoencoding



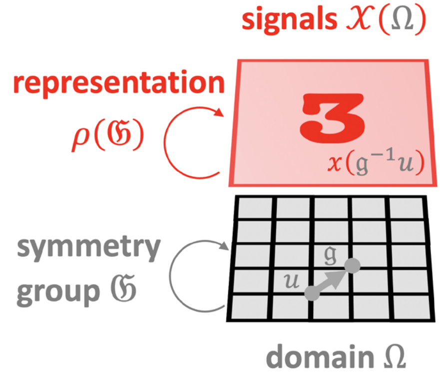
Deep Learning and Dynamics

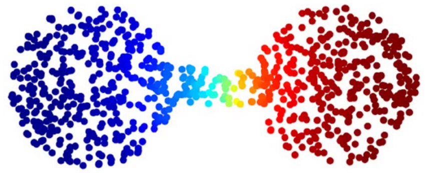
+ Optimal control, mean field games, neural SDEs
+ Normal form autoencoding for parameter dependent dynamics



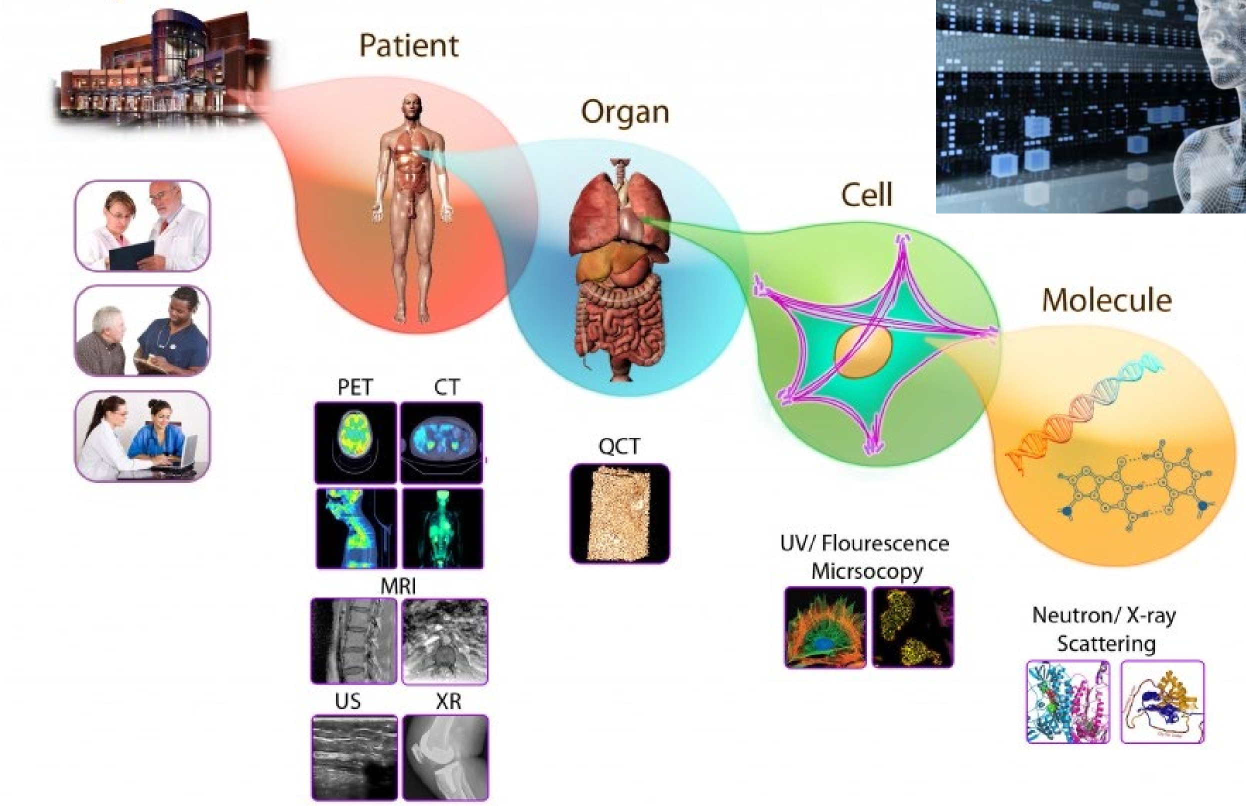
Geometric Deep Learning

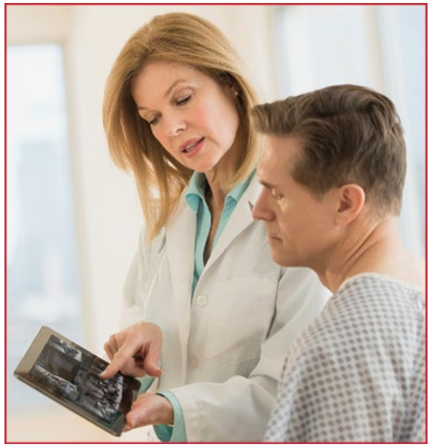
+ Geometric priors, Gauge mesh CNNs





Deep Learning and Inverse Problems in Biomedical Imaging





Multimodality, multiscale, multidimension, big data, mobile…

Deep Learning in Medical Imaging: Overview and Future Promise of an Exciting New Technique, IEEE Trans Med Imag, 2016

INVERSE PROBLEMS IN IMAGING

- Given measurements of the form
$$y^\delta = Ax + \eta^\delta$$
with x, y in appropriate Banach spaces and A (linear) operator between these spaces.

- Noise distribution η^δ known

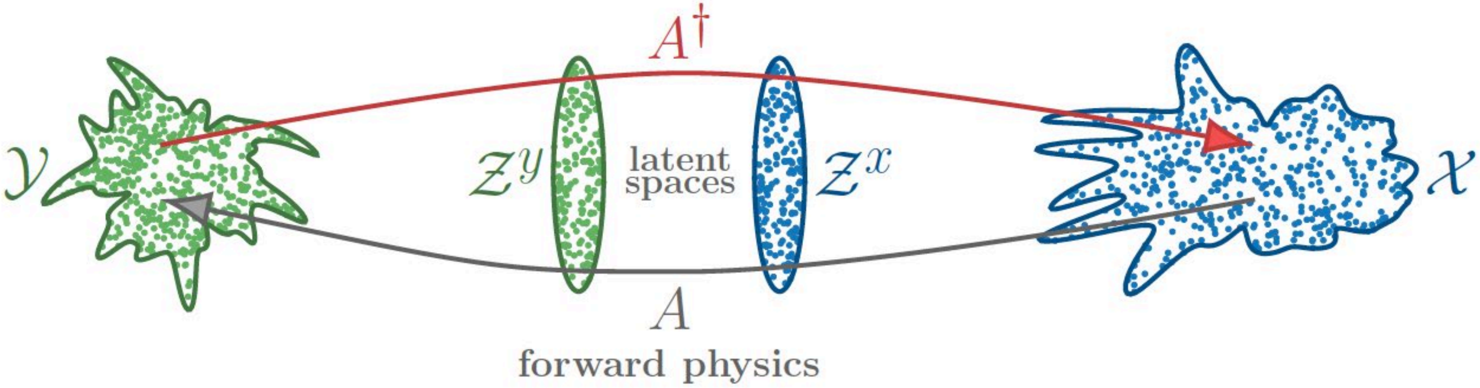
- If pseudo-inverse $A^\dagger$ available, it is typically **ill-posed** (following Hadamard)

- Example: A Radon transform (CT)

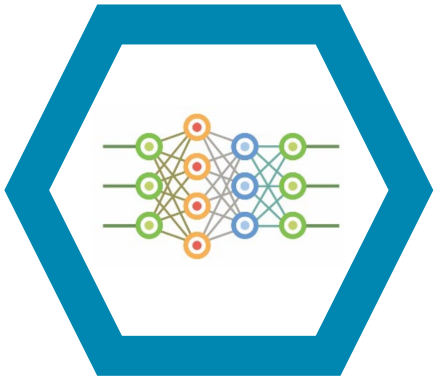
- **Classical solution**: Local Tikhonov regularization
$$x_\alpha = (A^*A + \alpha \mathrm{Id})^{-1} A^* y^\delta = V \underbrace{\left(S^2 + \alpha \mathrm{Id}\right)^{-1} S}_{S_\alpha^{-1}} U^* y^\delta$$

HOW TO LEARN A MEANINGFUL MAP?

HOW CAN WE CHARACTERIZE INFORMATIVE MAPS FOR INVERSE PROBLEMS AND DYNAMICS?



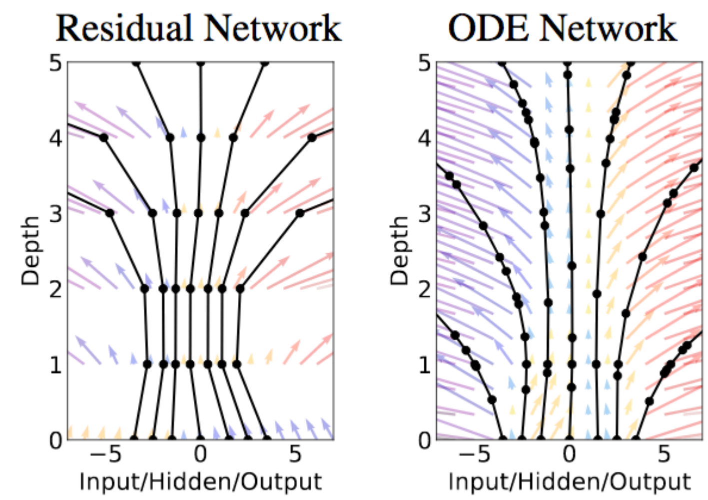
- Well-posed (existence, uniqueness, stability)?
- Robust inversion / parameter estimation?
- Structure preserving?
- Understandable decomposition?
- Expressive?
- Generalizable?



$$f(x) := \tau\left(W_L \tau\left(W_{L-1} \ldots \tau\left(W_1 x\right) \ldots\right)\right)$$

τ activation function (nonlinearity) W_i weight matrices (linear)

VARIATIONAL METHODS AND DEEP LEARNING



Deep Residual Neural Networks
are connected to
Partial Differential Equations

Variational methods	Norms nonconvex	Differential operators	Scale-space, Harmonic analysis	Regularization theory	Inverse problems	Vector fields, multimodality	Time dependent modeling
Deep networks	Activation functions ReLU, sigmoid	Convolutions, kernel methods	Scattering networks, Barron space	generalization properties	AEs GANs	network fusion transfer learning	Residual skip conn. flow-induced

Chen, Pock - Trainable Nonlinear Reaction Diffusion, 2016

Mallat - Understanding deep convolutional networks, 2016

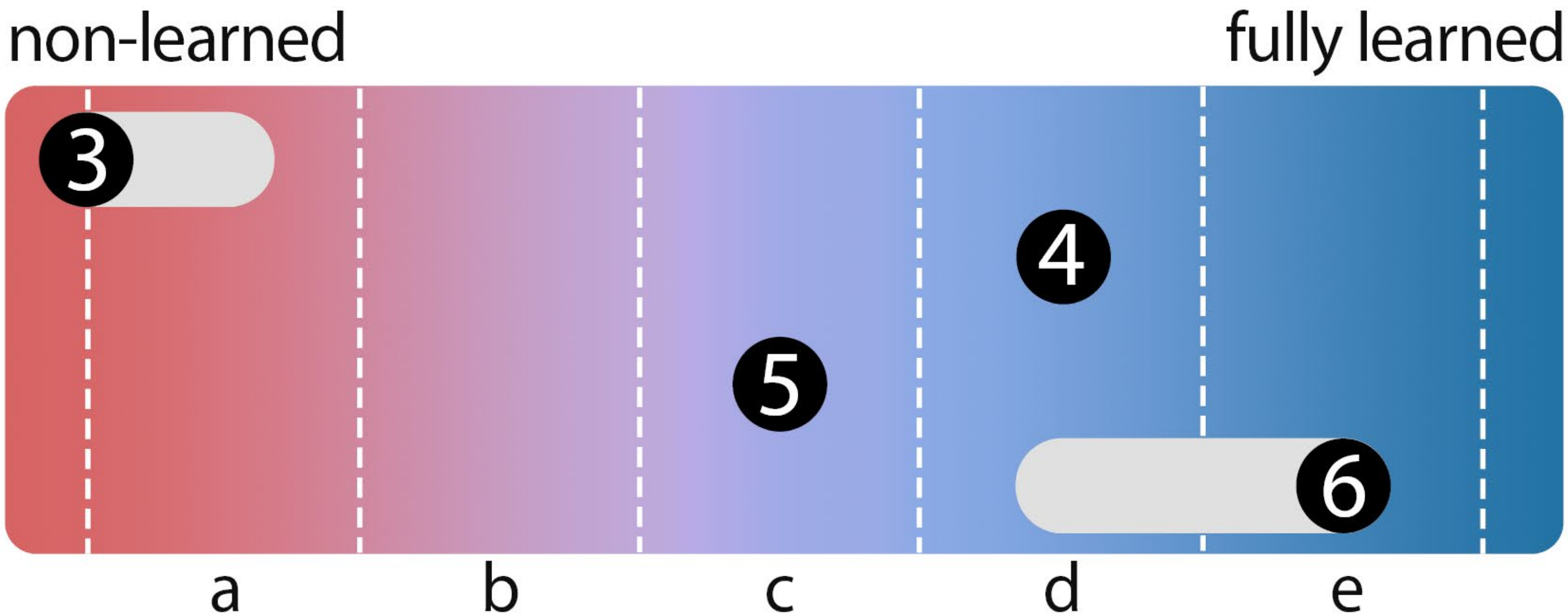
Chen et al - Neural Ordinary Differential Equations, 2018

Ciccone et al - Stable Deep Networks from Non-Autonomous DEs, 2018

Haber, Ruthotto - Stable architectures for deep neural networks, 2017

Inverse Problems and Deep Learning

Yoeri Boink



MODEL-DRIVEN

DATA-DRIVEN

Boink - Data-driven reconstruction methods for photoacoustic tomography – Learning structures by structured learning (PhD thesis, 2021)

DEEP LEARNING FOR INVERSE PROBLEMS

- Fully Learned Models
  $$x = \varphi_\theta(y)$$
- Post Processing
  $$x = \varphi_\theta(A^\dagger(y))$$
- Iterative Schemes
  $$x_0 = A^\dagger(y)$$
  $$x_{n+1} = \varphi_\theta(x_n, \nabla_{x_n} \|A(x_n) - y\|)$$
- Learning a Regularizer
- Review

Zhu et al – Image reconstruction by domain-transform manifold learning (Nature, 2018)

Jin, McCann, Froustey, Unser – Deep Convolutional Neural Network for Inverse Problems in Imaging (2018)

Putzky, Welling - Recurrent Inference Machines for Solving Inverse Problems (2017)

Meinhardt, Möller, Hazirbas, Cremers - Learning Proximal Operators (2017)

Banert, Ringh, Adler, Karlsson, Öktem – Data-driven nonsmooth optimization (2018)

Li, Schwab, Antholzer, Haltmeier - NETT Solving Inverse Problems with Deep Neural Networks (2018)

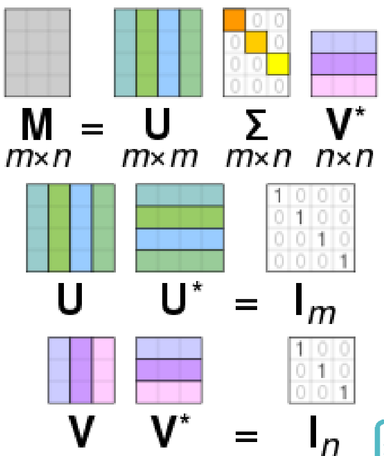
Lunz, Öktem, Schönlieb – Adversarial Regularizers in Inverse Problems (2018)

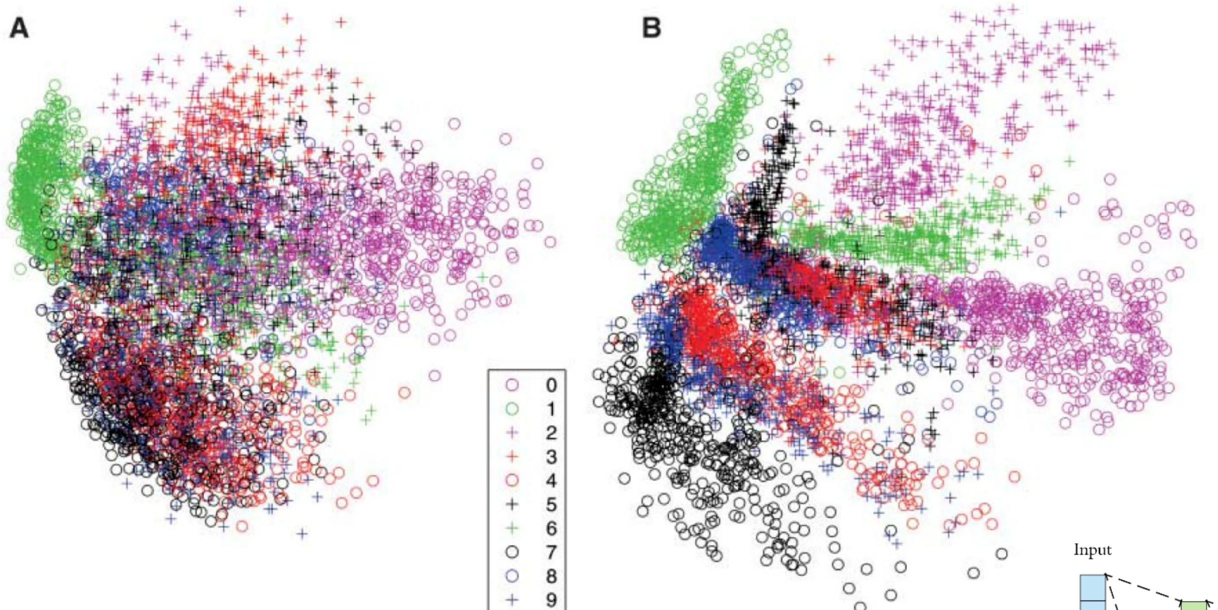
Arridge et al – Solving inverse problems using data-driven models (Acta Numerica, 2019)

DATA DIMENSIONALITY REDUCTION

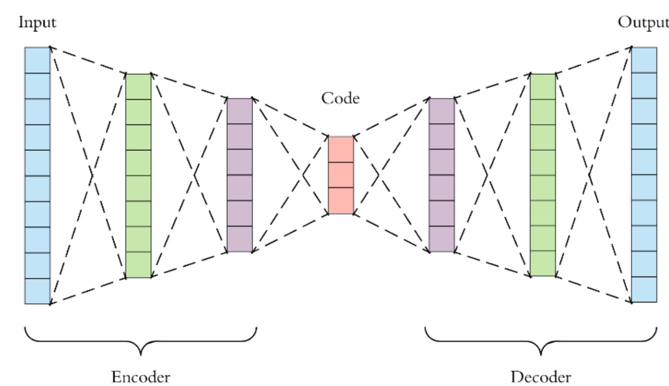
STRUCTURE, SCALE AND INTERPOLATION OF DATA

SVD
linear





Autoencoder
nonlinear

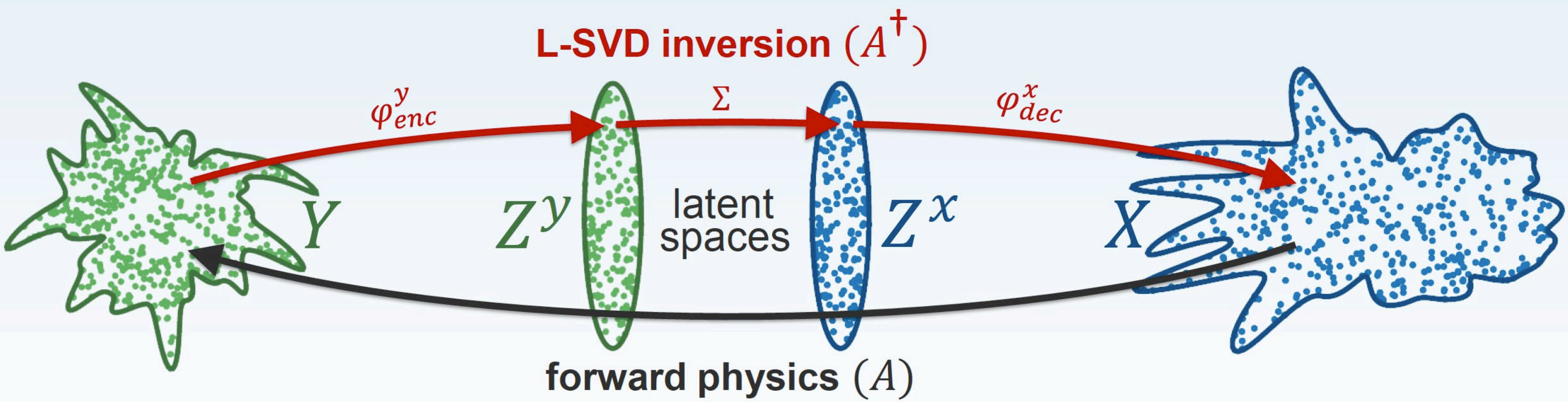


Hinton - Reducing the Dimensionality of Data with Neural Networks (Science, 2006)

Zeune et al - Deep Learning of Circulating Tumor Cells (Nature MI 2020)

LEARNED SINGULAR VALUE DECOMPOSITION (L-SVD)

- Truncated SVD for linear inverse problems $y^\delta = Ax + \eta^\delta,\;\; A = USV^* \;\;\Rightarrow\;\; \tilde{x} = V_K S_K^{-1} U_K^* y^\delta$
- Propose nonlinear, learned SVD $\hat{x}^\Sigma = \varphi^x_{dec}\left(\Sigma\, \varphi^y_{enc}(y^\delta)\right)$



- From model-driven to data-driven
- From linear to nonlinear encoding
- Regularization by hybrid autoencoding

LEARNED SINGULAR VALUE DECOMPOSITION (L-SVD)

Yoeri Boink

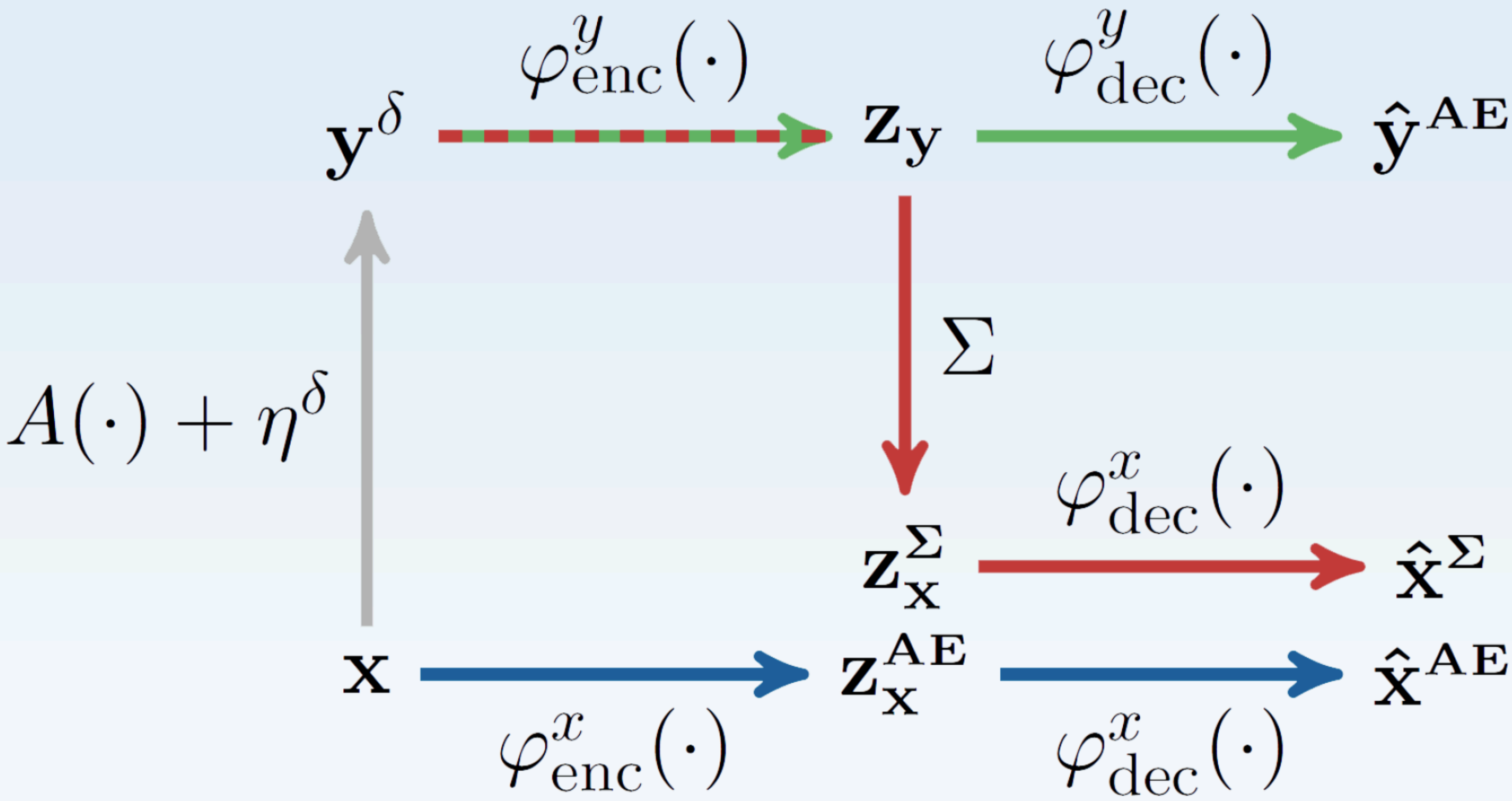


Figure 1: *in **green** the autoencoder for data y^δ; in **blue** the autoencoder for signal x; in **red** the reconstruction procedure.*

LEARNED SINGULAR VALUE DECOMPOSITION (L-SVD)

We define the nonlinear functions

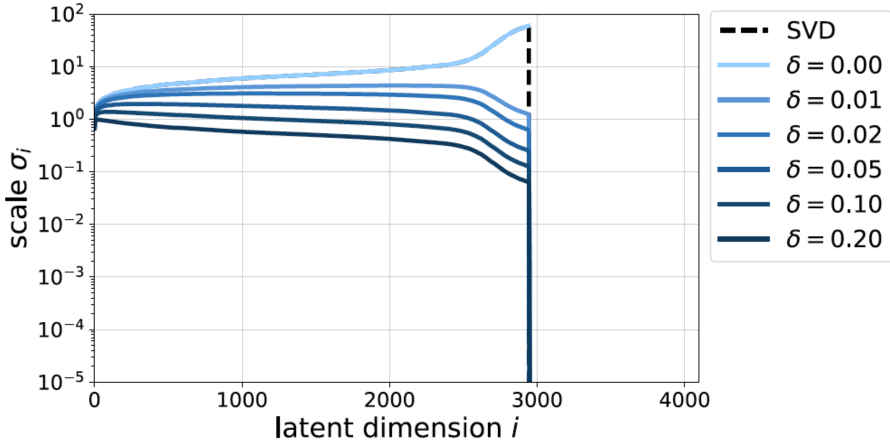
$$\varphi^y_{enc} : Y \mapsto Z^y, \quad \varphi^y_{dec} : Z^y \mapsto Y, \quad \varphi^x_{enc} : X \mapsto Z^x, \quad \varphi^x_{dec} : Z^x \mapsto X$$

and we define the square matrix $\Sigma \in \mathbb{R}^{k \times k}$. *Moreover we define the variables*

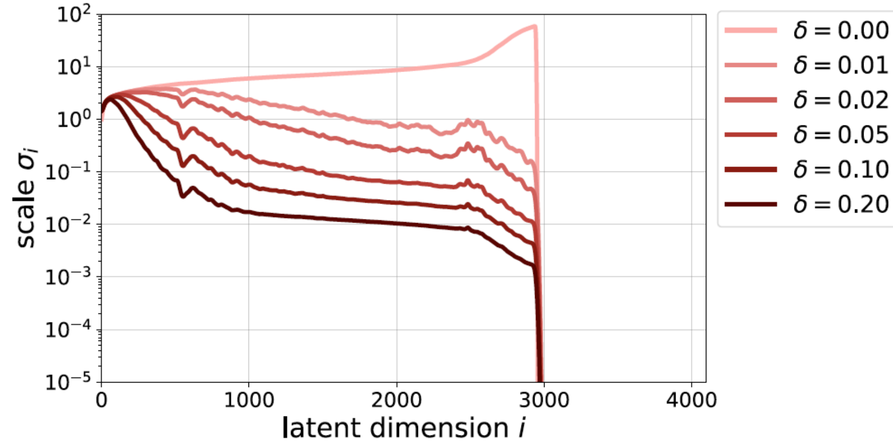
$$\begin{aligned} z_y &:= \varphi^y_{enc}(y^\delta), & z_x^{AE} &:= \varphi^x_{enc}(x), & z_x^\Sigma &:= \Sigma z_y, \\ \hat{y}^{AE} &:= \varphi^y_{dec}(z_y), & \hat{x}^{AE} &:= \varphi^x_{dec}(z_x^{AE}), & \hat{x}^\Sigma &= \varphi^x_{dec}(z_x^\Sigma). \end{aligned}$$

$$\boxed{\min_{\mathrm{par}_{\mathrm{NN}}} \left\{ \sum_{i=1}^{\#\mathrm{train}} \underbrace{D_1\left(\hat{x}^\Sigma_{(i)}, x_{(i)}\right)}_{\mathrm{reconstruction}} + \alpha_y \underbrace{D_2\left(\hat{y}^{\mathrm{AE}}_{(i)}, y_{(i)}\right)}_{\mathrm{autoencoder}} + \alpha_x \underbrace{D_3\left(\hat{x}^{\mathrm{AE}}_{(i)}, x_{(i)}\right)}_{\mathrm{autoencoder}} \right\}}$$

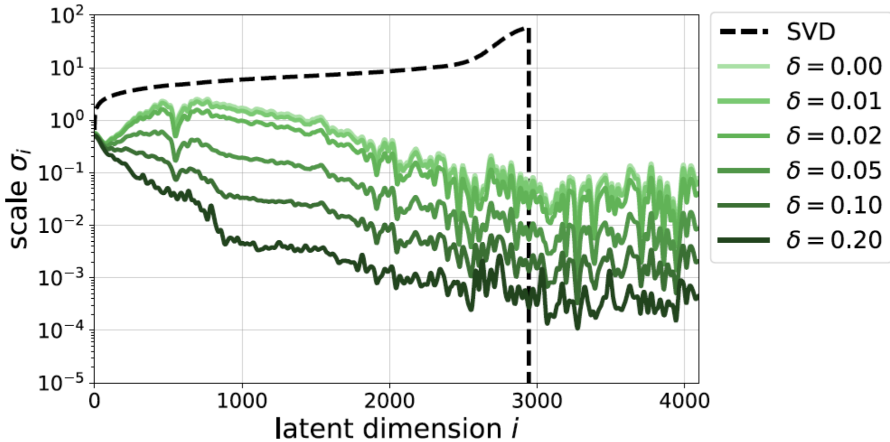
LEARNED REGULARIZATION (SCALES FROM NOISE)



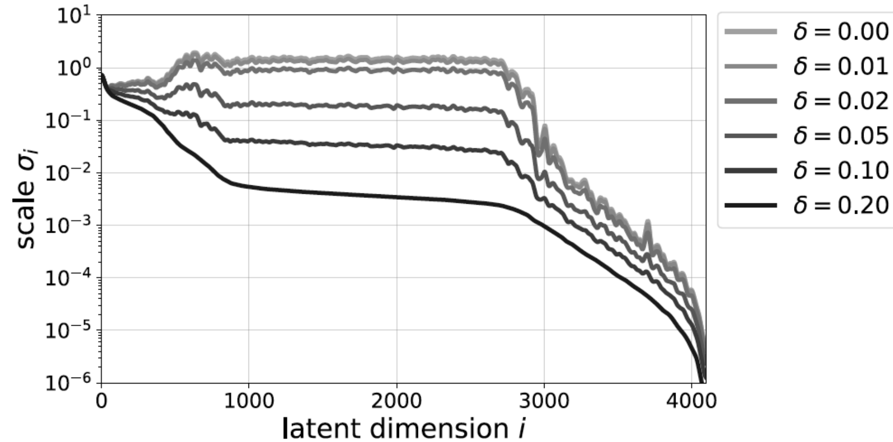
(a) Tikhonov regularisation



(b) Data-driven Tikhonov: U,V fixed



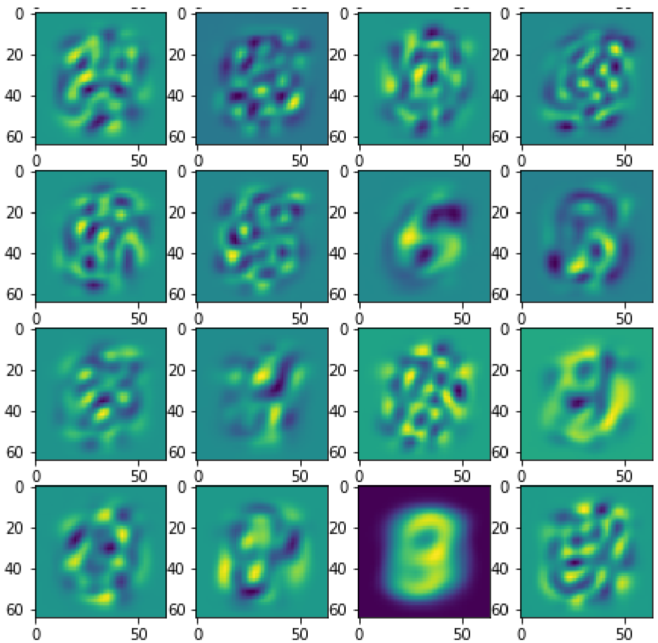
(c) fully learned L-SVD initialised with SVD



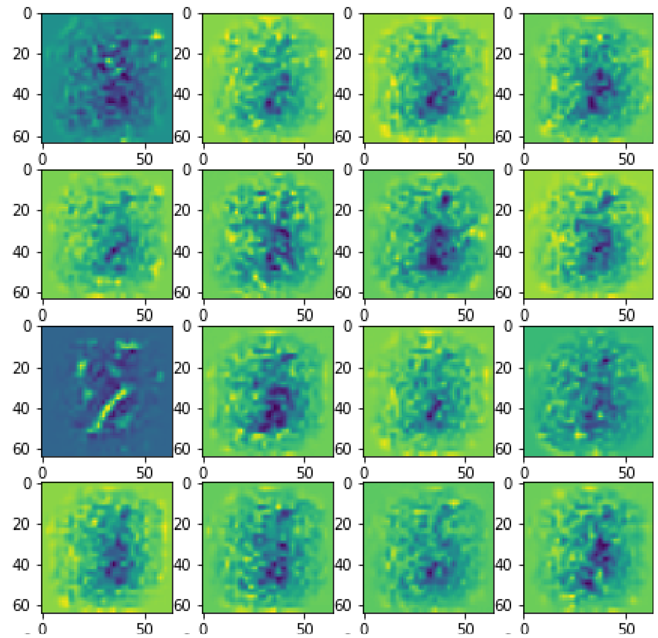
(d) fully learned L-SVD initialised randomly

STRUCTURE: LATENT REPRESENTATIONS

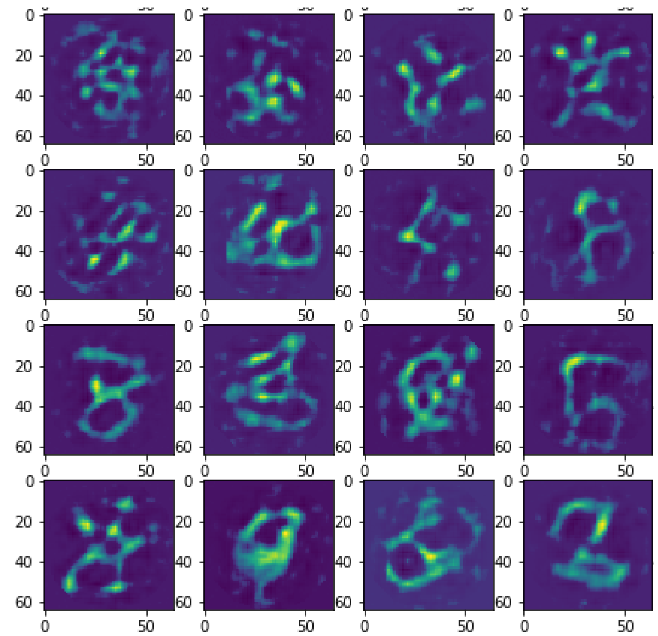
linear AE/L-SVD



nonlinear AE



L-SVD



- Selected unit vectors from **latent space decoded to image space**
- Nonlinear L-SVD learns more interpretable representation
- Combines features from data space, image space and operator
- Autoencoder provides ‘smoother’ and better connected structures

$$
\begin{tikzcd}
Y \arrow[r, "\psi"] & Z_Y \arrow[r, "\psi^*"] \arrow[d, "\Sigma^{-1}"] & Y \arrow[d, "A^{-1}"] \\
X \arrow[u, "A"] \arrow[r, "\phi"] & Z_X \arrow[r, "\phi^*"] & X
\end{tikzcd}
$$

Boink, Brune - Learned SVD - Deep Learning for Solving Inverse Problems via Hybrid Autoencoding (arXiv, 2021)

Learning Dynamics

transport – control

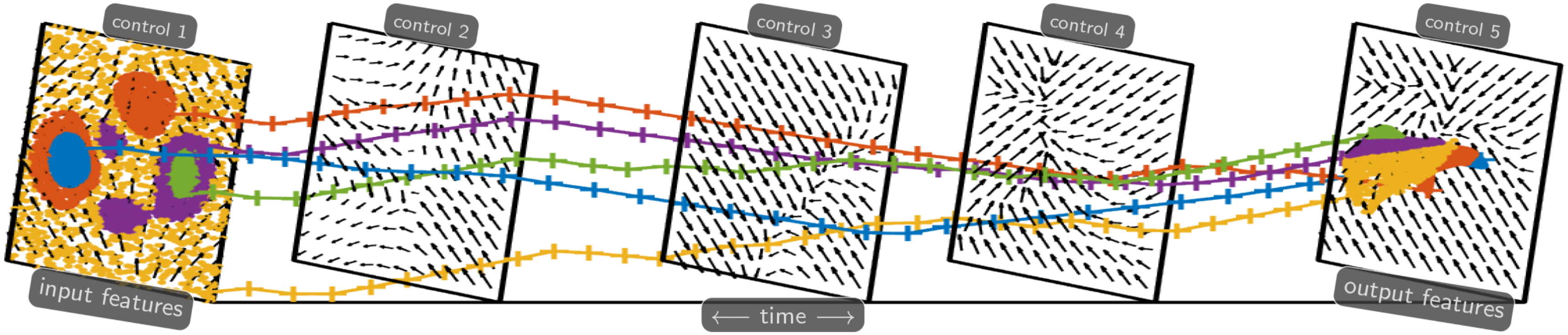
Machine learning

- **"by" dynamical systems: Use dynamical systems to perform machine learning**
  - use (discrete or continuous) dynamical systems to generate large classes of nonlinear trial functions (hypothesis space)
  - formulate as a problem in control theory; algorithms motivated by control theory

- **“of” dynamical systems: Learn an unknown dynamical system**
  - example: model reduction. Given a big dynamical system, learn an approximate much smaller dynamical system
  - issues: short term accuracy and long-term qualitative consistency

- **"for" dynamical systems: Using machine learning to address issues of interest** for a dynamical system
  - supervised learning of time series data
  - typically use recurrent neural networks

OPTIMAL CONTROL APPROACHES TO DEEP LEARNING



Deep Learning meets optimal control / parameter estimation.

- new ways to analyze and design neural networks
- expose similarities to trajectory problem, optimal transport, image registration, . . .
- training algorithms motivated by (robust) optimal control
- discrete ResNet $\rightsquigarrow$ continuous problem $\rightsquigarrow$ discrete architecture

He et al – Deep Residual Learning for Image Recognition, 2016

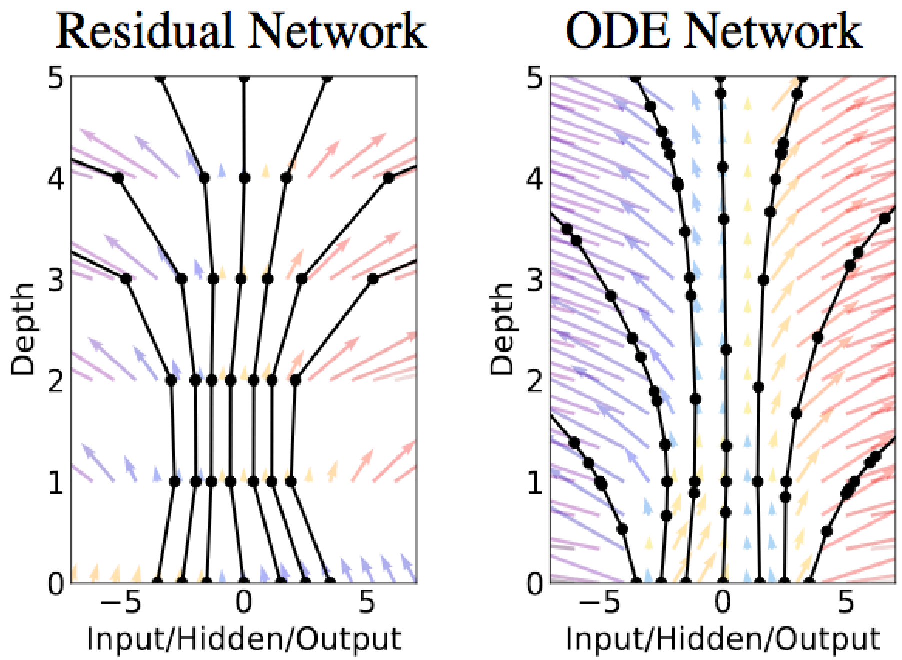
Benning et al – Deep learning as optimal control problems, 2019

Haber et al – Stable architectures for deep neural networks, 2017

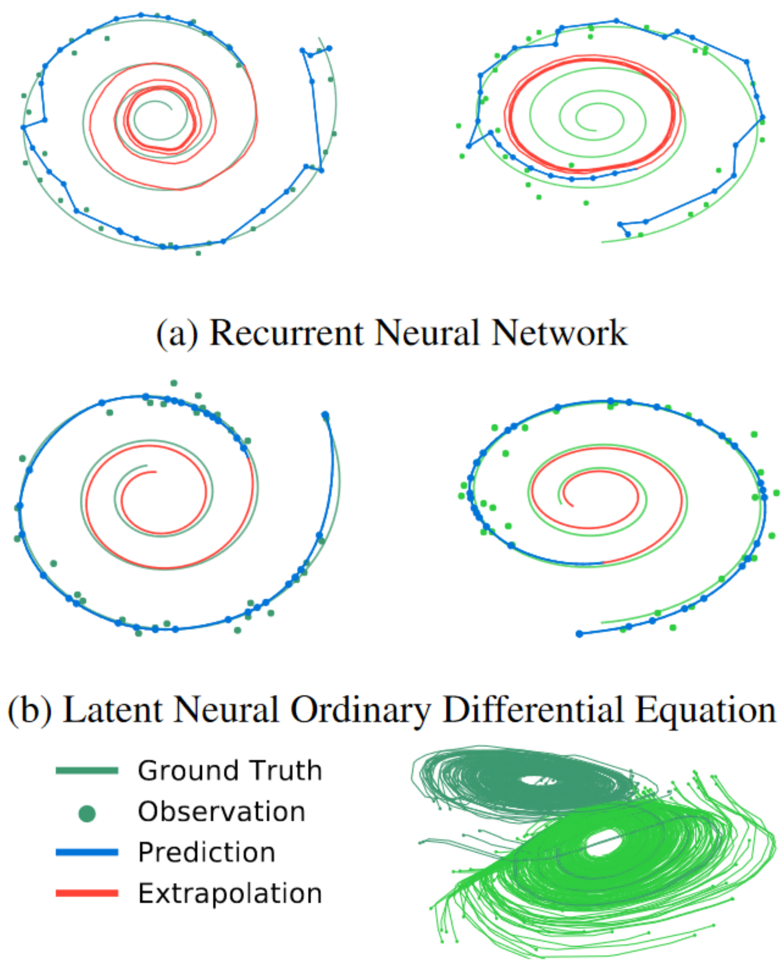
NEURAL ODES – CONTINUOUS NORMALIZING FLOWS

Generative Latent ODE models

learning via Pontryagin's adjoint sensitivity method



$$\mathbf{h}_{t+1} = \mathbf{h}_t + f(\mathbf{h}_t, \theta_t) \qquad \frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), t, \theta)$$



(a) Recurrent Neural Network

(b) Latent Neural Ordinary Differential Equation

(c) Latent Trajectories

Chen et al - Neural Ordinary Differential Equations, 2018

GENERATIVE MODELING WITH NORMALIZING FLOWS

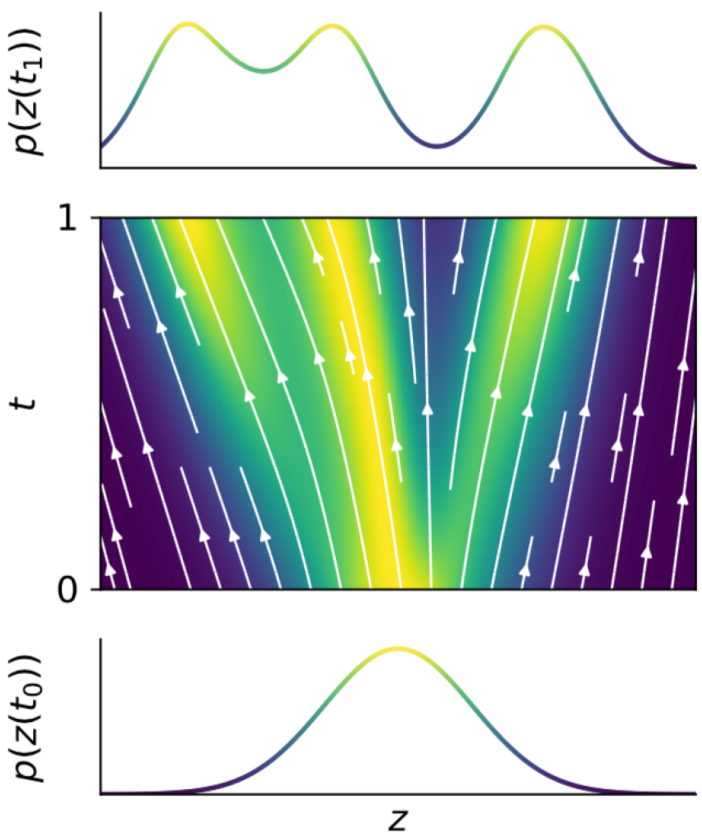


image: Grathwohl et al. 2019

D Rezende, S Mohamed
Variational Inference with Normalizing Flows. *SIAM SISC*,39(5), arXiv, 2015.

W Grathwohl et al.
FFJORD: Free-form Continuous Dynamics for Scalable Reversible Generative Models. *arXiv*, 2018.

L Yang, GE Karniadakis
Potential Flow Generator with L_2 OT Regularity for Generative Models. *arXiv:1908.11462v1*, 2018.

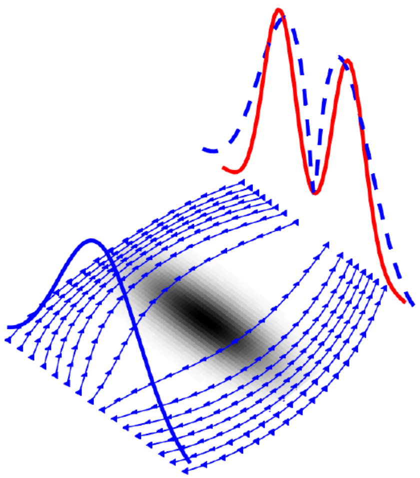
L Zhang, Weinan E, L Wang
Monge-Ampère Flow for Generative Modeling, *arXiv:1809.10188v1*, 2018.

J Lin, K Lensink, E Haber
Fluid Flow Mass Transport for Generative Networks, *arXiv:1910.01694v2*, 2019.

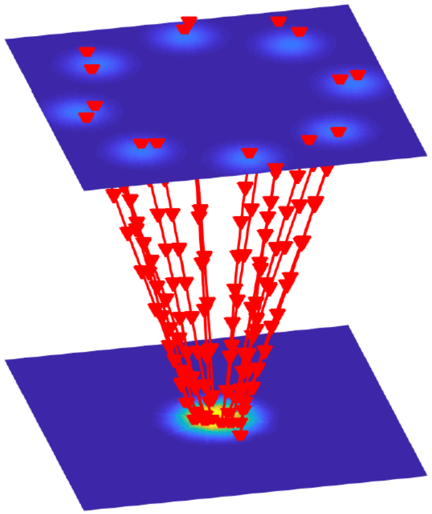
Potential for medical imaging: data-driven priors and regularizers

MACHINE LEARNING FOR HIGH-DIMENSIONAL OPTIMAL TRANSPORT

Optimal Transport as Mean Field Game

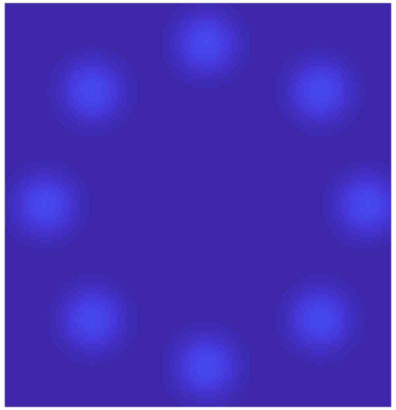


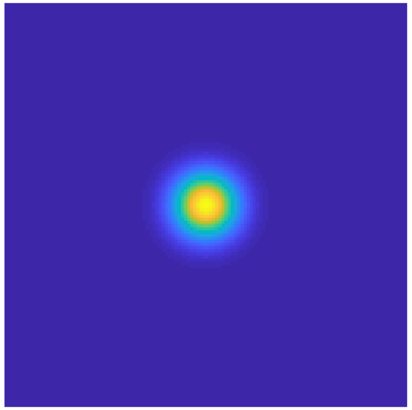
- **Motivation: High-Dimensional OT**
  - Generative modeling
  - Challenge: curse of dimensionality
- **Three Options for OT as Mean Field Game**
  - macroscopic view: variational problem
  - microscopic view: coupled optimization problem
  - Hamilton-Jacobi-Bellman (HJB) equation

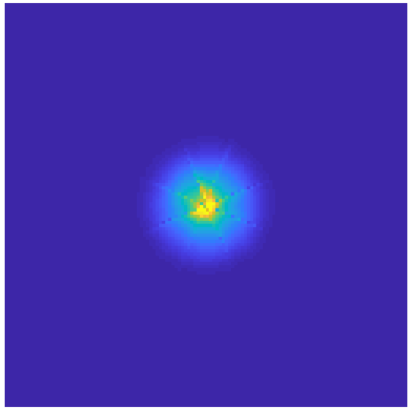


Ruthotto et al – A Machine Learning Framework for Solving High-Dimensional Mean Field Game and Mean Field Control Problems (PNAS, 2020)

OPTIMAL MASS TRANSPORT / WASSERSTEIN DISTANCE







Optimal Mass Transport

Given an initial density, ρ_0, and a target density, ρ_1, find the velocity field v that renders the push-forward of ρ_0 equal to ρ_1 and minimizes the transport costs, i.e., solve

$$\text{minimize}_{v,\rho} \quad \int_0^1 \int_\Omega \|v(x,t)\|^2 \rho(x,t)\,dxdt$$

$$\text{subject to} \quad \partial_t \rho + \nabla \cdot (\rho v) = 0, \quad \rho(\cdot,0) = \rho_0(\cdot), \quad \rho(\cdot,1) = \rho_1(\cdot)$$

Benamou, Brenier - A computational fluid mechanics solution to the Monge-Kantorovich mass transfer problem (Numer. Math. 2000)

The books by Cedric Villani

Peyre, Cuturi - Computational OT

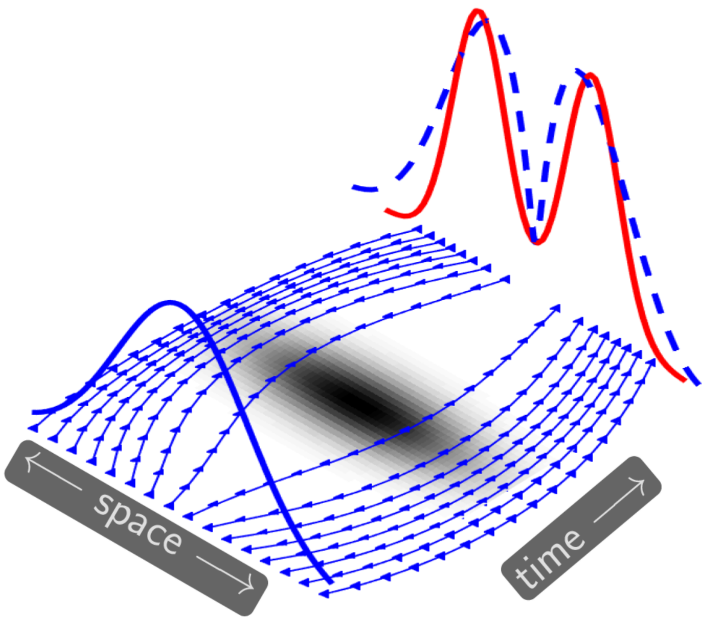
MEAN FIELD GAMES / MEAN FIELD CONTROL (EULERIAN VIEW)

MFGs model large populations of rational agents playing non-cooperative differential game.

$$\text{minimize}_{v,\rho}\ \mathcal{J}_{\text{MFG}} := \int_0^1 \int_{\mathbb{R}^d} L\left(x, v(x,t)\right) \rho(x,t) dx dt + \int_0^1 \mathcal{F}(\rho(\cdot,t)) dt + \mathcal{G}(\rho(\cdot,1))$$

$$\text{subject to} \quad \partial_t \rho(x,t) + \nabla \cdot (\rho(x,t) v(x,t)) = 0, \quad \rho(x,0) = \rho_0(x),$$

- ρ_0: initial density (given)
- L: transport costs.
  Today: $L(x,v) = \frac{1}{2}\|v\|^2$
- $\mathcal{F}$: running costs.
  E.g., congestion, preference
  For OT: $\mathcal{F} \equiv 0$
- $\mathcal{G}$: terminal costs.
  E.g., cross entropy of ρ_1 and $\rho(\cdot,1)$



AGENT-BASED MFG MODEL (LAGRANGIAN VIEW)

Let (x, t) be state of a single agent that aims at choosing v that minimizes

$$J_{x,t}(v) = \int_t^T \|v(s)\|^2 + F\left(z(s), \rho(z(s), s)\right) ds + G\left(z(T), \rho(z(T), T)\right),$$

where position changes as in

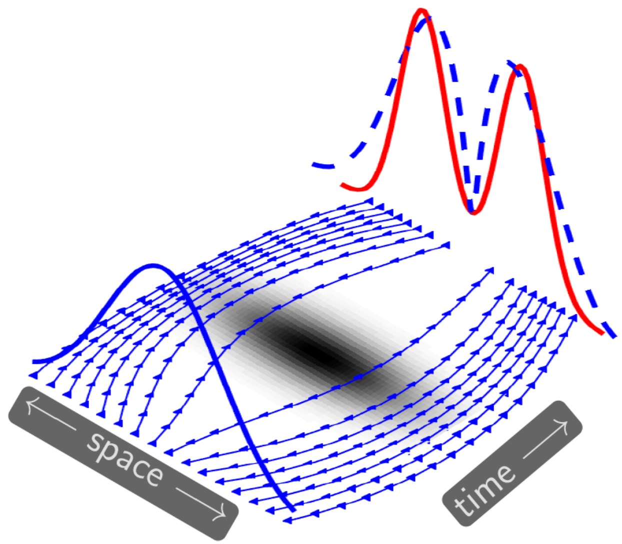
$$\partial_t z(t) = v\left(t\right),\ 0 \leq t \leq T, \quad z(0) = x.$$

- F: running costs (Ex: congestion, preference)
- G: terminal costs (Ex: cross entropy)
- Define value function

$$\Phi(x, t) = \inf_v J_{x,t}(v)$$

- MFG is called $\boxed{\text{potential MFG}}$ if

$$F(x, \rho) = \frac{\delta \mathcal{F}(\rho)}{\delta \rho}(x), \quad G(x, \rho) = \frac{\delta \mathcal{G}(\rho)}{\delta \rho}(x),$$



HAMILTON-JACOBI-BELLMANN (HJB) AND FOKKER-PLANCK

Lasry & Lions '06: First-order optimality conditions of $\boxed{\text{potential MFG}}$ are

Novelty: stochastic potential MFG

$$-\partial_t \phi(x,t) - \nu \Delta \phi(x,t) + H(x, \nabla \phi(x,t)) = F(x,\rho) \qquad \text{(HJB)}$$

$$\partial_t \rho(x,t) - \nu \Delta \rho(x,t) - \nabla \cdot (\rho(x,t) \nabla_p H(x, \nabla \phi)) = 0 \qquad \text{(FP)}$$

$$\rho(x,0) = \rho_0(x), \quad \phi(x,T) = g(x, \rho(\cdot, T))$$

$$H(x,p) = \sup_v (-\langle p, v \rangle - L(x,v)) \qquad \text{Hamiltonian}$$

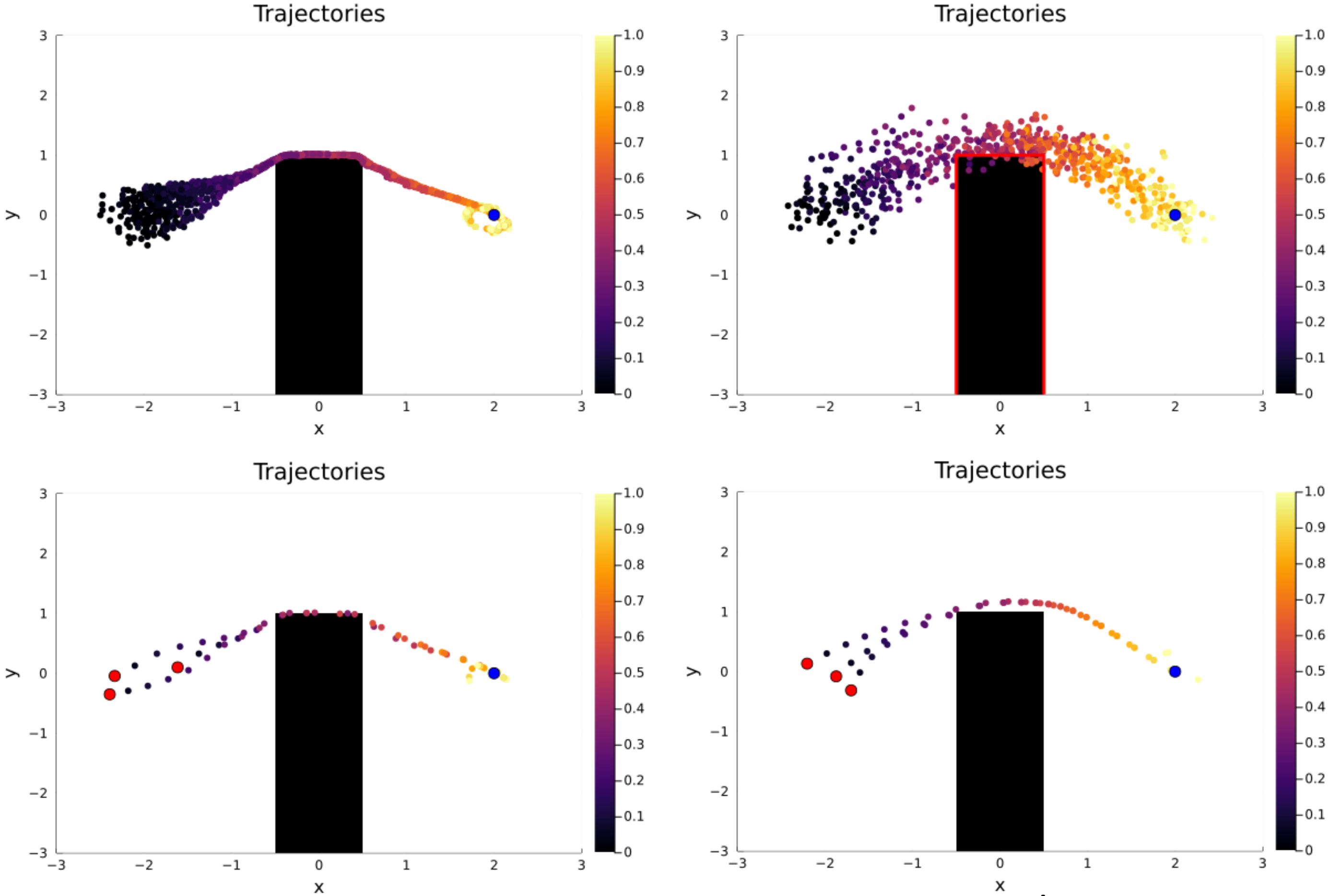
$$dz = v(z,t)dt + \sqrt{2\nu} dW(t), z \sim \rho_0 \qquad \text{Agent dynamics}$$

$$v(z,t) = -\nabla_p H(z, \nabla \phi) \qquad \text{at optimality}$$

Lin et al - Apac-net: Alternating the population and agent control via two neural networks to solve high-dimensional stochastic mean field games (arXiv, 2020)

NEURAL SDE FOR SOLVING STOCHASTIC MFG

2 step method: 1.) Learn generator Neural SDE based on ResNet
2.) Use value function



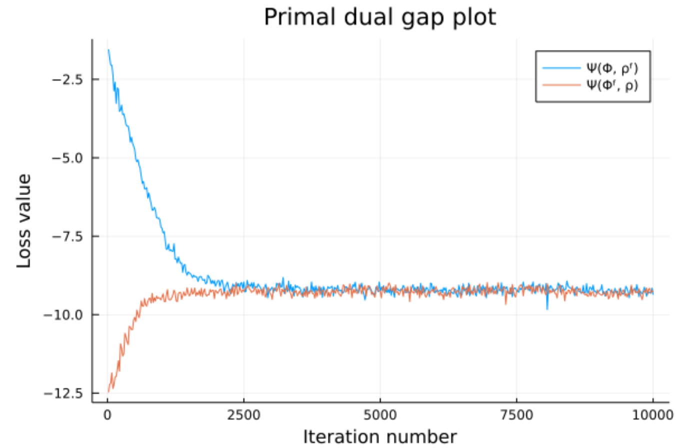
low diffusivity/stochasticity

high diffusivity/stochasticity

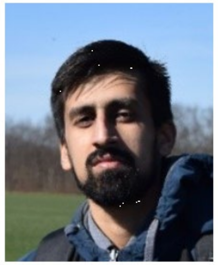
Sven Dummer

$$\Psi(\rho^*, \phi^*) = \sup_{C^2(\Omega_T)} \left(\Psi(\rho, \phi^*)\right)$$

$$\Psi(\rho^*, \phi^*) = \inf_{C^2(\Omega_T)} \left(\Psi(\rho^*, \phi)\right)$$

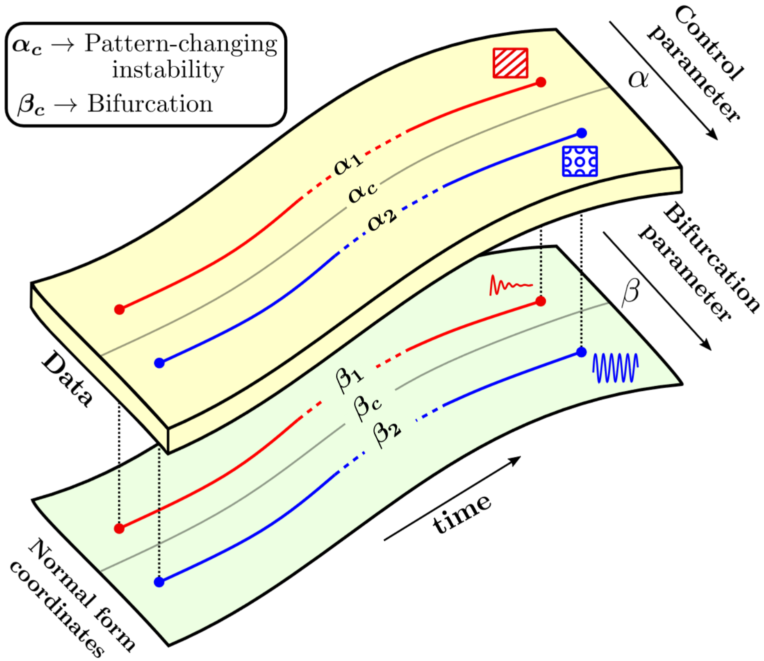


Deep learning of normal form autoencoders for universal, parameter-dependent dynamics



Manu Kalia

Hil Meijer



Nathan Kutz

UNIVERSITY of WASHINGTON

Steven Brunton

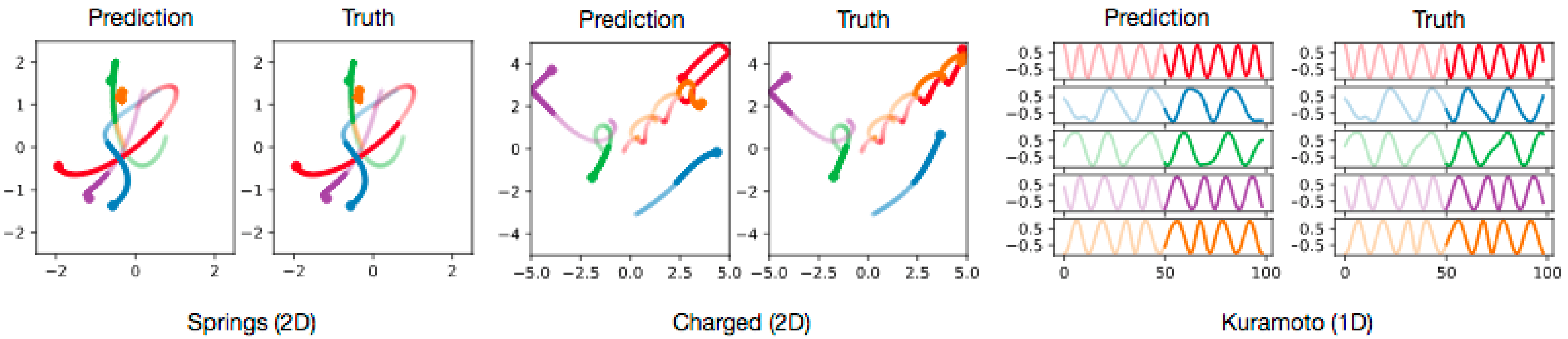
Kalia, et al - Deep learning of normal form autoencoders for universal, parameter-dependent dynamics (Neurips 2020)

Kalia, et al - Learning normal form autoencoders for data-driven discovery of universal, parameter-dependent governing equations (arXiv, 2021)

DEEP LEARNING AND DYNAMICAL SYSTEMS

PHYSICS-INFORMED MACHINE LEARNING & MODEL-ORDER REDUCTION

- **Inverse problems & data assimilation**
- Koopman operators for spectral analysis
- Persistent homology & optimal transport on graphs
- **Discovery of equations & normal forms**
- **Physics-informed machine learning**



Raissi et al - Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations (arXiv:1711.10561, 2017)

Brunton et al - Discovering governing equations from data by sparse identification of nonlinear dynamical systems (PNAS, 2016)

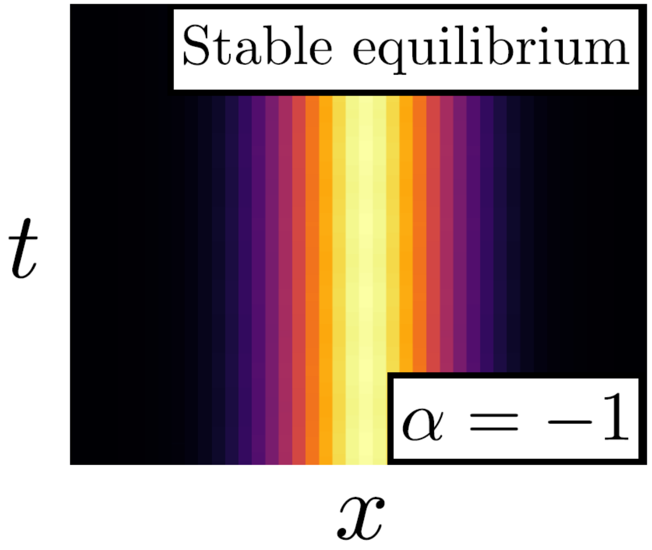
Hermann et al - Deep neural network expression of posterior expectations in Bayesian PDE inversion (Inverse Problems, 2020)

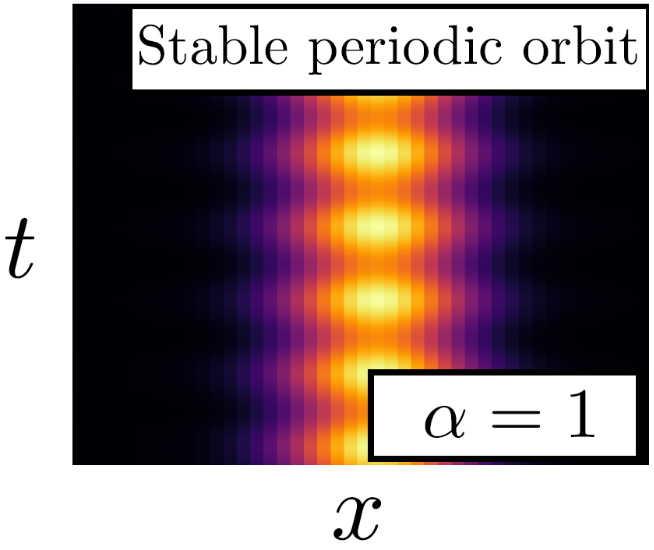
MODEL-ORDER REDUCTION & INSTABILITIES

Consider a spatiotemporal system, parameterized by α. Different choices of α produce different patterns.

Can we construct **underlying low-dimensional models that capture α-dependence?**

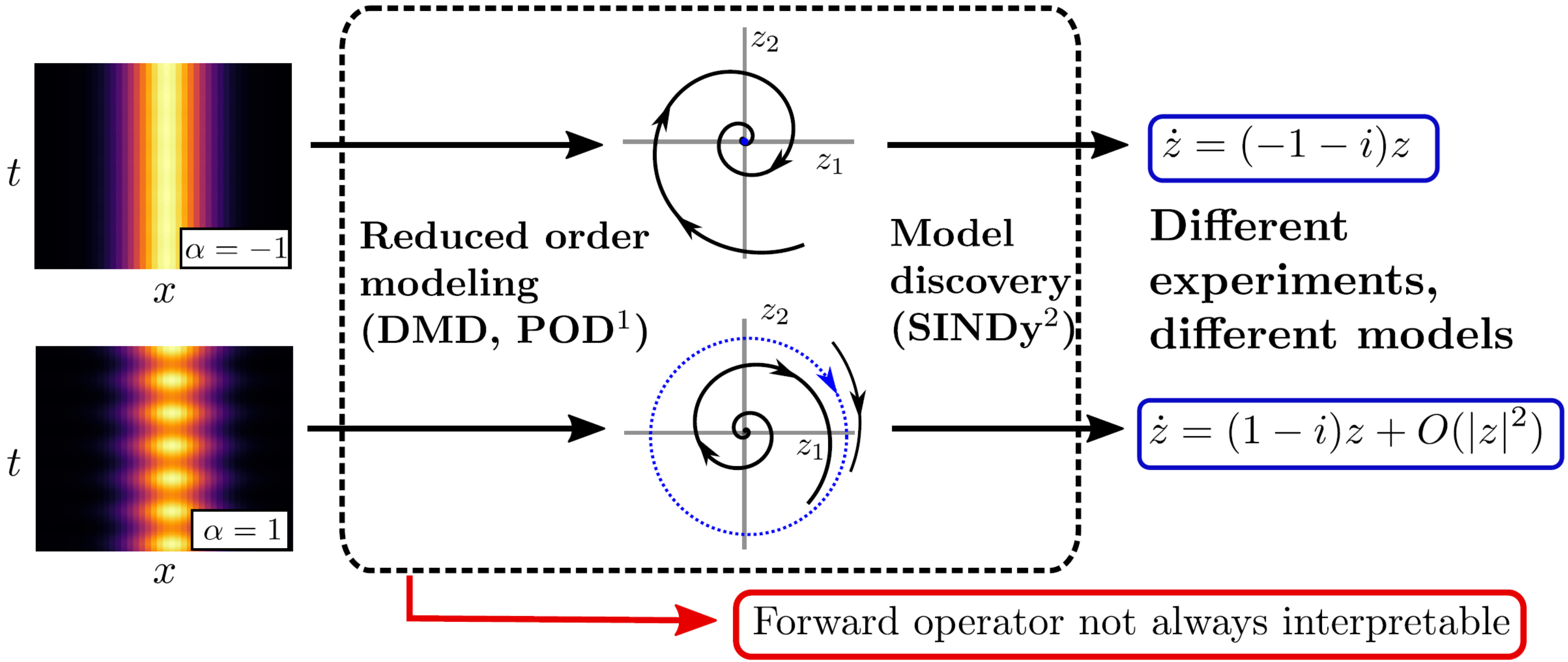
$$\dot{x} = f(x, \alpha), \quad \text{(Original dynamics)}$$
$$\dot{z} = g(z, \beta), \quad \text{(Low dim. model)}$$





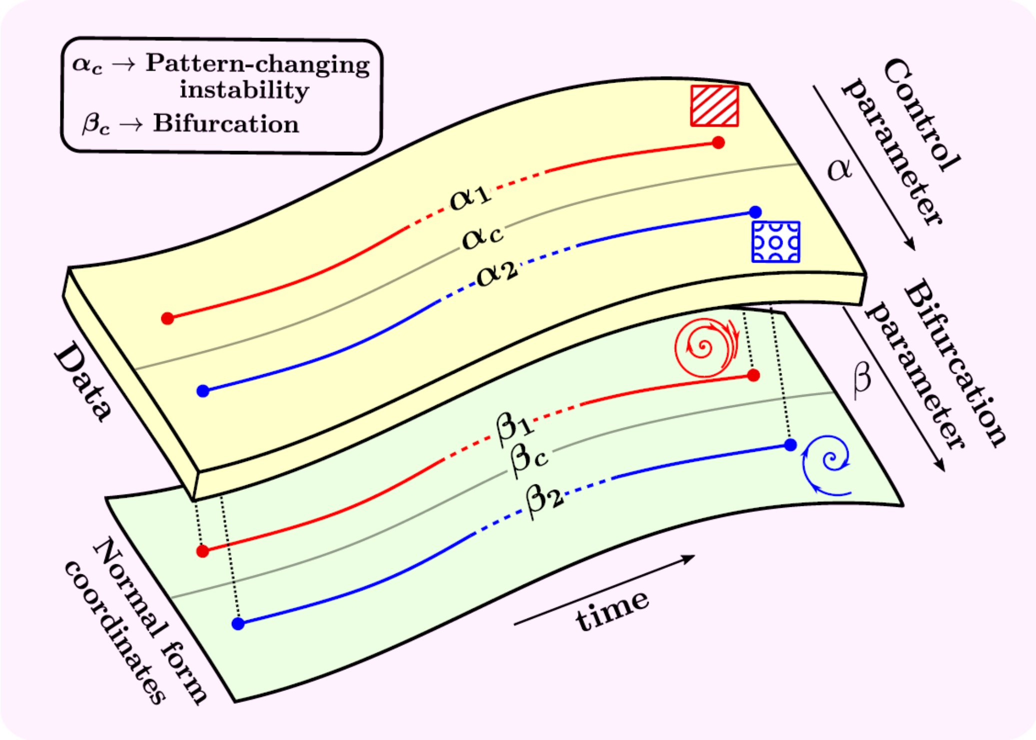
DEEP LEARNING MODEL-ORDER REDUCTION

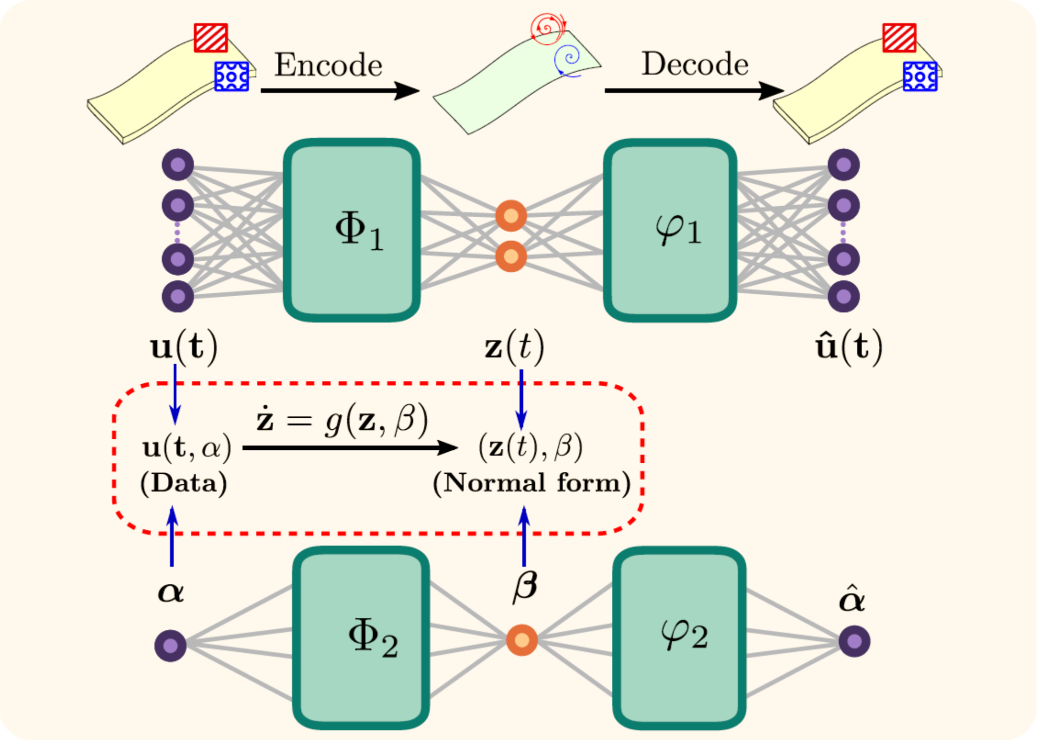
PHYSICS-INFORMED NEURAL NETWORKS[3]

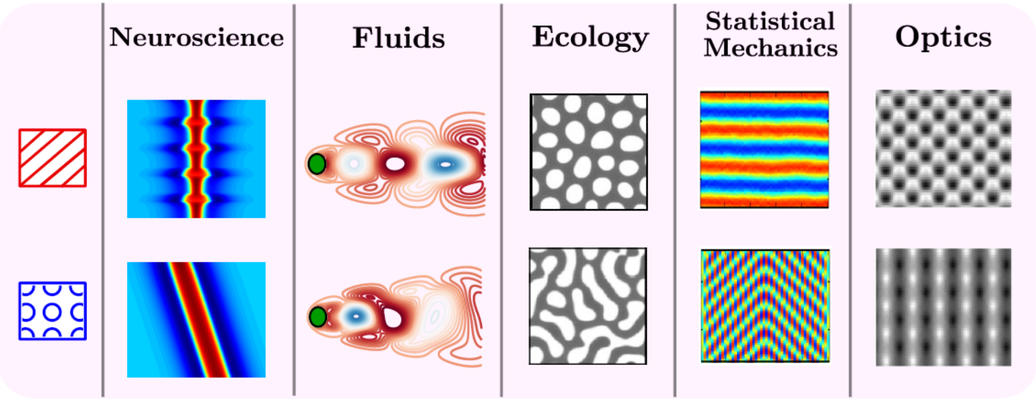


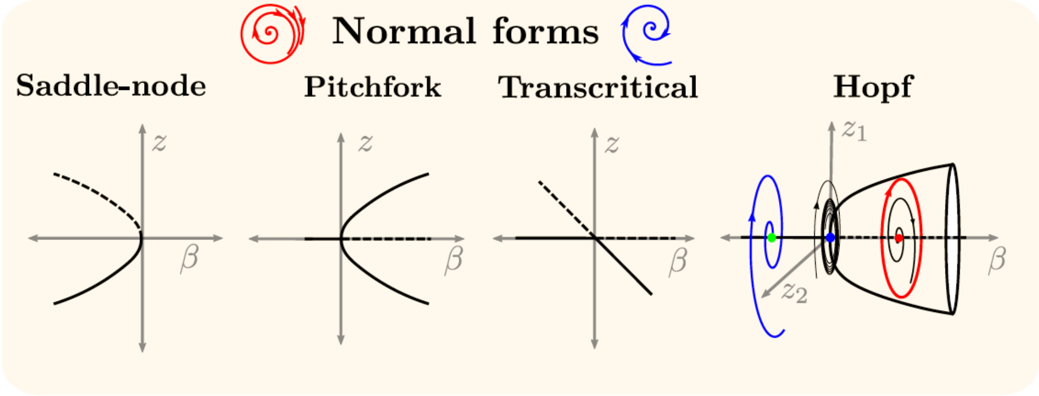
[1] Berkooz et al. *Annual review of fluid mechanics* (1993) ; Tu et al. *Journal of Computational Dynamics* (2014)
[2] Brunton et al. *PNAS* (2016) [3] Raissi et al. Journal of Comp. Phys. (2019)

DEEP LEARNING OF INSTABILITIES/BIFURCATIONS

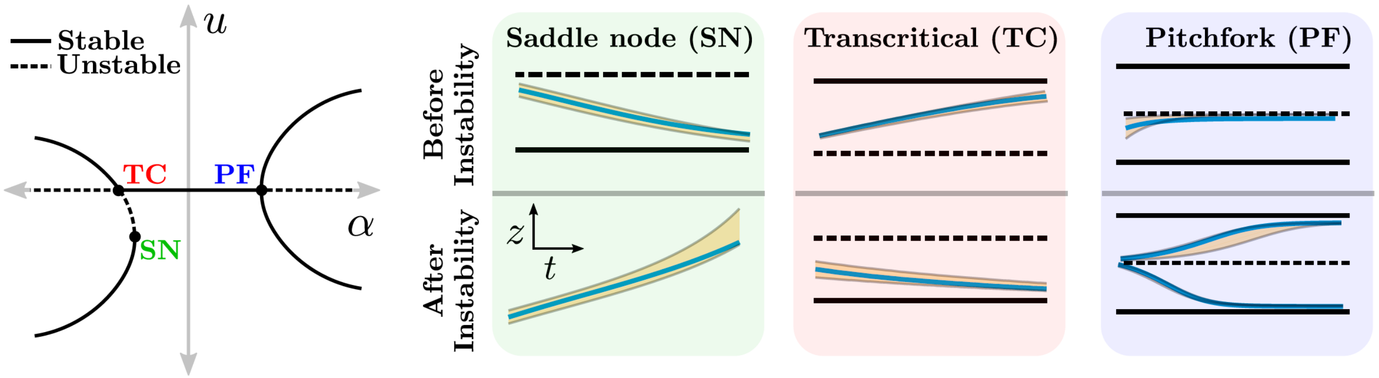


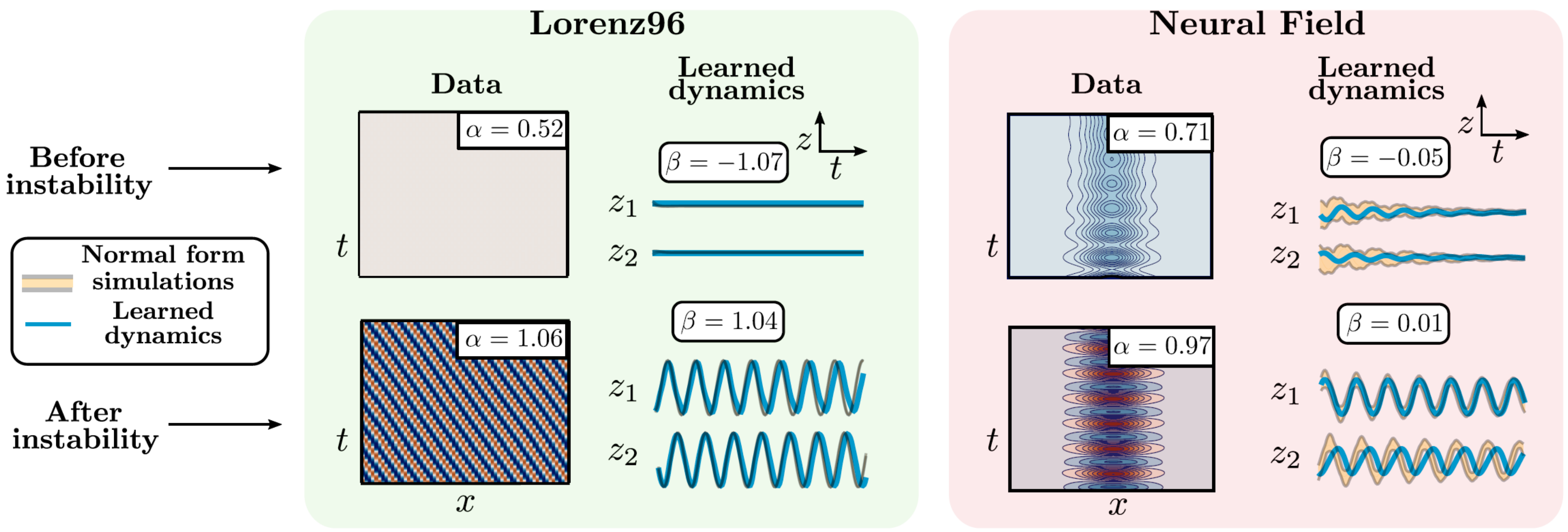




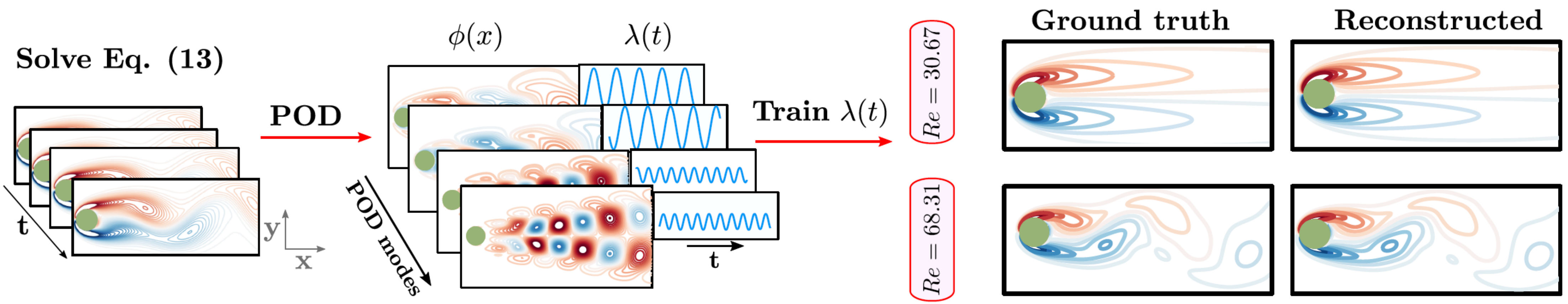


LEARNING NORMAL FORM IN DIFFERENT TYPES OF BIFURCATIONS





NORMAL FORMS & BIFURCATION IN FLUID FLOW PAST A CYLINDER

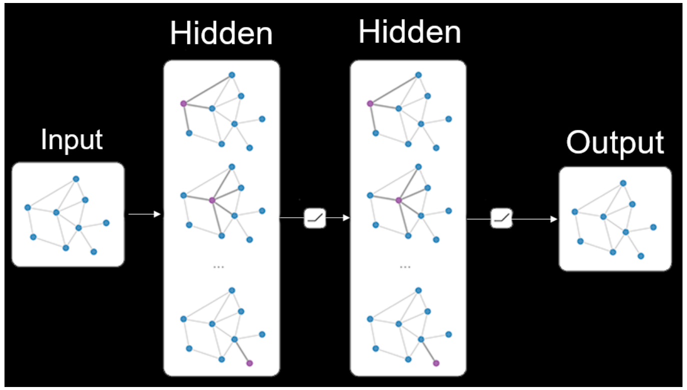


Kalia, et al - Deep learning of normal form autoencoders for universal, parameter-dependent dynamics (Neurips, 2020)

Kalia, et al - Learning normal form autoencoders for data-driven discovery of universal, parameter-dependent governing equations (arXiv, 2021)

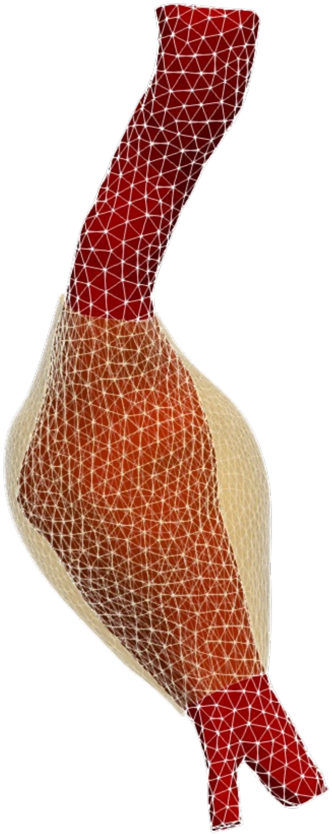
Geometric Priors

Graph-based DL and beyond



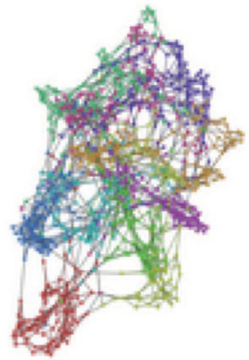
Bronstein, et al – Geometric Deep Learning, Grids, Groups, Graphs, Geodesics, and Gauges (arXiv, 2021)

GEOMETRIC DEEP LEARNING IN GRAPH DOMAINS

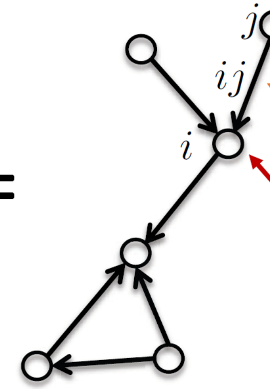


Graphs G are defined by :

- Vertices V
- Edges E
- Adjacency matrix A



=



$$h_j = \begin{bmatrix} \vdots \end{bmatrix} \in \mathbb{R}^{d_v}$$

$$e_{ij} = \begin{bmatrix} \vdots \end{bmatrix} \in \mathbb{R}^{d_e}$$

$$h_i = \begin{bmatrix} \vdots \end{bmatrix} \in \mathbb{R}^{d_v}$$

$$\underbrace{\begin{gathered} \mathcal{G} = (V, E, A) \\ V = \{1, .., n\} \end{gathered}}$$

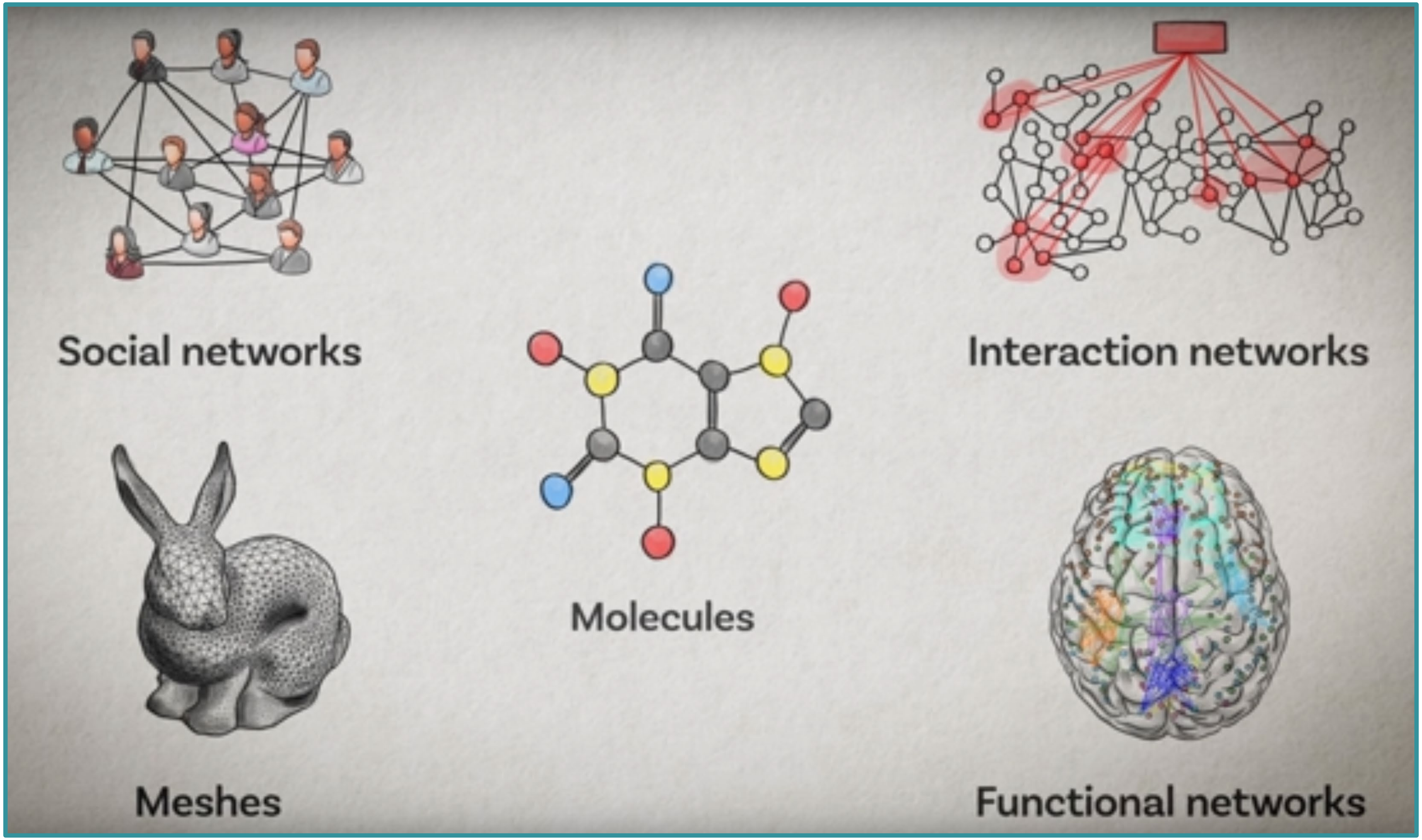
$$g = \begin{bmatrix} \vdots \end{bmatrix} \in \mathbb{R}^{d_g}$$

Graph features :

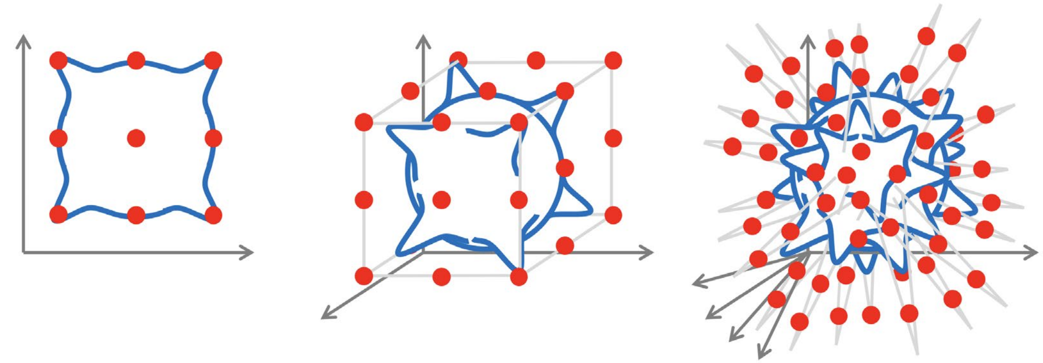
- Node features : $h_i, \ h_j$ (atom type)
- Edge features : e_{ij} (bond type)
- Graph features : g (molecule energy)

PointNet-like methods
(MP) graph CNN methods
Mesh CNN methods

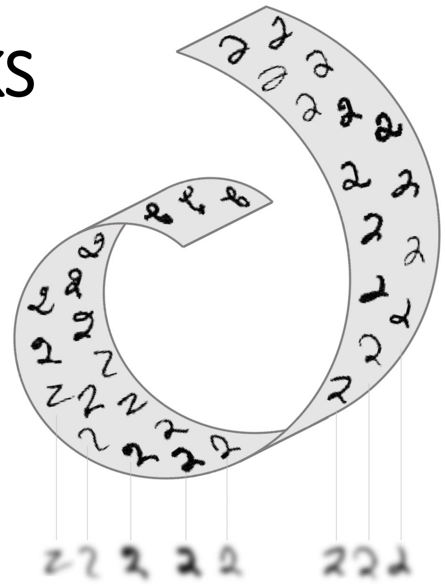
Qi, et al - PointNet: Deep learning on point sets for 3d classification and segmentation (CVPR, 2017)


Social networks
Interaction networks
Molecules
Meshes
Functional networks

CURSE OF DIMENSIONALITY

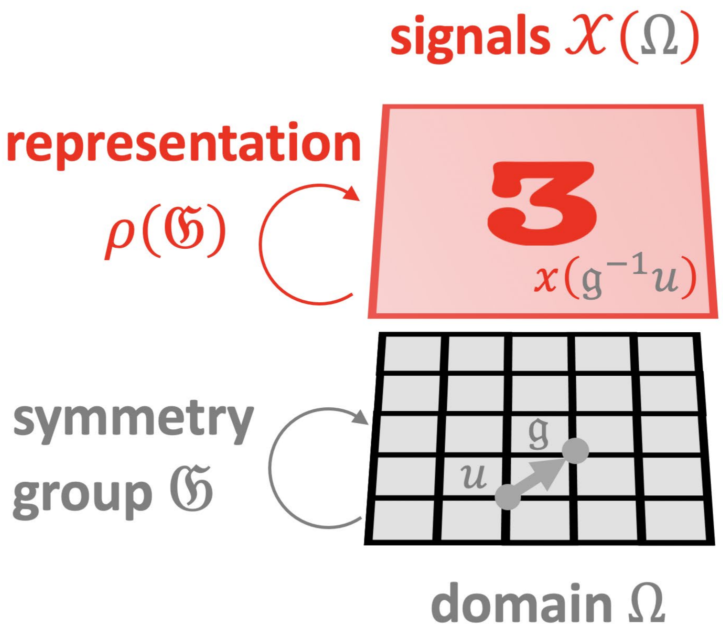


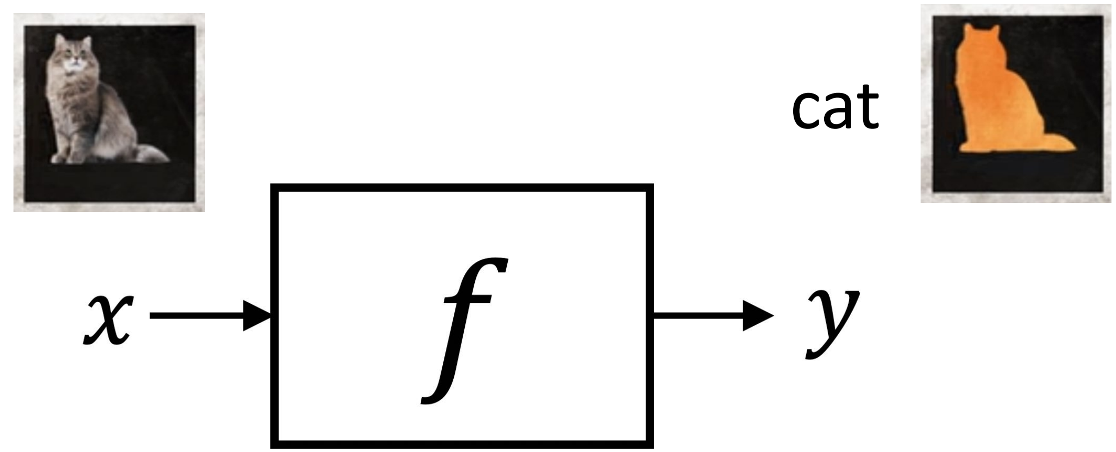
SHALLOW NEURAL NETWORKS



Bach - Breaking the Curse of Dimensionality with Convex Neural Networks (JMLR, 2017)

GEOMETRIC PRIORS – GROUP SYMMETRIES

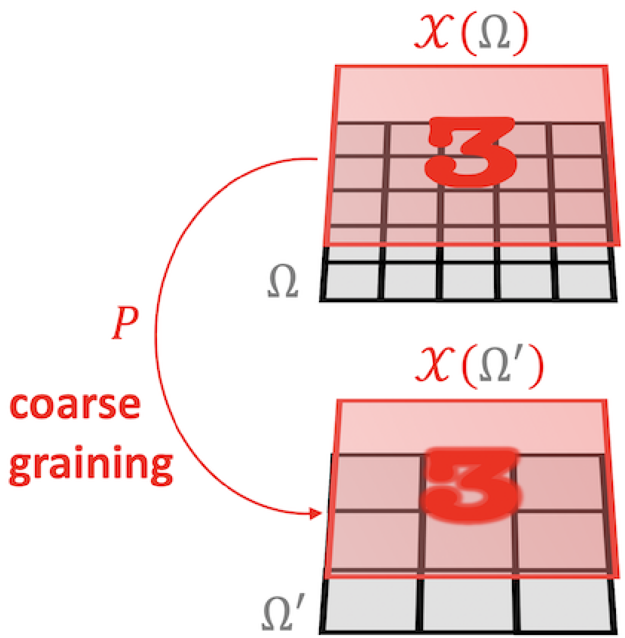


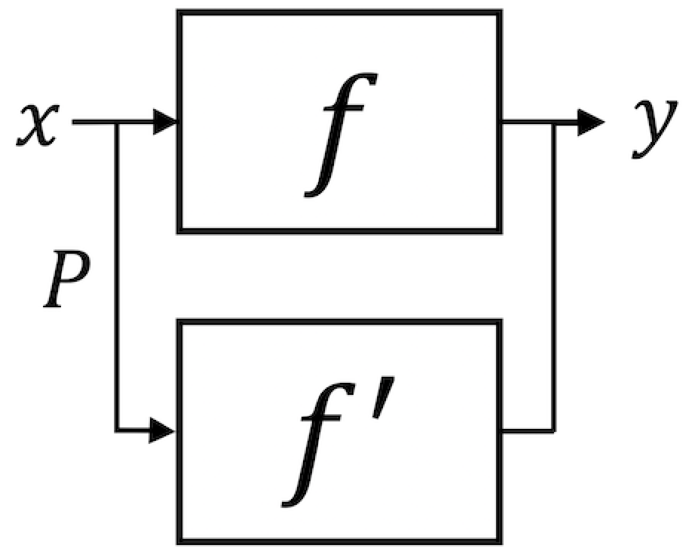


$$f : \mathcal{X}(\Omega) \to \mathcal{Y} \text{ is } \mathfrak{G}\text{-invariant if } f(\rho(\mathfrak{g})x) = f(x) \text{ for all } \mathfrak{g} \in \mathfrak{G} \text{ and } x \in \mathcal{X}(\Omega)$$

$$f : \mathcal{X}(\Omega) \to \mathcal{X}(\Omega) \text{ is } \mathfrak{G}\text{-equivariant if } f(\rho(\mathfrak{g})x) = \rho(\mathfrak{g})f(x) \text{ for all } \mathfrak{g} \in \mathfrak{G}$$

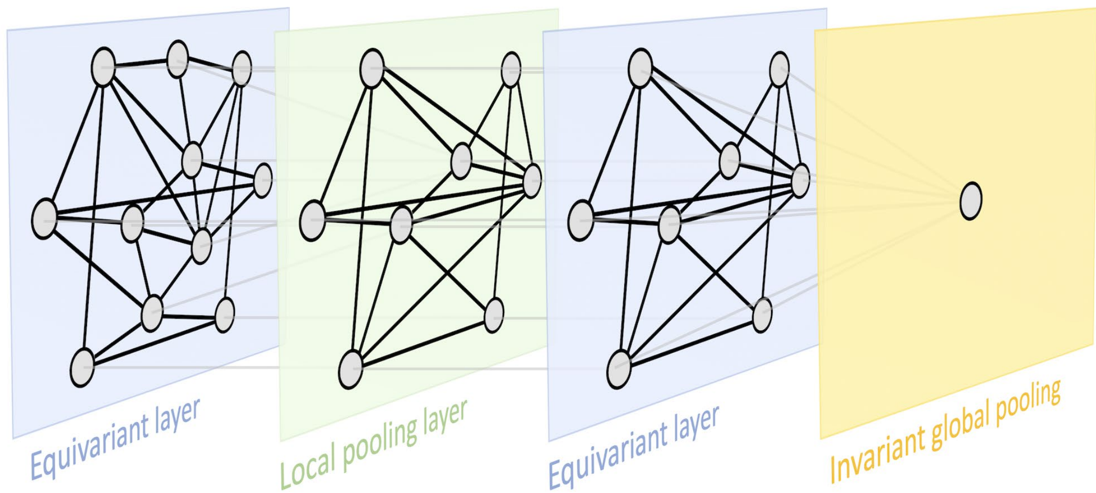
GEOMETRIC PRIORS - SCALE SEPARATION





$$f \approx f' \circ P, \text{ where } P : \mathcal{X}(\Omega) \to \mathcal{X}(\Omega')$$

GEOMETRIC DEEP LEARNING BLUEPRINT



Architecture	Domain Ω	Symmetry group $\mathfrak{G}$
CNN	Grid	Translation
Spherical CNN	Sphere / SO(3)	Rotation SO(3)
Intrinsic / Mesh CNN	Manifold	Isometry Iso(Ω) / Gauge symmetry SO(2)
GNN	Graph	Permutation Σ_n
Deep Sets	Set	Permutation Σ_n
Transformer	Complete Graph	Permutation Σ_n
LSTM	1D Grid	Time warping

permutation-invariant aggregation operator, e.g. sum

$$f(\mathbf{x}_i) = \phi\Big(\mathbf{x}_i, \underset{j \in \mathcal{N}_i}{\square} \psi(\mathbf{x}_j)\Big)$$

new feature of node i

learnable functions

importance of node j to the representation of i

$$f(\mathbf{x}_i) = \phi\Big(\mathbf{x}_i, \underset{j \in \mathcal{N}_i}{\square} c_{ij}\psi(\mathbf{x}_j)\Big)$$

"convolutional"

Defferard et al, 2016
Kipf, Welling, 2016
Smets et al, 2020

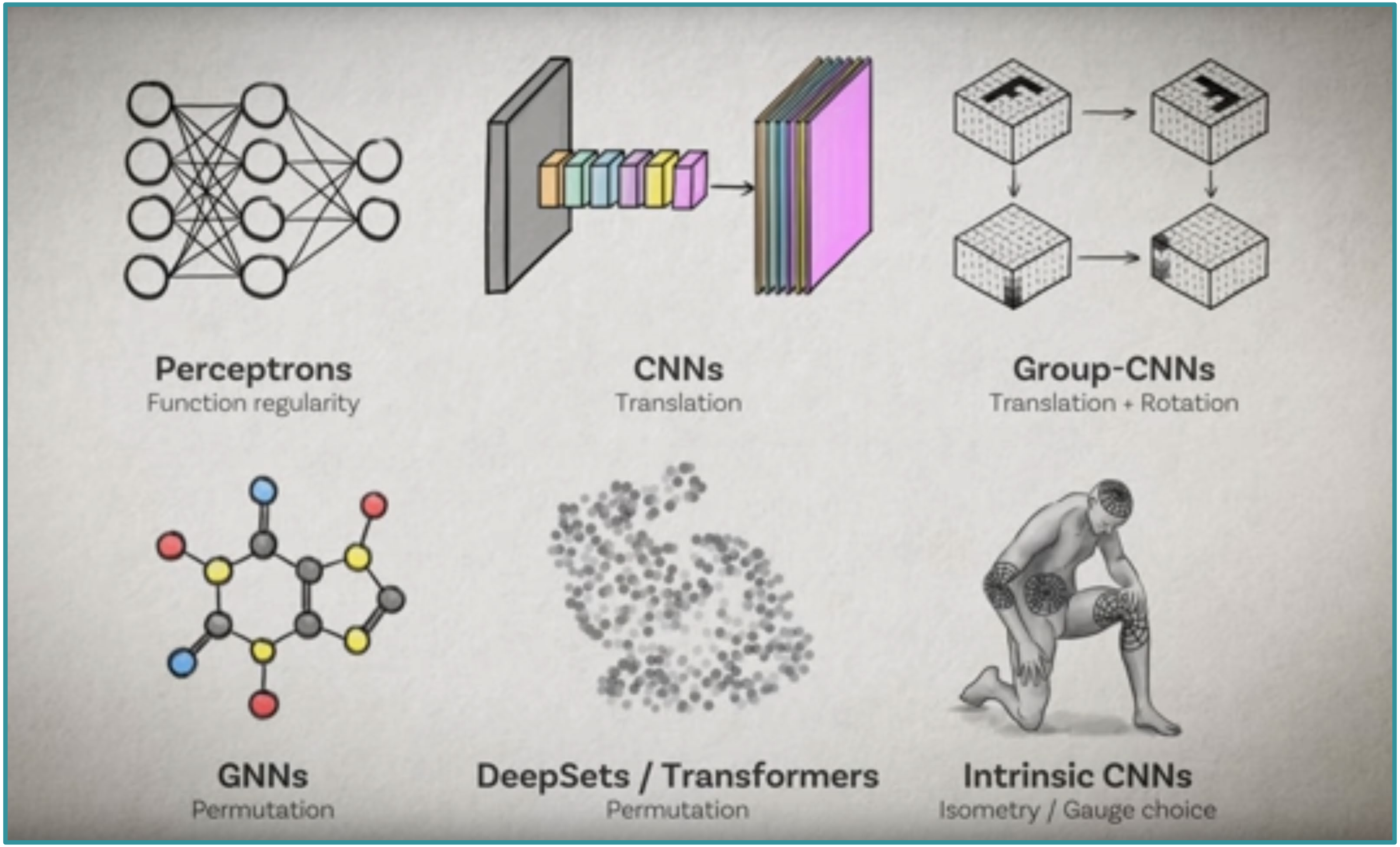
$$f(\mathbf{x}_i) = \phi\Big(\mathbf{x}_i, \underset{j \in \mathcal{N}_i}{\square} a(\mathbf{x}_i, \mathbf{x}_j)\psi(\mathbf{x}_j)\Big)$$

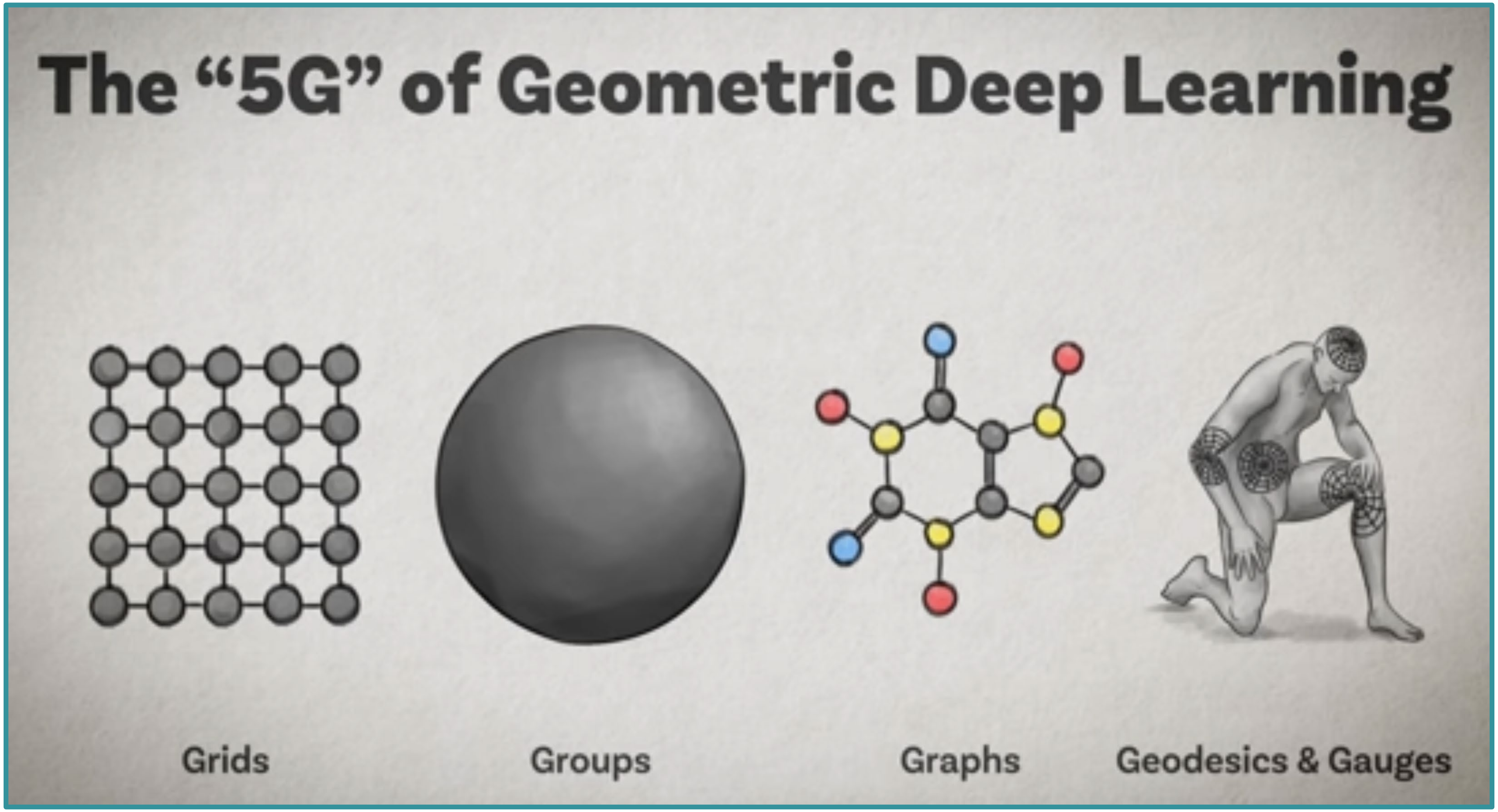
"attentional"

Monti et al, 2017
Velickovic et al, 2016

$$f(\mathbf{x}_i) = \phi\Big(\mathbf{x}_i, \underset{j \in \mathcal{N}_i}{\square} \psi(\mathbf{x}_i, \mathbf{x}_j)\Big)$$

"message passing"


Perceptrons
Function regularity
CNNs
Translation
Group-CNNs
Translation + Rotation
GNNs
Permutation
DeepSets / Transformers
Permutation
Intrinsic CNNs
Isometry / Gauge choice



Bronstein, et al – Geometric Deep Learning, Grids, Groups, Graphs, Geodesics, and Gauges (arXiv, 2021)

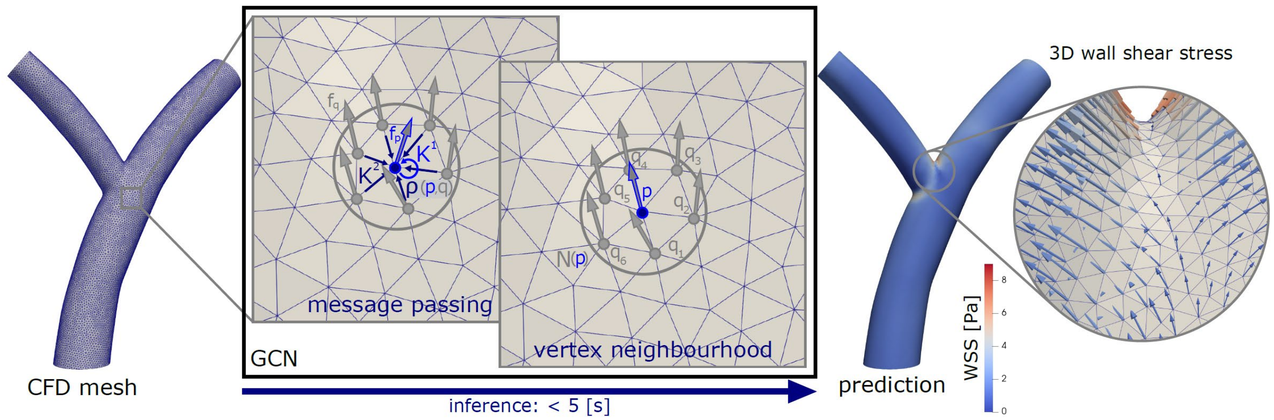
GEOMETRIC DEEP LEARNING FOR PRECISION MEDICINE

Graph kernels

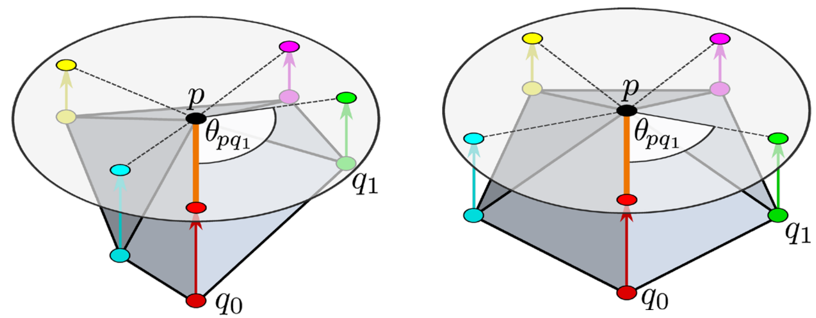
Isotropic: Graph convolutional networks

Anisotropic: Graph attention networks

Equivariant: Gauge equivariant networks





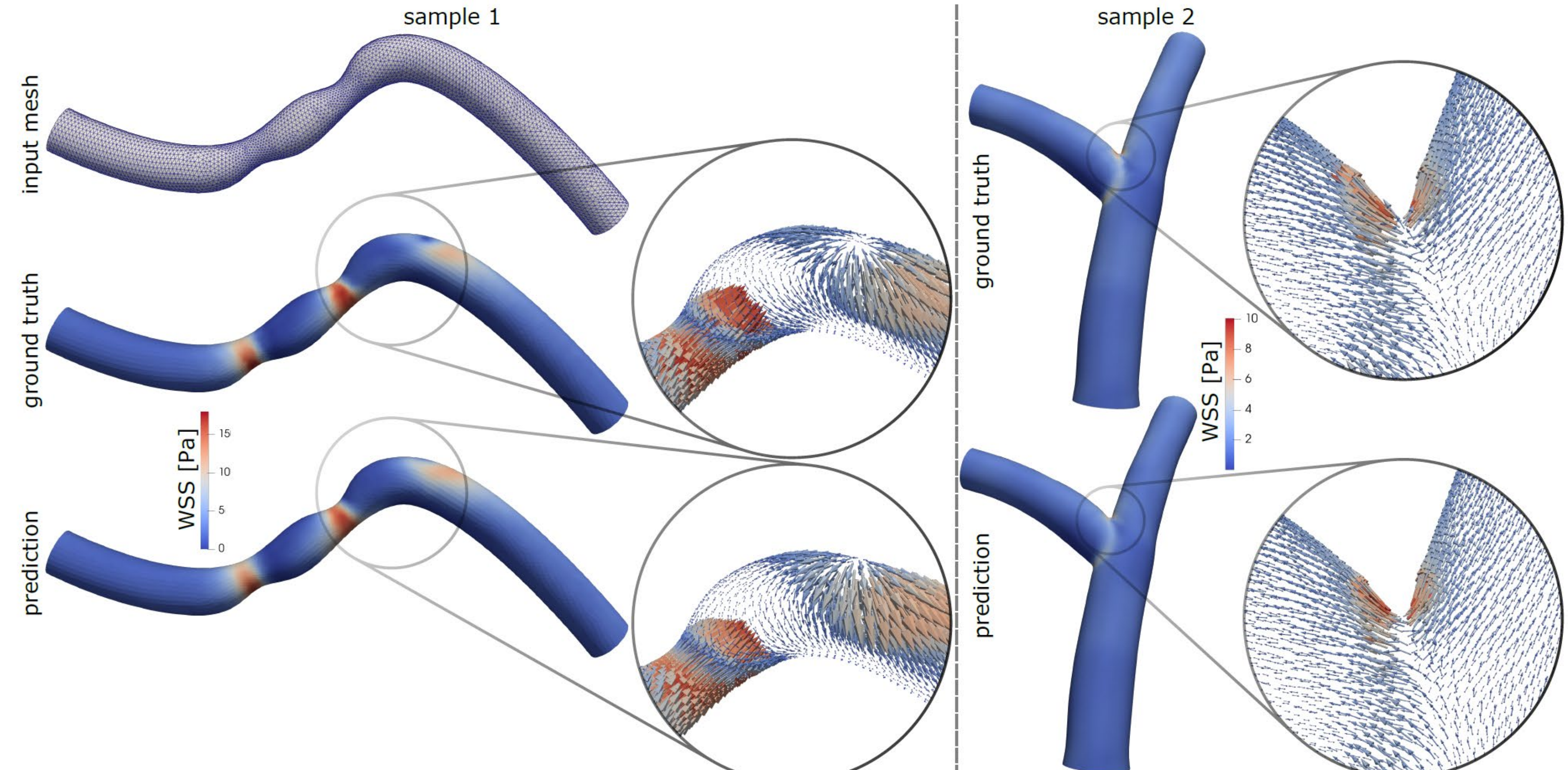



$$\left(\left(K^1, K^2(\cdot,\cdot)\right) * f\right)_p := f_p \cdot K^1 + \sum_{q \in N(p)} \rho(p,q) f_q \cdot K^2(p,q), \quad p \in \mathcal{V}$$

de Haan, Pim, et al. "Gauge equivariant mesh CNNs: Anisotropic convolutions on geometric graphs." ICLR (2021)

Milano, Francesco, et al. "Primal-dual mesh convolutional neural networks." NeurIPS (2020)

GEOMETRIC DEEP LEARNING FOR PRECISION MEDICINE



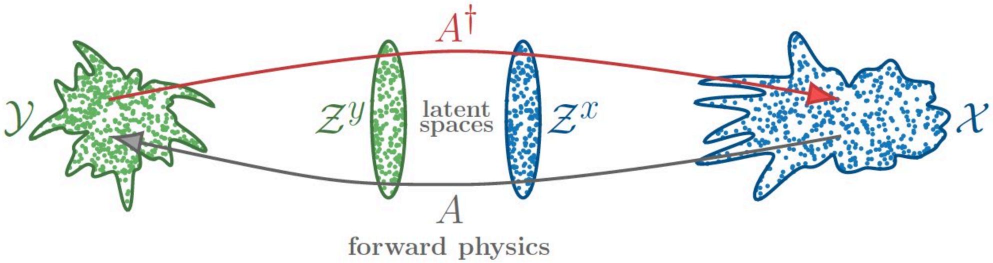
		NMAE [%]			ε [%]			$\triangle_{\mathrm{max}}$ [Pa]			$\triangle_{\mathrm{mean}}$ [Pa]		
		mean	median	75th	mean	median	75th	mean	median	75th	mean	median	75th
Single Arteries	SAGE-CNN	2.2	2.0	2.6	32.4	30.0	37.0	10.41	7.80	14.65	1.11	1.01	1.32
	FeaSt-CNN	1.2	1.1	1.5	19.0	18.6	22.4	5.83	5.13	8.17	0.60	0.57	0.77
	GEM-CNN	**0.6**	**0.6**	**0.8**	**9.9**	**9.5**	**11.6**	**3.94**	**3.68**	**5.46**	**0.32**	**0.31**	**0.41**
	SAGE-CNN†	10.5	9.6	12.8	149.2	128.1	181.2	26.73	23.96	36.17	5.31	4.84	6.50
	FeaSt-CNN†	8.3	7.5	10.1	123.7	111.1	152.9	25.63	22.93	34.52	4.22	3.82	5.13
	GEM-CNN†	0.6	0.6	0.8	9.8	9.4	11.4	3.80	3.39	5.53	0.32	0.31	0.42

Improved generalizability due to group equivariance

SUMMARY

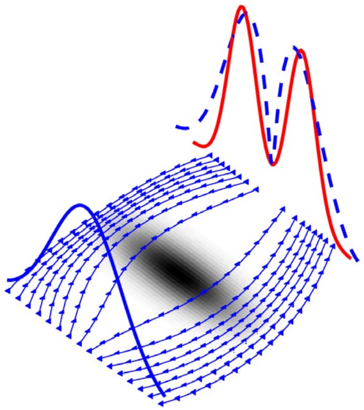
Inverse Problems and Deep Learning

+ Learned SVD via hybrid autoencoding



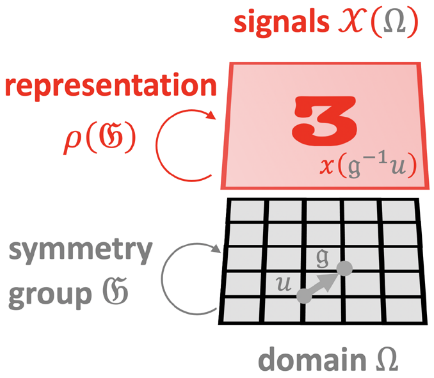
Deep Learning and Dynamics

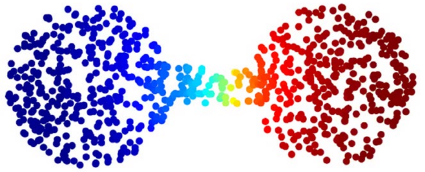
+ Optimal control, mean field games, neural SDEs
+ Normal form autoencoding for parameter dependent dynamics



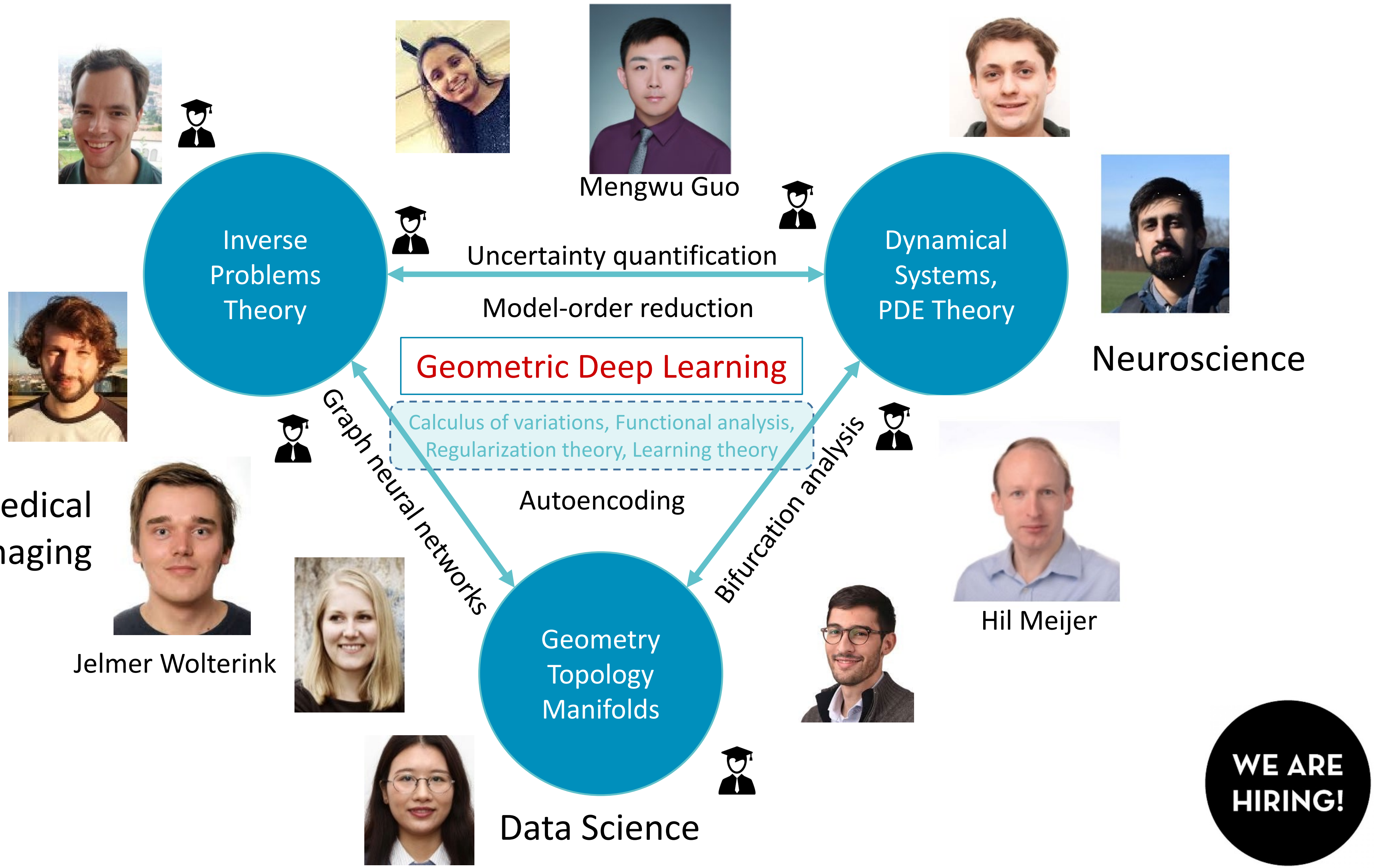
Geometric Deep Learning

+ Geometric priors, Gauge mesh CNNs

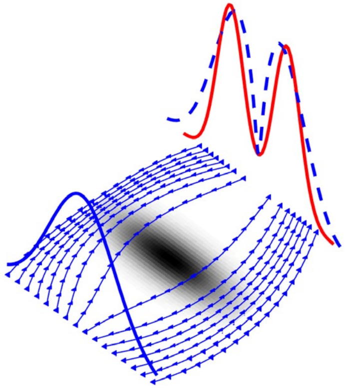


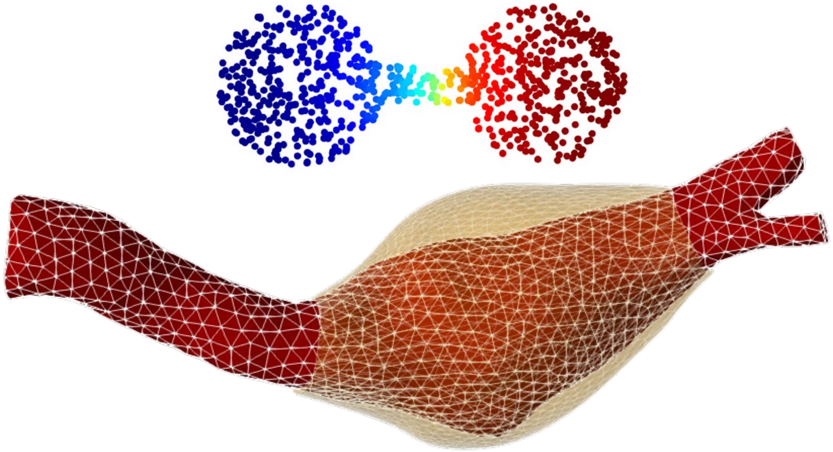


Intelligent Imaging & Analysis



OPEN QUESTIONS





Model-Informed GNNs - Are GNNs more generalizable for model-informed machine learning problems compared to classical neural networks?

- **Modeling**: How to combine GNNs properly with PDEs to achieve model-informed GNN?
- **Expressiveness**: Are flow-induced function spaces more expressive for GNNs with model-informed constraints than Barron spaces?
- **Generalization**: How does the double descent phenomena show up in GNNs (when we know more about the dynamics underlying)?

THANK YOU FOR YOUR ATTENTION

6 open positions https://www.utwente.nl/en/eemcs/sacs/vacancies/ (all Aug 15)

Please contact me directly if you have questions: c.brune@utwente.nl https://twitter.com/ChristophBrune

WE ARE HIRING!

References

- **Inverse Problems and Deep Learning**
  - Boink, Brune - Learned SVD - Deep Learning for Solving Inverse Problems via Hybrid Autoencoding (arXiv, 2021)
- **Model Discovery, Learning Dynamics**
  - Kalia, et al - Deep learning of normal form autoencoders for universal, parameter-dependent dynamics (Neurips, 2020)
  - Kalia, et al - Learning normal form autoencoders for data-driven discovery of universal, parameter-dependent governing equations (arXiv, 2021)
- **Deep Learning in Precision Medicine**
  - Zeune et al - Deep learning of circulating tumor cells (Nature Machine Intelligence, 2020)
  - Wolterink et al - Graph convolutional networks for coronary artery segmentation in cardiac CT angiography, GLMI 2019