Technische Universität Berlin

Stochastic Normalizing Flows and the Power of Patches in Inverse Problems

Gabriele Steidl

TU Berlin, Germany

Joint work with

Fabian Altekrüger, Paul Hagemann, Johannes Hertrich (TU Berlin)

Alexander Denker, Peter Maass (U Bremen)

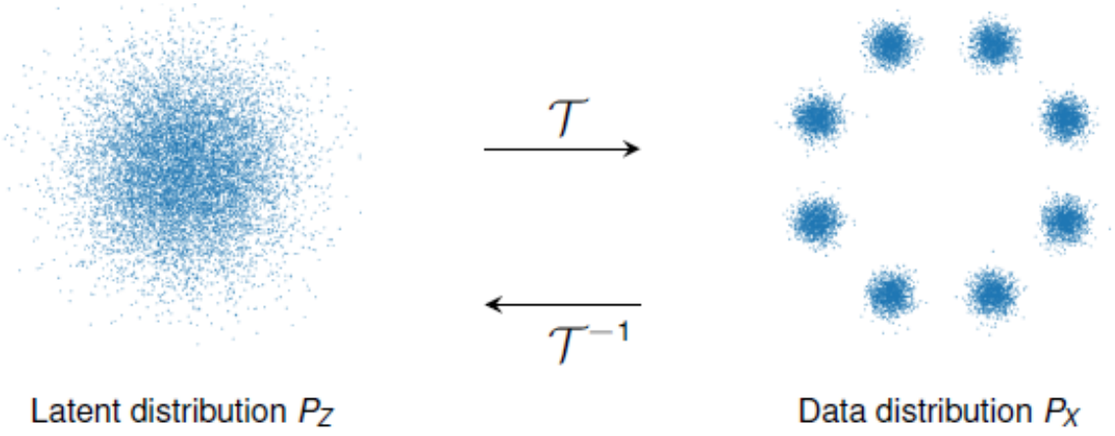
Outline

1. Normalizing Flows
2. Applications in Inverse Problems - The Power of Patches

3. Stochastic Normalizing Flows
4. Numerical Results
5. Conclusions

Generative neural networks

- Generative Adversarial Networks (GANs) (Goodfellow et al. 2014)
- Variational Auto-Encoders (Kingma, Welling 2014)
- Invertible Networks
  - Residual Networks (Behrmann, Chen et al. 2019)
  - Normalizing Flows - Directly Invertible Neural Networks (Dingh et al. 2017, Aridizzone et al. 2019)
  - for continuous normalizing flows
    see, e.g. Ruthotto/Haber 2020, Hagemann/Hertrich/St. Overview paper: Generalized Normalizing Flows via Markov Chains 2021



Normalizing Flows

Parameter depending concatenation of diffeomorphisms of the form

$$\mathcal{T}(\cdot;\theta) = \mathcal{T}_T \circ \cdots \circ \mathcal{T}_1,$$

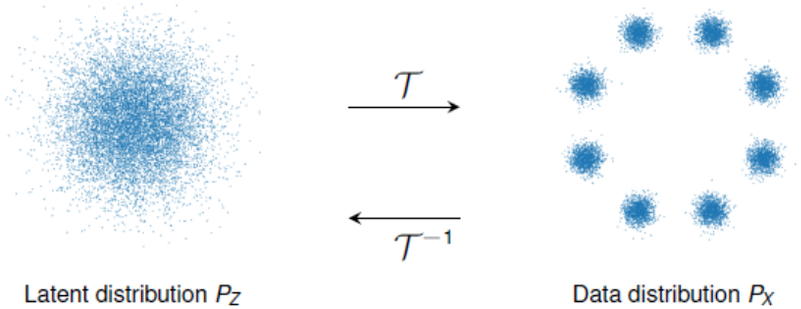
where, **up to some permutation matrices**, the $\mathcal{T}_k$ are of the form

$$\mathcal{T}_k(z_1, z_2) = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} := \begin{pmatrix} z_1 \, \mathrm{e}^{s_{k,2}(z_2)} + t_{k,2}(z_2) \\ z_2 \, \mathrm{e}^{s_{k,1}(x_1)} + t_{k,1}(x_1) \end{pmatrix}$$

with neural networks $s_{k,j}, t_{k,j}$, $k = 1, \ldots, L$, $x_j, z_j \in \mathbb{R}^{n_j}$, $j = 1, 2$.

$$\mathcal{T}_k^{-1}(x_1, x_2) = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} (x_1 - t_{k,2}(\xi_2)) \, \mathrm{e}^{-s_{k,2}(z_2)}, \\ (x_2 - t_{k,1}(x_1)) \, \mathrm{e}^{-s_{k,1}(x_1)} \end{pmatrix}$$

Aim: $P_X \approx \mathcal{T}_\# P_Z := P_Z \circ \mathcal{T}^{-1}$



Loss Function: Kullback-Leibler divergence (KL)

$$\mathcal{L}_{\mathrm{NF}}(\theta) = \mathrm{KL}(P_X, \mathcal{T}_{\#}P_Z) = \int \log\left(\frac{p_X(x)}{p_{\mathcal{T}_{\#}P_Z}(x)}\right) p_X(x)\mathrm{d}x$$

$$= \underbrace{\int \log\left(p_X(x)\right) p_X(x)\mathrm{d}x}_{\text{const}} - \int \log\left(p_{\mathcal{T}_{\#}P_Z}(x)\right) p_X(x)\mathrm{d}x$$

with transformation formula $p_{\mathcal{T}_{\#}P_Z} = p_Z\left(\mathcal{T}^{-1}\right)\left|\det\nabla\mathcal{T}^{-1}\right|$ and Gaussian distribution P_Z

$$\mathcal{L}_{\mathrm{NF}}(\theta) \sim -\int \log\left(p_Z\left(\mathcal{T}^{-1}(x)\right)\ \left|\det\nabla\mathcal{T}^{-1}(x)\right|\right) p_X(x)\mathrm{d}x$$

$$= -\mathbb{E}_{x\sim P_X}[\log p_Z\left(\mathcal{T}^{-1}\right)] - \mathbb{E}_{x\sim P_X}[\log\left(\left|\det\nabla\mathcal{T}^{-1}\right|\right)]$$

$$= \|\mathcal{T}^{-1}(\cdot)\|^2_{L_2(dP_x)} - \mathbb{E}_{x\sim P_X}[\left(\left|\det\nabla\mathcal{T}^{-1}\right|\right)]$$

Refs: SGD (Optimizer Adam: Kingma et al. 2015), Inertial Stoch. PALM (Hertrich/St. 2022)

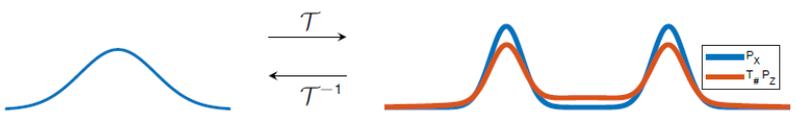
Remark on Kullback-Leibler Divergence

KL is (only) the Bregman distance of the Shannon entropy:

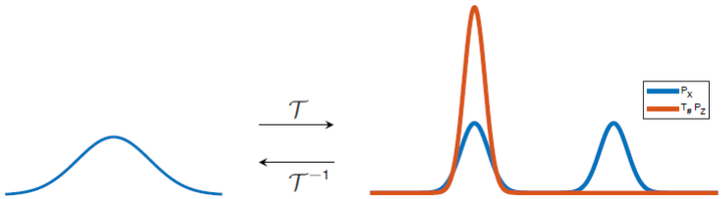
- $\mathrm{KL}(\mu, \nu) \geq 0$ for all $\mu, \nu \in \mathcal{P}(\mathbb{R}^d)$ with equality $\Leftrightarrow$ if $\nu = \mu$
- not symmetric and no triangular inequality
- finite if $P_X \ll P_Y$ and $\mathrm{KL}(P_X, P_Y) = +\infty$ otherwise

Different properties of $\mathrm{KL}(P_X, \mathcal{T}_\# P_Z)$ and $\mathrm{KL}(\mathcal{T}_\# P_Z, P_X)$:

- inverse problems: operator known versus operator not known
- mode covering (unrealistic samples possible) versus mode seeking (mode collapse)



$\mathrm{KL}(P_X, \mathcal{T}_\# P_Z)$



$\mathrm{KL}(\mathcal{T}_\# P_Z, P_X)$

Refs: Backward KL: Altekrüger/Hertrich: WPPFlows 2022)

Outline

1. Normalizing Flows
2. Applications in Inverse Problems - The Power of Patches
3. Stochastic Normalizing Flows
4. Numerical Results
5. Conclusions

Applications in Inverse Problems

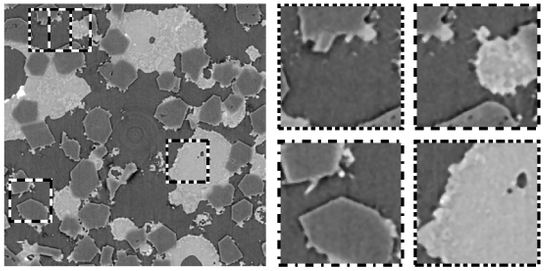
Inverse Problem for a certain class of images:

$$y = \text{noisy}(f(x))$$

Find solution as minimizer of variational model:

$$\mathcal{J}(x) = \underbrace{\mathcal{D}(f(x), y)}_{\text{data term}} + \lambda \underbrace{\mathcal{R}(x)}_{\text{regularizer}}, \qquad \lambda > 0$$

Idea: Learn regularizer from many patches $P_i(x_j)$, $i = 1, \ldots, N$ of few images x_j, $j = 1, \ldots, n$



The Power of Patches

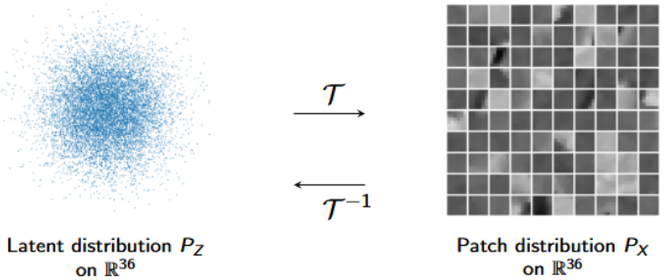
Refs: Buardes et al. 2005, Dabov et al. (BM3D) 2008; Lebrun/Morel (Denoising cuisine) 2013; Bortoli/Desolneux/Galerne/Leclaire 2019 ...,

Laus et al. 2017, Houdard et al. 2018; Hertrich et al. 2021 ...

Technische Universität Berlin

1. Learn Normalizing Flow $\mathcal{T} = \mathcal{T}(\theta, \cdot)$

$$\mathcal{L}_{\mathrm{NF}}(\theta) = \sum_{j=1}^{n} \sum_{i=1}^{N} \frac{\|\mathcal{T}^{-1}\big(P_i(x_j)\big)\|^2}{2} - \log|\det \nabla \mathcal{T}^{-1}\big(P_i(x_j)\big)|$$



2. Variational Model

$$\mathcal{J}(x) = \mathcal{D}\left(f(x), y\right) + \lambda\, \mathrm{patchNR}(x)$$

with learned $\mathcal{T} = \mathcal{T}(\theta, \cdot)$

$$\mathrm{patchNR}(x) := \Big(\frac{1}{N} \sum_{i=1}^{N} \frac{\|\mathcal{T}^{-1}\big(P_i(x)\big)\|^2}{2} - \log|\det \nabla \mathcal{T}^{-1}\big(P_i(x)\big)|\Big)$$

1	2
3	4
5	6
7	8
9	10
11	12
13	14
15	16
17	18
19	20
21	22
23	24
25	26
27	28
29	30
31	32
33	34
35	36

Bayesian Approach - MAP:

$$\begin{aligned}\hat{x} &\in \operatorname*{argmax}_x \log\left(p_{X|Y=y}(x)\right) \\ &= \operatorname*{argmin}_x \left\{ -\log\left(p_{Y|X=x}(y)\right) - \log\left(p_X(x)\right) \right\} \\ &= \operatorname*{argmin}_x \left\{ \underbrace{\mathcal{D}(x,y)}_{\text{data-fidelity term}} + \lambda \underbrace{\mathcal{R}(x)}_{\text{regularizer}} \right\}\end{aligned}$$

Proposition:

Let $P_Z = \mathcal{N}(0, I)$ and let $\mathcal{T}\colon \mathbb{R}^s \to \mathbb{R}^s$ be a bi-Lipschitz diffeomorphism. Then

$$\exp\left(-\lambda\, \text{patchNR}(x)\right) \in L^1(\mathbb{R}^d), \qquad \lambda > 0.$$

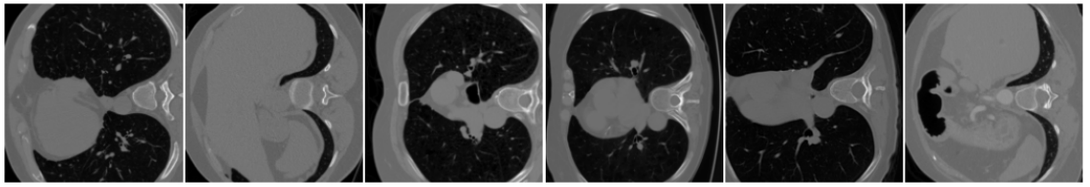
Advantages:

- Sparsity of training data: often not many training images are available - need just patches!
- Flexibility of operator and image classes: regularizer fits for every operator in the corresponding image classes

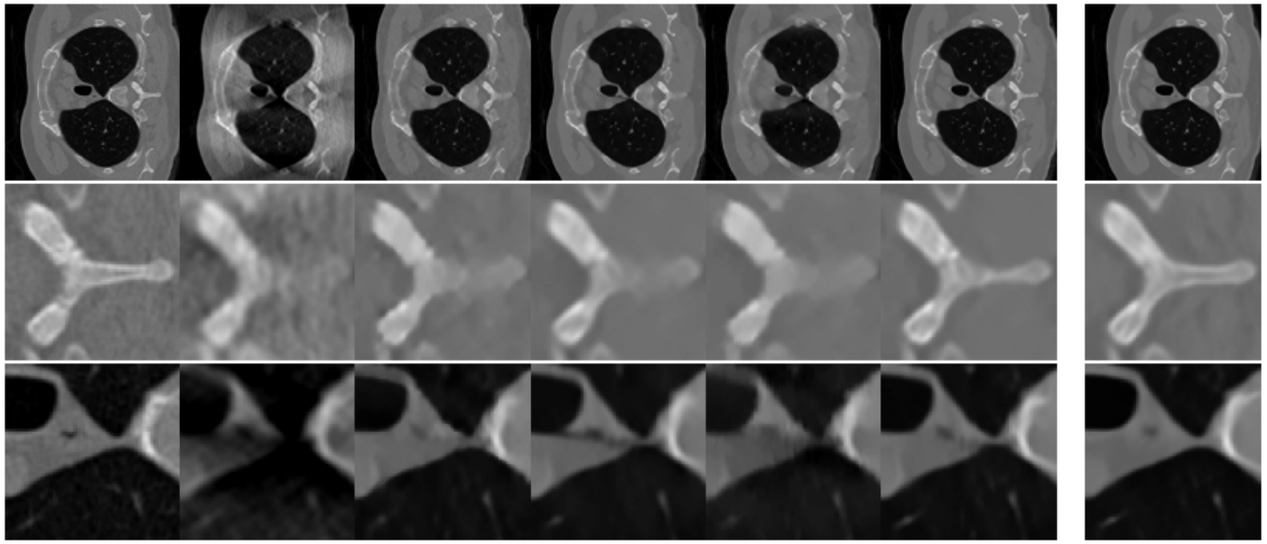
Computerized Tomography

Data term: Poisson (like) noise

$$\mathcal{D}(Ax, y) = \sum_{i=1}^{d} e^{-(Ax)_i} N_0 - e^{-y_i} N_0 \big(- (Ax)_i + \log(N_0) \big)$$

Regularizer:  $n = 6$ images

Results: Limited Angles - $36°/180°$ (same $\mathcal{R}$ for Low Dose Tomography!)



Refs: DIP-TV: Ulyanov et al. 2018, EPLL: Zoran et al. 2011, LocalAR: Prost et al. 2021, FBP+UNet: Jin et al. 2017, 35820 supervised samples

Computerized Limited Angle Tomography

	FBP	DIP + TV	EPLL	localAR	patchNR	FBP+UNet (s)
PSNR	21.96 ± 2.25	32.57 ± 3.25	32.78 ± 3.46	31.06 ± 2.95	**33.20** ± 3.55	33.75 ± 3.58
LPIPS	0.305 ± 0.117	**0.191** ± 0.165	0.216 ± 0.175	0.222 ± 0.166	0.201 ± 0.176	0.171 ± 0.134
SSIM	0.531 ± 0.097	0.803 ± 0.146	0.801 ± 0.151	0.779 ± 0.142	**0.811** ± 0.151	0.820 ± 0.140

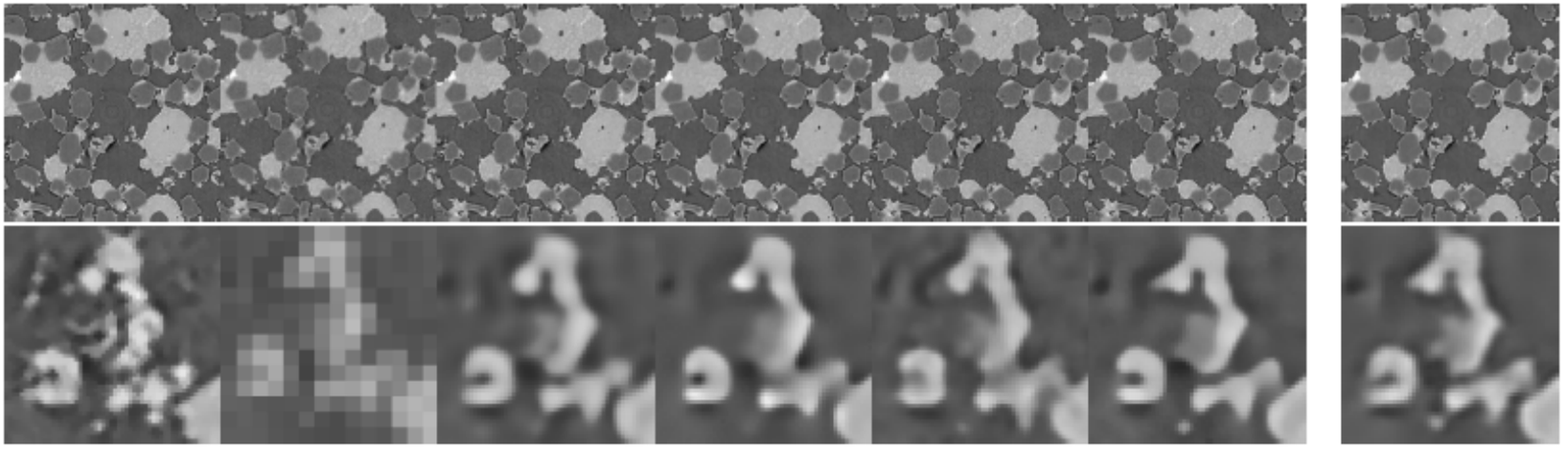
Refs LPIPS :Zhang, Efros, et al. 2018

Superresolution

Data term: Gaussian noise $\mathcal{D}(Ax, y) = \frac{1}{2}\|Ax - y\|^2$

Regularizer: learned from 1 high resolution synchotron image

Results:



HR | LR | DIP + TV | EPLL | WPP | patchNR | ACNN (supervised)

	bicubic (not shown)	DPIR (not shown)	DIP+TV	EPLL	WPP	patchNR	ACNN (supervised)
PSNR	25.63 ± 0.56	27.78 ± 0.53	27.99 ± 0.54	28.11 ± 0.55	27.80 ± 0.37	**28.53** ± 0.49	28.89 ± 0.53
LPIPS	0.406 ± 0.013	0.322 ± 0.015	0.191 ± 0.009	0.244 ± 0.012	0.167 ± 0.014	**0.159** ± 0.008	0.203 ± 0.011
SSIM	0.699 ± 0.012	0.770 ± 0.011	0.764 ± 0.007	0.779 ± 0.010	0.749 ± 0.011	**0.780** ± 0.008	0.804 ± 0.010

Data: D. Bernard (U Bordeaux) (ANR-DFG project)

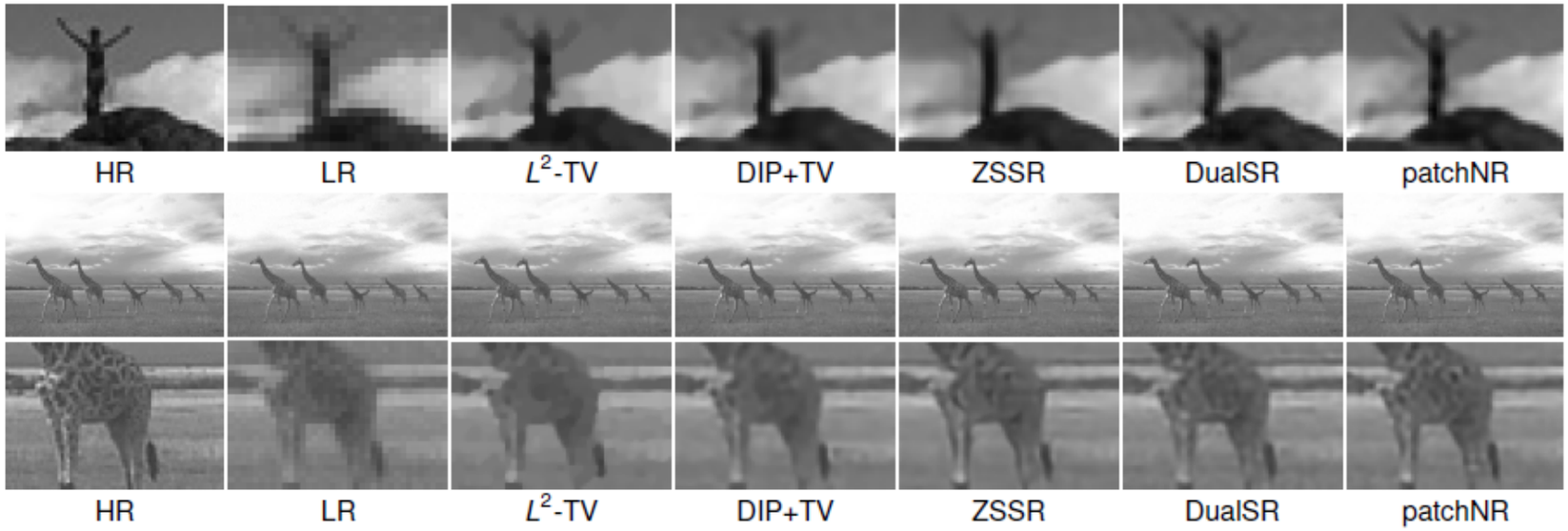
Refs: WPP: Hertrich et al. 2021, ACNN: Tian et al. 2021

Single-Shot Superresolution

Question: What to do when we do not have any high-resolution image for training the patchNR?

Assumption: Patch distribution of natural images is self-similar across the scales

Regularizer: learned from 1 low resolution image



	L^2-TV	DIP+TV	ZSSR	DualSR	patchNR
PSNR	28.35 ± 3.55	28.44 ± 3.69	28.83 ± 3.57	28.64 ± 3.47	**29.08** ± 3.58
LPIPS	**0.184** ± 0.073	0.215 ± 0.086	0.224 ± 0.085	0.216 ± 0.074	0.202 ± 0.076
SSIM	0.820 ± 0.072	0.821 ± 0.087	0.834 ± 0.066	0.829 ± 0.061	**0.846** ± 0.061

Refs: Glasner et al. 2009, ZSSR: Shocher et al. 2018 , Dual SR: Emad et al. 2021

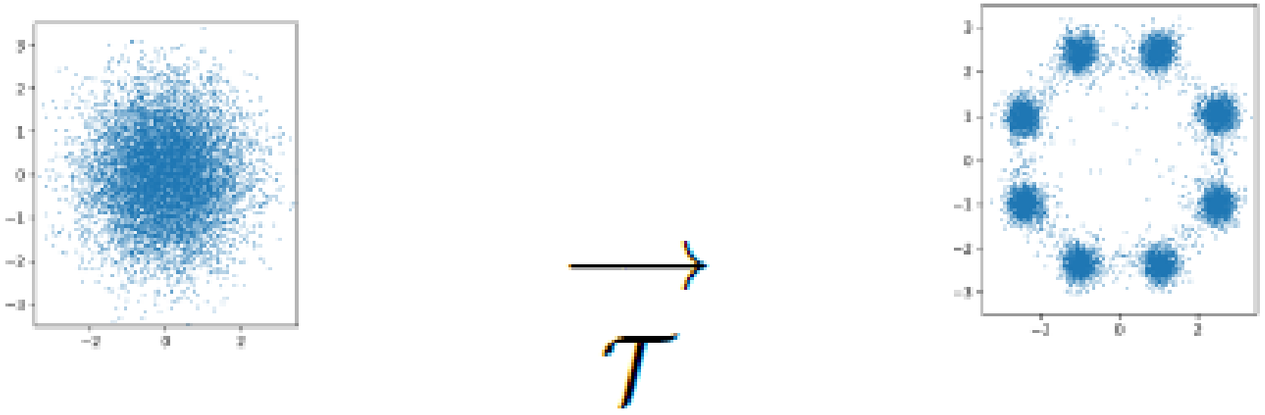
Outline

1. Normalizing Flows
2. Applications in Inverse Problems - The Power of Patches

3. Stochastic Normalizing Flows
4. Numerical Results
5. Conclusions

3. Stochastic Normalizing Flows

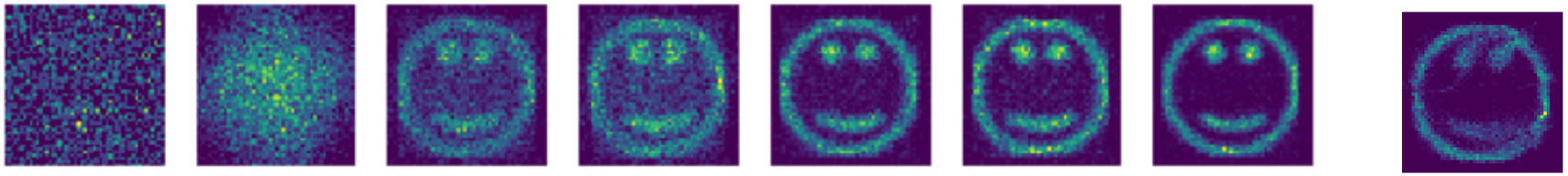
- **Drawback: Lack in expressiveness!**: unimodal distributions are hard to map to multimodal and heavy tailed ones
- Normalizing flows mapping unimodal distributions to multimodal ones must have an exploding Lipschitz constant!
  (Refs Lipschitz: Nagarajan et al. 2018, Gulrajani et al. 2018, Hagemann/Neumayer 2021; Stéphanovitch et al. 2022; Salmona et al. 2022)



- (One) **Solution:** Application of stochastic steps within a unifying framework of Markov chains
  (Refs SNF without Markov chains: Wu/Köhler/Noé 2020, Nielsen/Welling et al. 2021)

Stochastic Normalizing Flows

- Our framework of stochastic normalizing flows includes
  - Langevin layer,
  - Metropolis-Hastings (MH) layer,
  - Metropolis-adjusted Langevin (MALA) layer,
  - VAE layer,
  - diffusion normalizing flow layer (Song et al. 2020)



Alternation of INNs with MCMC layers, starting with the first one.

Up to now the only mathematically sound way to combine these layers is via Markov chains!

- A Markov kernel $\mathcal{K}\colon \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0,1]$ is a mapping such that
  - i) $\mathcal{K}(\cdot, B)$ is measurable for any $B \in \mathcal{B}(\mathbb{R}^d)$, and
  - ii) $\mathcal{K}(x, \cdot)$ is a probability measure for any $x \in \mathbb{R}^d$.
- For μ on $\mathcal{P}(\mathbb{R}^d)$, the measure $\mu \times \mathcal{K}$ on $\mathbb{R}^d \times \mathbb{R}^d$ is defined by

$$(\mu \times \mathcal{K})(A \times B) := \int_A \mathcal{K}(x, B) d\mu(x).$$

- Regular conditional distribution of X given Y: P_Y-a.s. unique Markov kernel $P_{Y|X=\cdot}(\cdot)$ with

$$P_X \times P_{Y|X=\cdot}(\cdot) = P_{(X,Y)}$$

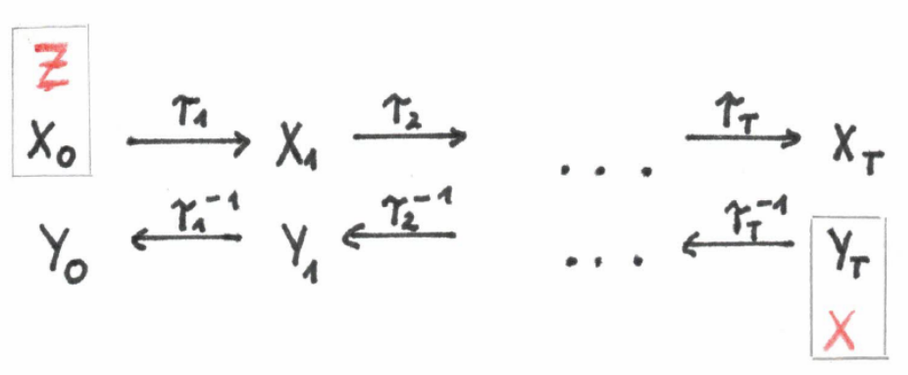
- $(X_0, \ldots, X_T)$ is called a Markov chain, if there exist Markov kernels

$$\mathcal{K}_t = P_{X_t|X_{t-1}=\cdot}(\cdot)\colon \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0,1]$$

such that

$$P_{(X_0,\ldots,X_T)} = P_{X_0} \times \mathcal{K}_1 \times \cdots \times \mathcal{K}_T.$$

Ordinary Normalizing Flow: $\mathcal{T}(\cdot;\theta) = \mathcal{T}_T \circ \cdots \circ \mathcal{T}_1$,



- $\mathcal{T}_t \colon \mathbb{R}^d \to \mathbb{R}^d$ generate a pair of Markov chains $\big((X_0, ..., X_T), (Y_T, ..., Y_0)\big)$

$$X_0 \sim P_Z, \quad X_t = \mathcal{T}_t(X_{t-1}) \quad \text{and}$$
$$Y_T \sim P_X, \quad Y_{t-1} = \mathcal{T}_t^{-1}(Y_t).$$

- Markov kernels $\mathcal{K}_t(x, \cdot) = P_{X_t | X_{t-1}=x} = \delta_{\mathcal{T}_t(x)}$ and $\mathcal{R}_t(x, \cdot) = \delta_{\mathcal{T}_t^{-1}(x)}$
- Minimize the ,,whole path"

$$\begin{aligned}\mathcal{L}_{\mathrm{NF}}(\theta) &= \mathrm{KL}(P_X, P_{\mathcal{T}_\# Z}) = \mathrm{KL}(P_{Y_T}, P_{X_T}) \\ &= \mathrm{KL}\left(P_{(Y_0,\ldots,Y_T)}, P_{(X_0,\ldots,X_T)}\right) \qquad \text{in general } \leq\end{aligned}$$

Stochastic normalizing flow (SNF) = pair of Markov chains

$$\big((X_0, \dots, X_T), (Y_T, ..., Y_0)\big)$$

that minimizes the loss function

$$\mathcal{L}_{\mathsf{SNF}}(\theta) = \mathrm{KL}(P_{(Y_0,...,Y_T)}, P_{(X_0,...,X_T)})$$

and has the following properties:

P1) P_{X_t}, P_{Y_t} have the densities $p_{X_t}, p_{Y_t} \colon \mathbb{R}^{d_t} \to \mathbb{R}_{>0}$ for any $t = 0, ..., T$.

P2) There exist Markov kernels $\mathcal{K}_t = P_{X_t|X_{t-1}}$ and $\mathcal{R}_t = P_{Y_{t-1}|Y_t}$, $t = 1, ..., T$ such that

$$P_{(X_0,...,X_T)} = P_{X_0} \times P_{X_1|X_0} \times \cdots \times P_{X_T|X_{T-1}},$$
$$P_{(Y_T,...,Y_0)} = P_{Y_T} \times P_{Y_{T-1}|Y_T} \times \cdots \times P_{Y_0|Y_1}.$$

P3) For P_{X_t}-almost every $x \in \mathbb{R}^{d_t}$, the measures $P_{Y_{t-1}|Y_t=x}$ and $P_{X_{t-1}|X_t=x}$ are absolutely continuous with respect to each other,

Important: We don't need that all conditional distributions $P_{X_t|X_{t-1}}$ have densities! Indeed they don't exist for INNs and MCMC (MH, MALA) layer.

Learning SNFs

Loss function: $\mathcal{L}_{\mathrm{SNF}}(\theta) = \mathrm{KL}(P_{(Y_0,\dots,Y_T)}, P_{(X_0,\dots,X_T)})$

Theorem: With $f_t(\cdot, x_t) = \frac{\mathrm{d}P_{Y_{t-1}|Y_t=x_t}}{\mathrm{d}P_{X_{t-1}|X_t=x_t}}$ we have

$$\mathcal{L}_{\mathrm{SNF}}(\theta) = \mathbb{E}_{(x_0,\dots,x_T)\sim P_{(Y_0,\dots,Y_T)}}\Big[\log\Big(\frac{p_X(x_T)}{p_Z(x_0)}\prod_{t=1}^{T}\frac{f_t(x_{t-1},x_t)p_{X_{t-1}(x_{t-1})}}{p_{X_t(x_t)}}\Big)\Big],$$

The right-hand side can be computed for the different layers:

- Deterministic layer: $\frac{p_{X_{t-1}}(x_{t-1})}{p_{X_t}(x_t)} = \frac{1}{|\nabla \mathcal{T}_t^{-1}(x_t)|}$ and $f_t(x_{t-1}, x_t) = 1$
- Langevin layer: $\frac{f_t(x_{t-1},x_t)p_{X_{t-1}}(x_{t-1})}{p_{X_t}(x_t)} = \exp\Big(\frac{1}{2}(\|\eta_t\|^2 - \|\tilde{\eta}_t\|^2)\Big)$,
  with proposal density $p_t\colon \mathbb{R}^d \to \mathbb{R}_{>0}$, $u_t = -\log(p_t)$ and
  $\eta_t := \frac{1}{a_2}\big(x_{t-1} - x_t - a_1\nabla u_t(x_{t-1})\big)$, $\tilde{\eta}_t := \frac{1}{a_2}\big(x_{t-1} - x_t + a_1\nabla u_t(x_t)\big)$
- MCMC layer: $\frac{f_t(x_{t-1},x_t)p_{X_{t-1}}(x_{t-1})}{p_{X_t}(x_t)} = \frac{p_t(x_{t-1})}{p_t(x_t)}$
- VAEs: $\frac{f_t(x_{t-1},x_t)p_{X_{t-1}}(x_{t-1})}{p_{X_t}(x_t)} = \frac{q_\phi(x_{t-1}|x_t)}{p_\theta(x_t|x_{t-1})}$
- Diffusion layer: $\frac{f_t(x_{t-1},x_t)p_{X_{t-1}}(x_{t-1})}{p_{X_t}(x_t)} = \exp\Big(\frac{1}{2}(\|\eta_t\|^2 - \|\tilde{\eta}_t\|^2)\Big)$,, where
  $\eta_t := \frac{1}{\sqrt{\epsilon}g_{t-1}}\big(x_{t-1} - x_t + \epsilon f_{t-1}(x_{t-1})\big)$, $\tilde{\eta}_t := \frac{1}{\sqrt{\epsilon}g_t}\big(x_{t-1} - x_t - \epsilon(f_t(x_t) - g_t^2 s_t(x_t))\big)$

Langevin Layer

- Markov chain:

$$X_t := X_{t-1} - a_1 \nabla u_t(X_{t-1}) + a_2 \xi_t,$$

where $a_1, a_2 > 0$ are some predefined constants, $\xi_t \sim \mathcal{N}(0, I)$ and $u_t(x) := -\log(p_t(x))$ with proposal density $p_t \colon \mathbb{R}^d \to \mathbb{R}_{>0}$

- Transition kernels $\mathcal{K}_t = \mathcal{R}_t$: $\mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0, 1]$:

$$\mathcal{K}_t(x, \cdot) = \mathcal{N}(x - a_1 \nabla u_t(x), a_2^2 I).$$

- Background: one step of explicit Euler discretization of the overdamped Langevin dynamics

$$X(0) = X^0,$$
$$dX(\tau) = -\nabla \Psi(x) d\tau + \sqrt{2\beta^{-1}} dW(\tau)$$

- Fokker-Planck equation

$$\rho(x, 0) = \rho^0(x)$$
$$\frac{\partial \rho}{\partial \tau} = \operatorname{div}(\rho \nabla \Psi(x)) + \beta^{-1} \Delta \rho$$

stationary solution as $\tau \to \infty$: $p_t(x) = C^{-1} \mathrm{e}^{-\beta \Psi(x)}$, C normalizing factor

MCMC Layer

- Markov chain:

$$X_t = X_{t-1} + 1_{[U,1]}\left(\alpha_t(X_{t-1}, X_{t-1} + \xi_t)\right)\, \xi_t, \quad \xi_t \sim \mathcal{N}(0, \sigma^2 I)$$

where

$$\alpha_t(x, y) := \min\left\{1, \frac{p_t(y)}{p_t(x)}\right\}$$

with proposal density $p_t \colon \mathbb{R}^d \to \mathbb{R}_{>0}$

- Transition kernels $\mathcal{K}_t = \mathcal{R}_t \colon \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0, 1]$

$$\mathcal{K}_t(x, A) := \int_A \mathcal{N}(y; x, \sigma^2 I)\alpha_t(x, y)\mathrm{d}y + \delta_x(A) \int_{\mathbb{R}^d} \mathcal{N}(y; x, \sigma^2 I)\left(1 - \alpha_t(x, y)\right)\mathrm{d}y$$

- one sampling step of Metropolis-Hastings algorithm (known to sample from the proposal distibution)

VAEs Layer

1. Autoencoders (AE): dimensionality reduction technique inspired by PCA

Neural network pair (E, D):

- encoder $E = E_\phi \colon \mathbb{R}^d \to \mathbb{R}^n$, $d > n$
- decoder $D = D_\theta \colon \mathbb{R}^n \to \mathbb{R}^d$

Minimize loss function:

$$\mathcal{L}_{\mathsf{AE}}(\phi, \theta) := \mathbb{E}_{x \sim P_X}\left[\|x - D_\theta(E_\phi(x))\|^2\right]$$

Refs: PhD Gonzalez et al. 2021

2. Variational Autoencoders (VAE): approximate P_X using simpler P_Z

- encoder $D(z) = D_\theta(z) := (\mu_\theta(z), \Sigma_\theta(z))$

$$\mathcal{K}(z, \cdot) := \mathcal{N}(\mu_\theta(z), \Sigma_\theta(z)), \quad p_\theta(x|z) = \mathcal{N}(x; \mu_\theta(z), \Sigma_\theta(z))$$

- decoder $E(x) = E_\phi(x) := (\mu_\phi(x), \Sigma_\phi(x))$

$$\mathcal{R}(x, \cdot) := \mathcal{N}(\mu_\phi(x), \Sigma_\phi(x)), \quad q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \Sigma_\phi(x))$$

- SNF $((X_0, X_1), (Y_1, Y_0))$ with

$$\mathcal{L}_{\text{SNF}}(\theta, \phi) = \mathbb{E}_{(z,x)\sim P_{(Y_0,Y_1)}}\Big[\log\Big(\frac{p_X(x) f_1(z,x)}{p_{X_1}(x)}\Big)\Big]$$

$$= -\mathbb{E}_{x\sim P_X}\Big[\mathcal{L}_{\theta,\phi}(x)\Big] + \underbrace{\mathbb{E}_{x\sim P_X}[\log(p_X(x))]}_{const}$$

with the evidence lower bound (ELBO) function

$$\mathcal{L}_{\theta,\phi}(x) := \mathbb{E}_{z\sim q_\phi(\cdot|x)}\big[\log(p_\theta(x|z)p_Z(z)) - \log(q_\phi(z|x))\big]$$

Diffusion Layer

SDE: $\mathrm{d}X_t = g_t(X_t)\mathrm{d}t + h_t\mathrm{d}B_t$

with respect to the Brownian motion B_t, such that it holds approximately $X_S \sim P_X$ for some $S > 0$ and some data distribution P_X.

Explicit Euler discretization with step size $\epsilon > 0$:

$$X_t = X_{t-1} + \epsilon g_{t-1}(X_{t-1}) + \sqrt{\epsilon} h_{t-1}\xi_{t-1}, \quad t = 1, ..., T,$$

where $\xi_{t-1} \sim \mathcal{N}(0, I)$ is independent of $X_0, ..., X_{t-1}$

Markov kernel:

$$\mathcal{K}_t(x, \cdot) = P_{X_t|X_{t-1}=x} = \mathcal{N}(x + \epsilon g_{t-1}(x), \epsilon h_{t-1}^2).$$

Song et. al. parametrized the functions $g_t \colon \mathbb{R}^d \to \mathbb{R}^d$ by some a-priori learned score network and achieved competitive performance in image generation. Motivated by the time-reversal process of the SDE Zhang/Chen 2021 introduced the backward layer

$$\mathcal{R}_t(x, \cdot) = P_{Y_{t-1}|Y_t=x} = \mathcal{N}(x + \epsilon(g_t(x) - h_t^2 s_t(x)), \epsilon h_t^2)$$

Conditional Stochastic Normalizing Flows

Inverse Problems:

$$Y = F(x) + \Xi, \quad \Xi \sim \mathcal{N}(0, \sigma) \Rightarrow Y \sim \mathcal{N}(F(x), \sigma)$$
$$Y = F(X) + \Xi$$

Aim: Sample from $P_{X|Y=y}$ for some y

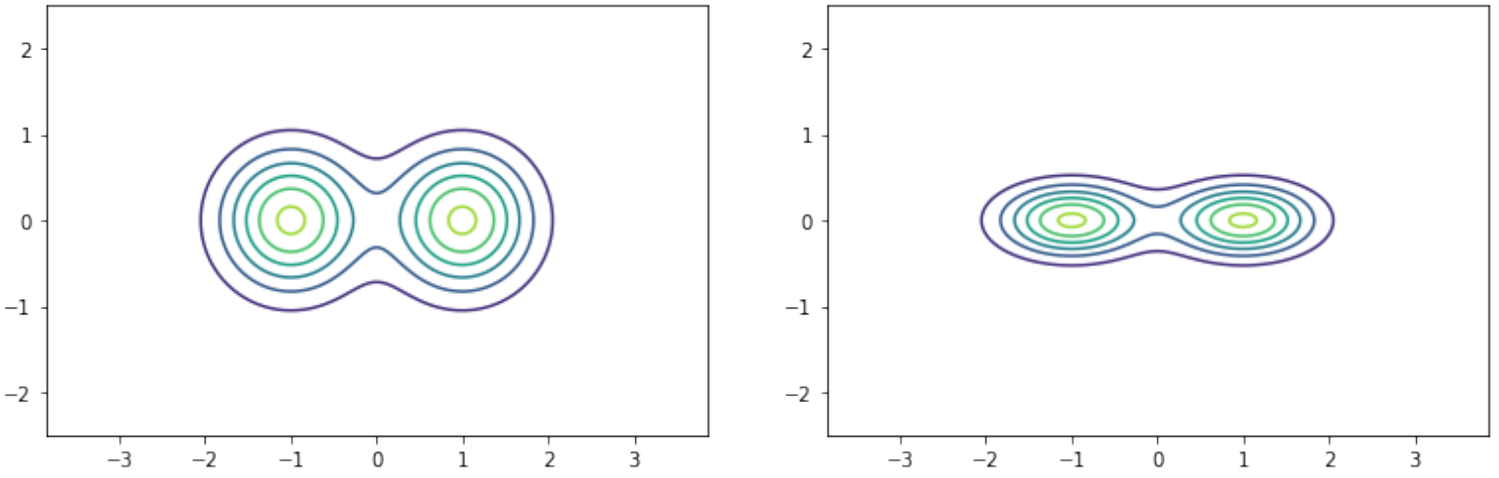


Illustration of the prior density p_X (left) and the posterior density $p_{X|Y=y}$ for $y = 0$ (right) within the inverse problem with $F(x_1, x_2) = x_2$ and $\eta \sim \mathcal{N}(0, 0.1^2)$

Conditional Stochastic Normalizing Flows

Conditional SNF conditioned to $Y = y$ is pair of Markov chains

$$((X_0, ..., X_T), (Y_T, ..., Y_0))$$

with the following properties:

cP1) the conditional distributions $P_{X_t|Y=y}$ and $P_{Y_t|Y=y}$ have densities

$$p_{X_t}(y, \cdot)\colon \mathbb{R}^{d_t} \to \mathbb{R}_{>0}, \quad p_{Y_t}(y, \cdot)\colon \mathbb{R}^{d_t} \to \mathbb{R}_{>0}$$

for P_Y-almost every y and all $t = 1, \dots, T$,

cP2) for P_Y-almost every y, there exist Markov kernels $\mathcal{K}_t\colon \mathbb{R}^{\tilde{d}} \times \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0, 1]$ and $\mathcal{R}_t\colon \mathbb{R}^{\tilde{d}} \times \mathbb{R}^d \times \mathcal{B}(\mathbb{R}^d) \to [0, 1]$ such that

$$P_{(X_0,\dots,X_T)|Y=y} = P_{X_0} \times \mathcal{K}_1(y, \cdot, \cdot) \times \cdots \times \mathcal{K}_T(y, \cdot, \cdot),$$
$$P_{(Y_T,\dots,Y_0)|Y=y} = P_{Y_T} \times \mathcal{R}_T(y, \cdot, \cdot) \times \cdots \times \mathcal{R}_1(y, \cdot, \cdot).$$

cP3) for P_{Y,X_t}-almost every pair $(y, x) \in \mathbb{R}^{\tilde{d}} \times \mathbb{R}^d$, the measures $P_{Y_{t-1}|Y_t=x,Y=y}$ and $P_{X_{t-1}|X_t=x,Y=y}$ are absolute continuous with respect to each other.

Outline

1. Normalizing Flows
2. Applications in Inverse Problems - The Power of Patches
3. Stochastic Normalizing Flows
4. Numerical Results
5. Conclusions

Sampling from a Posterior Distribution

Lemma: Let $X \sim \sum_{k=1}^{K} w_k \mathcal{N}(m_k, \Sigma_k)$. Suppose that

$$Y = AX + \Xi, \quad A : \mathbb{R}^d \to \mathbb{R}^{\tilde{d}}, \quad \Xi \sim N(0, b^2 I)$$

Then

$$P_{X|Y=y} \propto \sum_{k=1}^{K} \tilde{w}_k \mathcal{N}(\cdot|\tilde{m}_k, \tilde{\Sigma}_k),$$

where

$$\tilde{\Sigma}_k := (\tfrac{1}{b^2} A^\mathsf{T} A + \Sigma_k^{-1})^{-1}, \qquad \tilde{m}_k := \tilde{\Sigma}_k (\tfrac{1}{b^2} A^\mathsf{T} y + \Sigma_k^{-1} \mu_k).$$

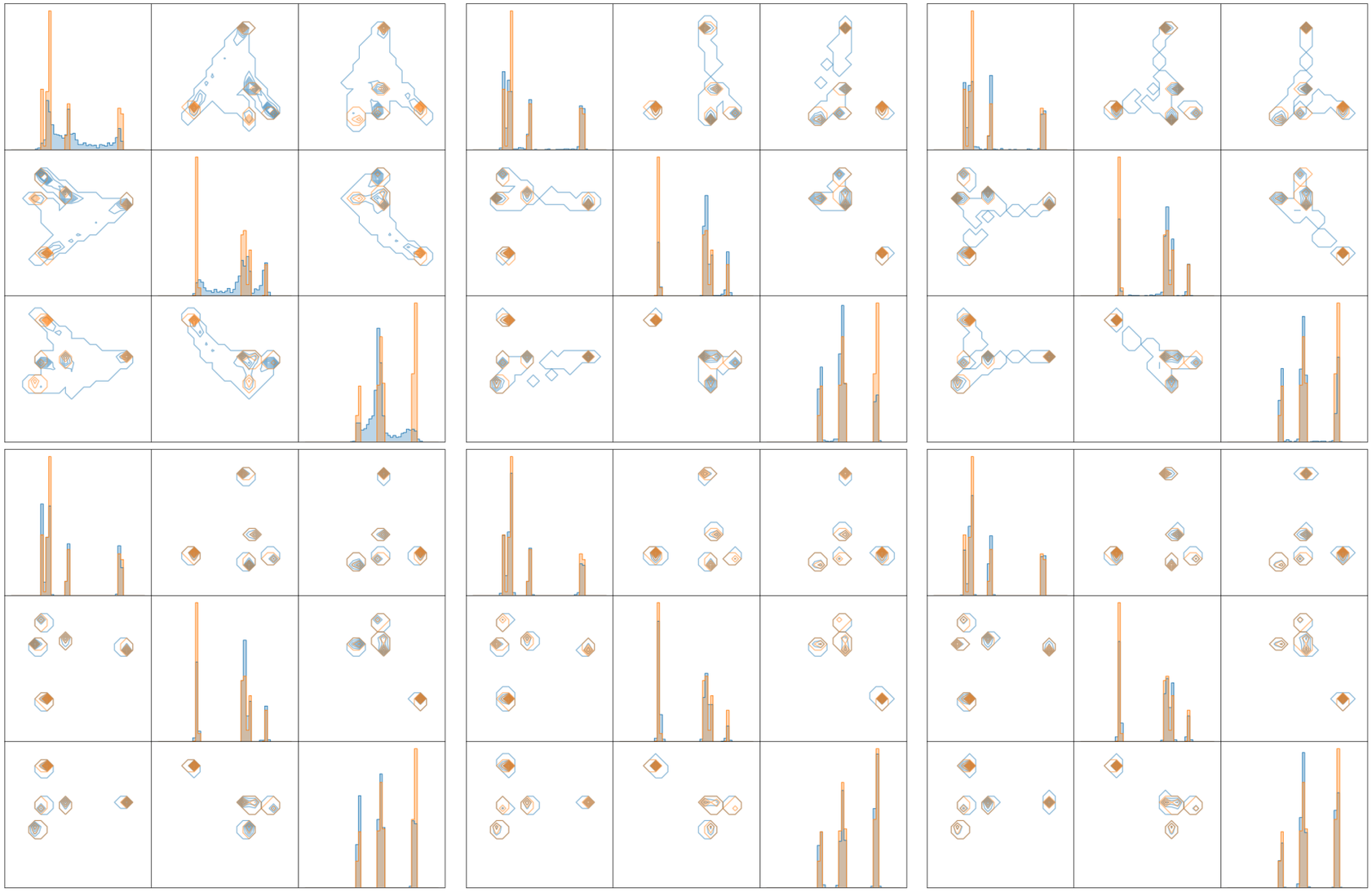
and

$$\tilde{w}_k := \frac{w_k}{|\Sigma_k|^{\frac{1}{2}}} \exp\left(\frac{1}{2}(\tilde{m}_k \tilde{\Sigma}_k^{-1} \tilde{m}_k - m_k \Sigma_k^{-1} m_k)\right).$$

Experiment: with GMM in $\mathbb{R}^{100}$ with $K = 5$ mixture components

- A diagonal matrix
- $\Xi \sim \mathcal{N}(0, 0.05 I)$
- proposal densities: $p_t^y(x) = c_y \left(p_Z(x) p_{X|Y=y}(x)\right)^{1/2}$

1. Approximation of the Posterior for Gaussian Mixtures



Top: INN, INN + MALA, VAE, Bottom: INN + VAE, VAE+MALA, INN+VAE+MALA.

Method	INN	INN+MALA	VAE	VAE+INN	VAE+MALA	INN+VAE+MALA
W_1	3.55	2.92	1.22	1.18	0.8	0.82

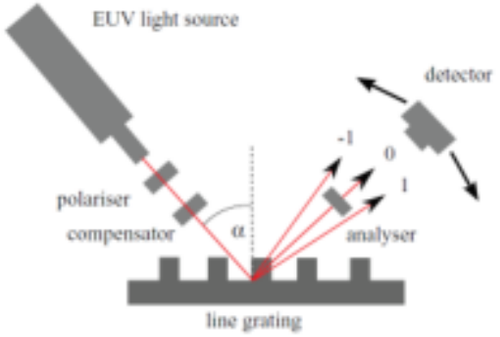
2. Example from Scatterometry

- parameters in x-space describe the geometry of a line gratings
- by applying the operator $F\colon \mathbb{R}^3 \to \mathbb{R}^{23}$ (from a nonlinear PDE and learned by NN) we get the diffraction pattern

$$Y = F(X) + \eta,$$

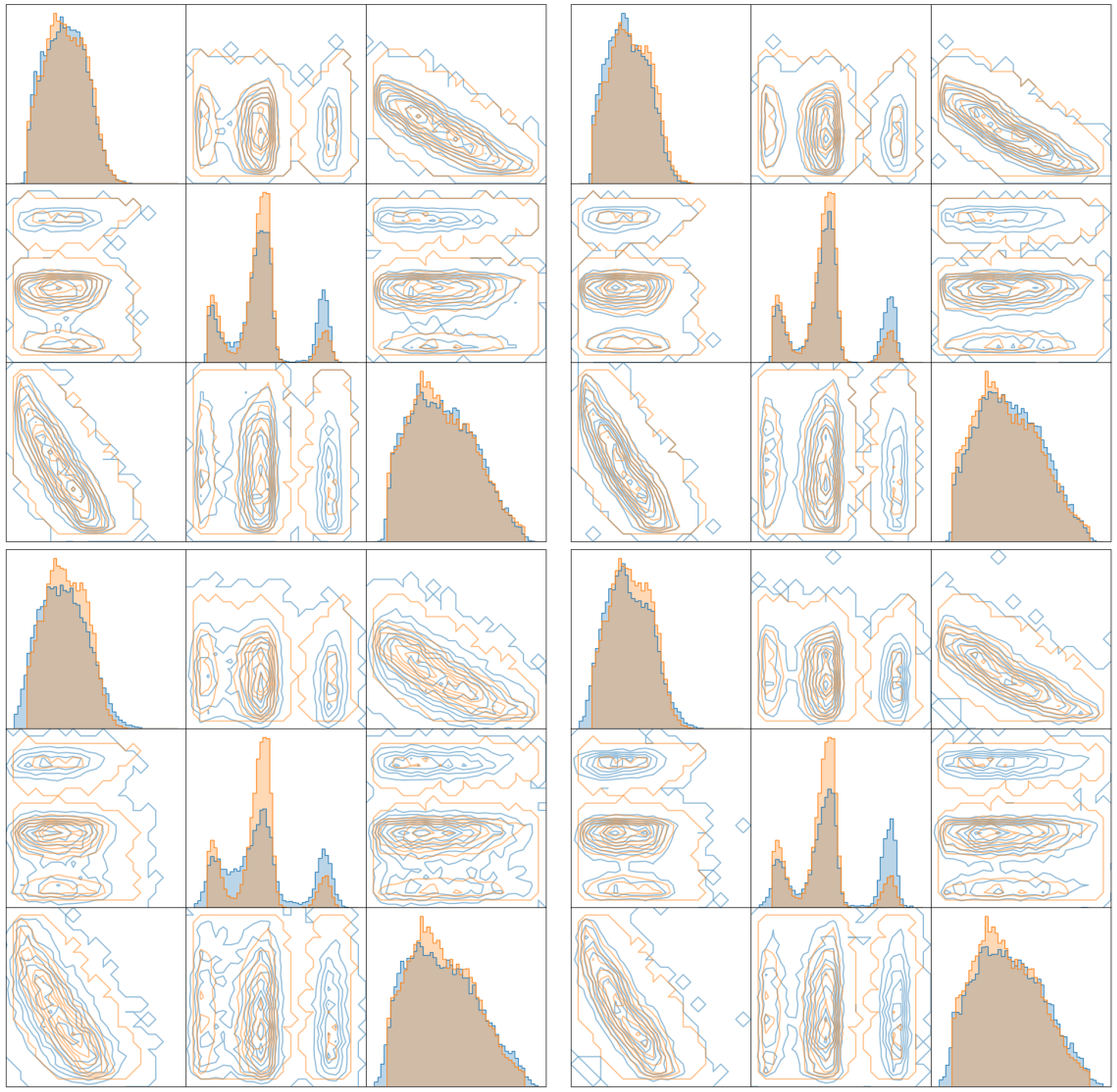
where we have a multiplicative + additive noise model

$$\eta = aF(X)\eta_1 + b\eta_2, \quad \eta_1, \eta_2 \sim \mathcal{N}(0, I), \quad a, b > 0$$



Cooperation with Physikalisch-Technische Bundesanstalt Berlin

2. Example from Scatterometry



Top: INN, INN+MALA, Bottom: VAE, VAE+MALA

Method	INN	INN+MALA	VAE	VAE+MALA
KL	0.76	0.59	0.98	0.69

Conclusions

- Normalizing flows = powerful tool for
  - generative approaches
  - solving inverse problems
  - uncertainty quantification (WPPFlows)
- Use power of patches for unsupervised learning from very few images
- Works for different inverse problems!!

- Normalizing flows with limited expressivity
  multimodal versus unimodal distributions (exploding Lipschitz constant)
- Stochastic normalizing flows (SNF) can help to improve expressivity
- Markov chains incorporate many known concepts (VAEs, Langevin flow, MCMC, ...) in a mathematically sound way

References

1. F. Altekrüger, A. Denker, P. Hagemann, J. Hertrich, P. Maass and G. Steidl,
   PatchNR: Learning from Small Data by Patch Normalizing Flow Regularization,
   ArXiv 2022

2. F. Altekrüger and J. Hertrich,
   WPPNets and WPPFlows: Unsupervised NN training with Wasserstein patch priors,
   SIAM J. Imaging Sciences, submitted 2022

3. P. Hagemann, J. Hertrich and G. Steidl,
   Stochastic normalizing flows for inverse problems: A Markov chain viewpoint,
   SIAM J. Uncertainty Quantification, 2022

4. J. Hertrich and G. Steidl,
   Inertial stochastic PALM and its application for learning Student-t mixture models,
   Sampling Theo, Signal Proc. and Data Anal., 2022

5. J. Hertrich, S. Neumayer and G. Steidl,
   Convolutional proximal neural networks and Plug-and-Play algorithms,
   Linear Algebra and its Applications 2021

6. M. Hasannasab, J. Hertrich, S. Neumayer, G. Plonka, S. Setzer, G. Steidl,
   Parseval proximal neural networks,
   Journal of Fourier Analysis and Applications 2021

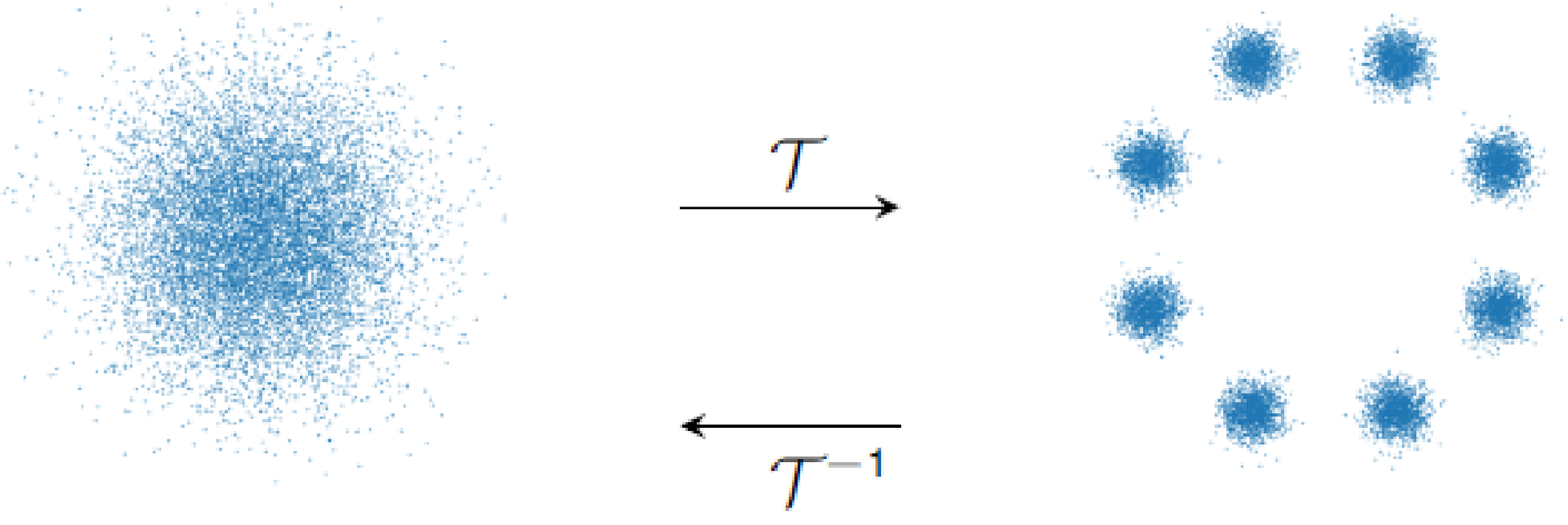
- Programs are available at Gitup

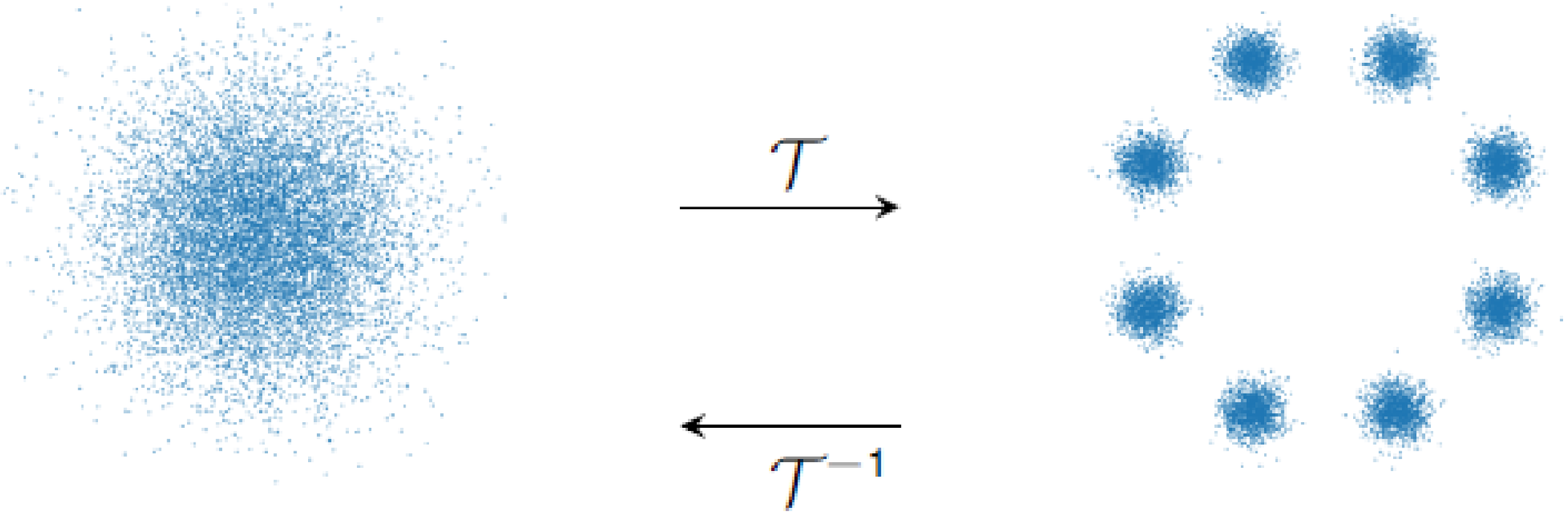
Technische Universität Berlin

Many thanks for your attention!

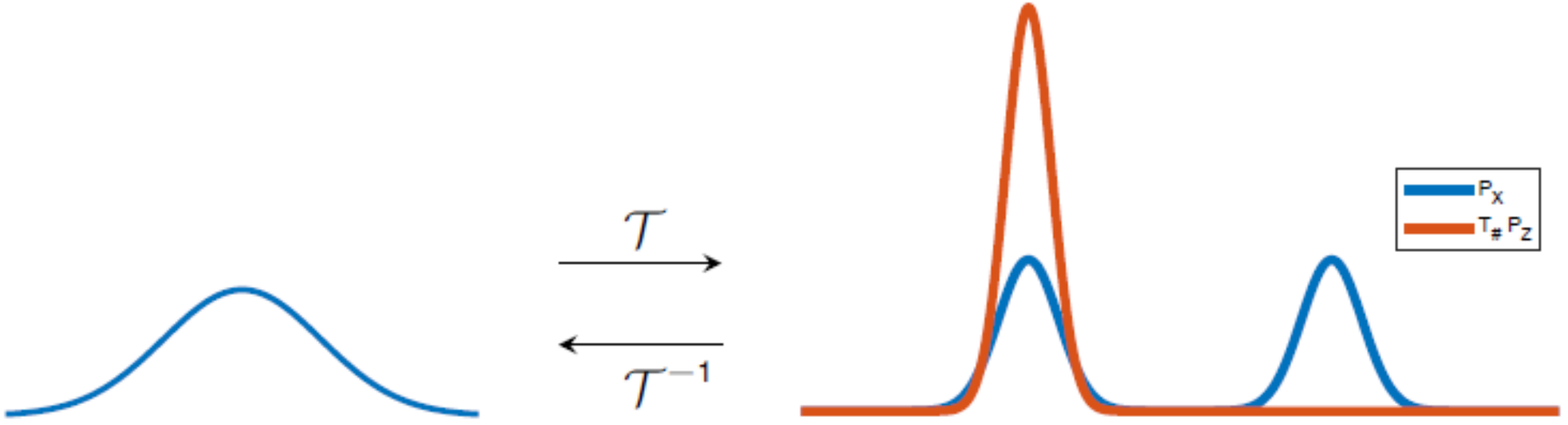
- Mathematics and Image Analysis (MIA) 2023
  1-3 February 2023, Berlin, Germany
  https://www.wias-berlin.de/workshops/MIA2023/

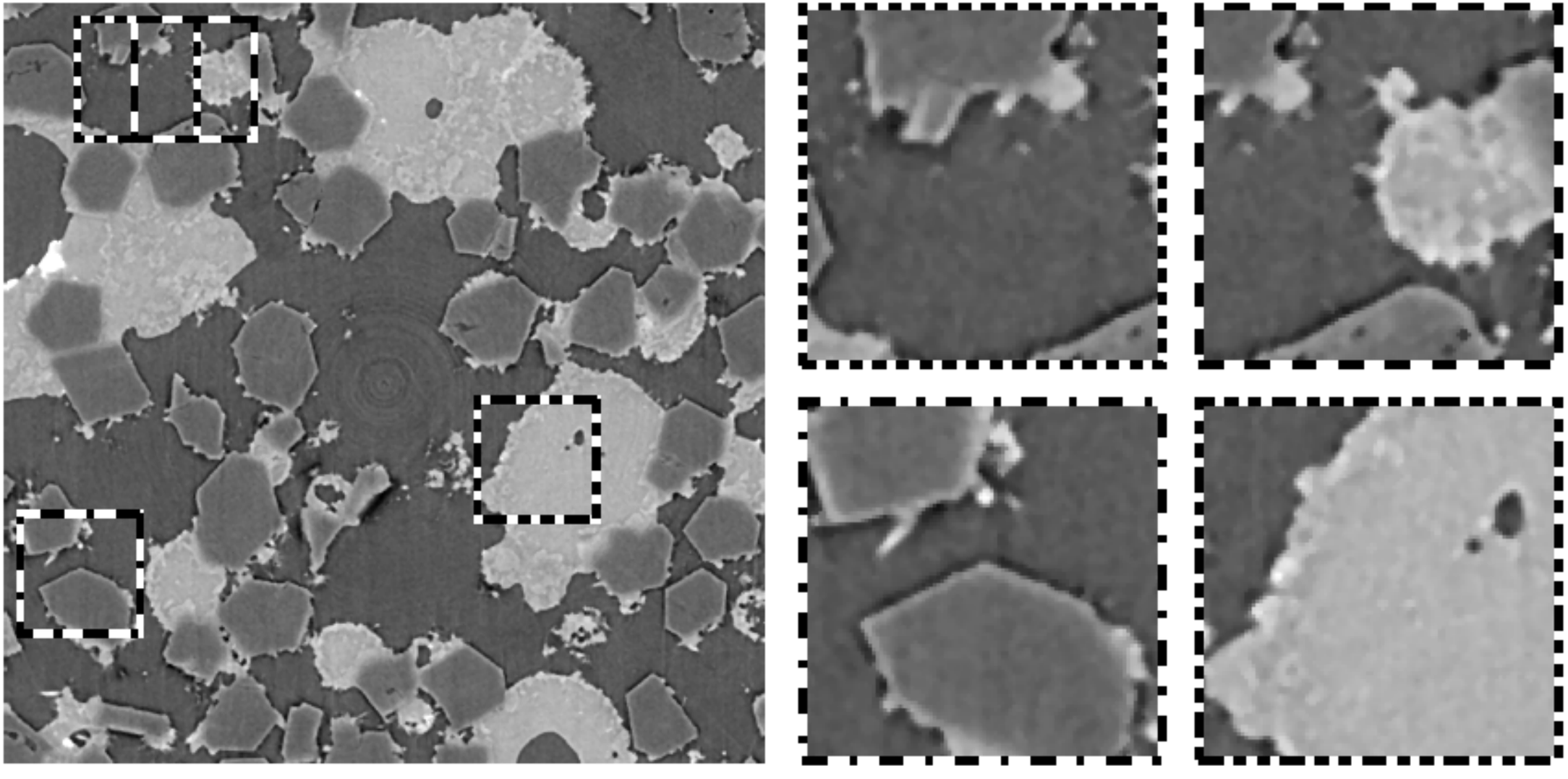
- Advertisement of a Postdoc position (4 years) TU Berlin and a PhD position (DFG-RTG: TU Berlin /Charité, 3 years)


T
T^{-1}
Latent distribution P_Z
Data distribution P_X


T
T^{-1}
Latent distribution P_Z
Data distribution P_X


T
T^{-1}
P_X
$T_\# P_Z$

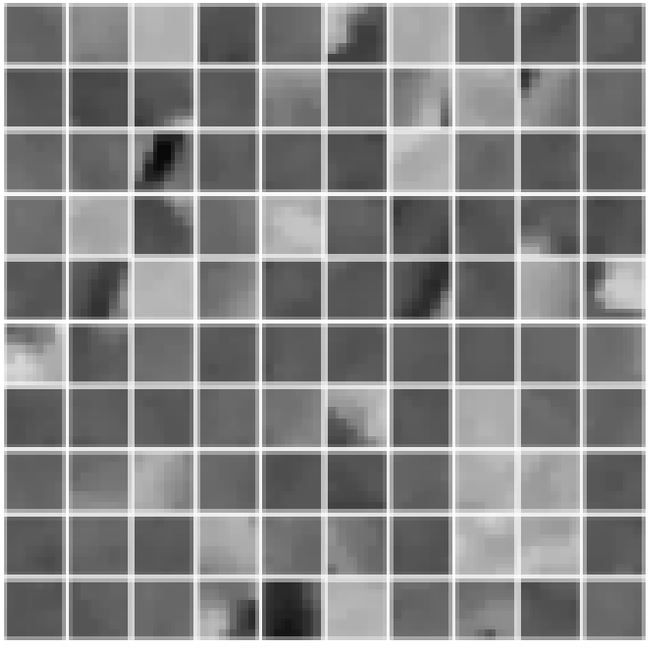

T
T^{-1}
P_X
$T_\# P_Z$



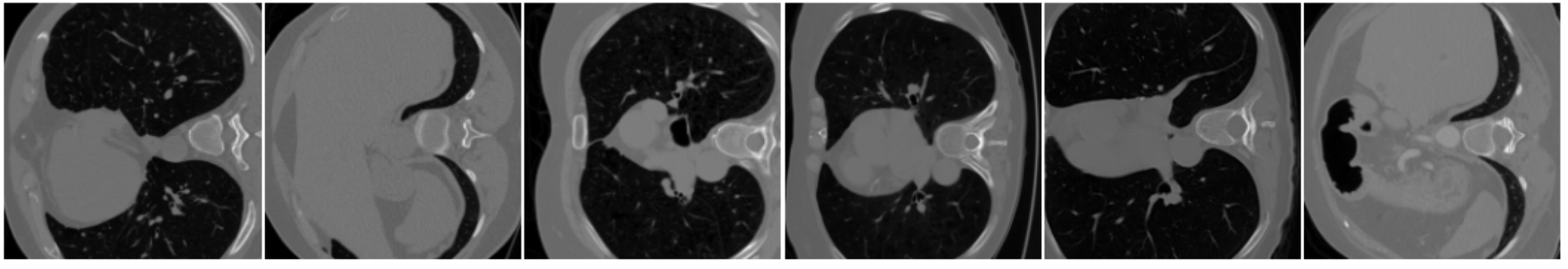
Latent distribution P_Z
on $\mathbb{R}^{36}$

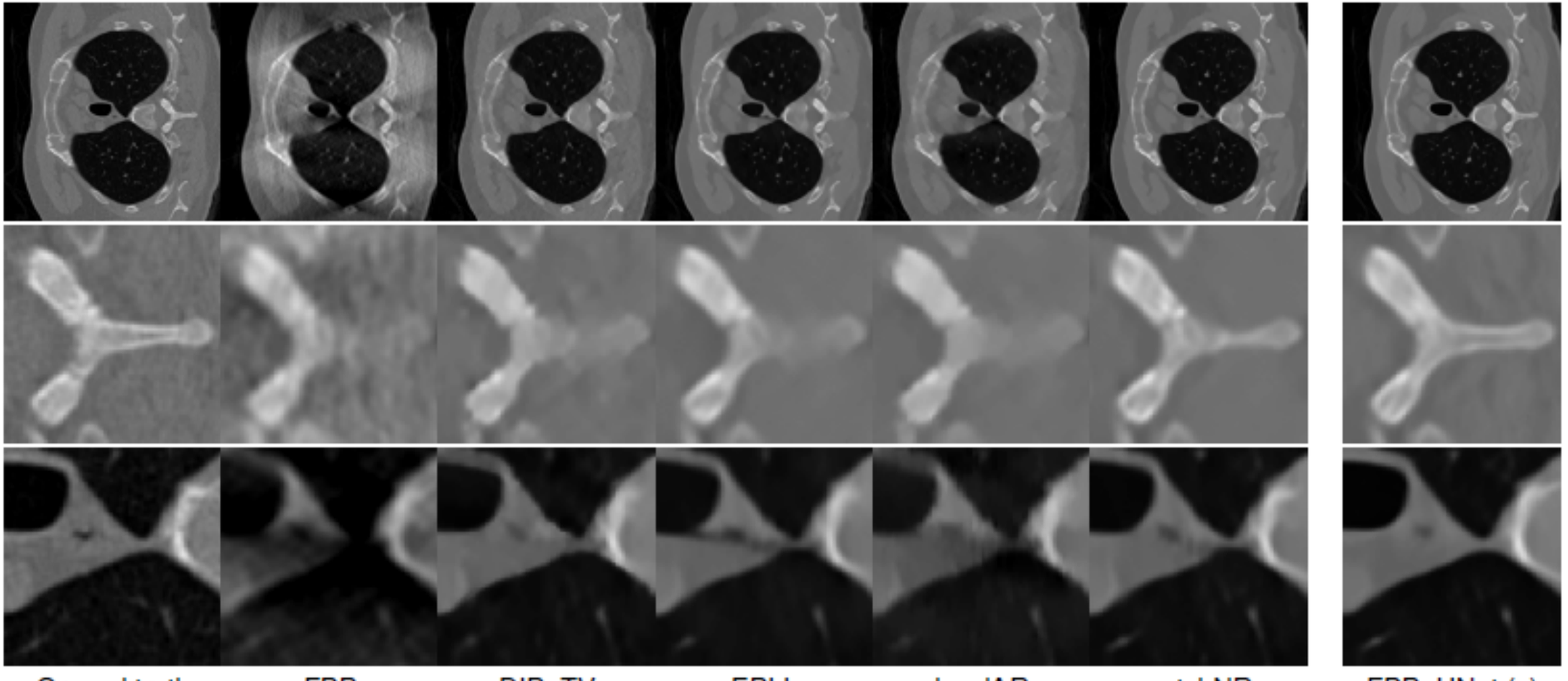
T

T^{-1}

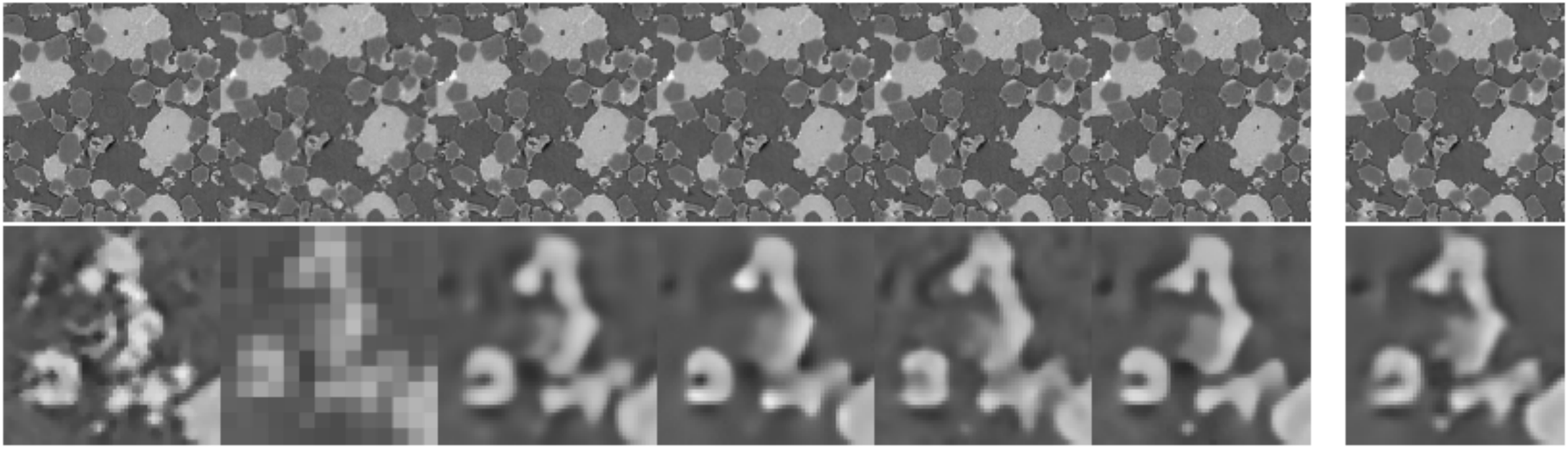


Patch distribution P_X
on $\mathbb{R}^{36}$

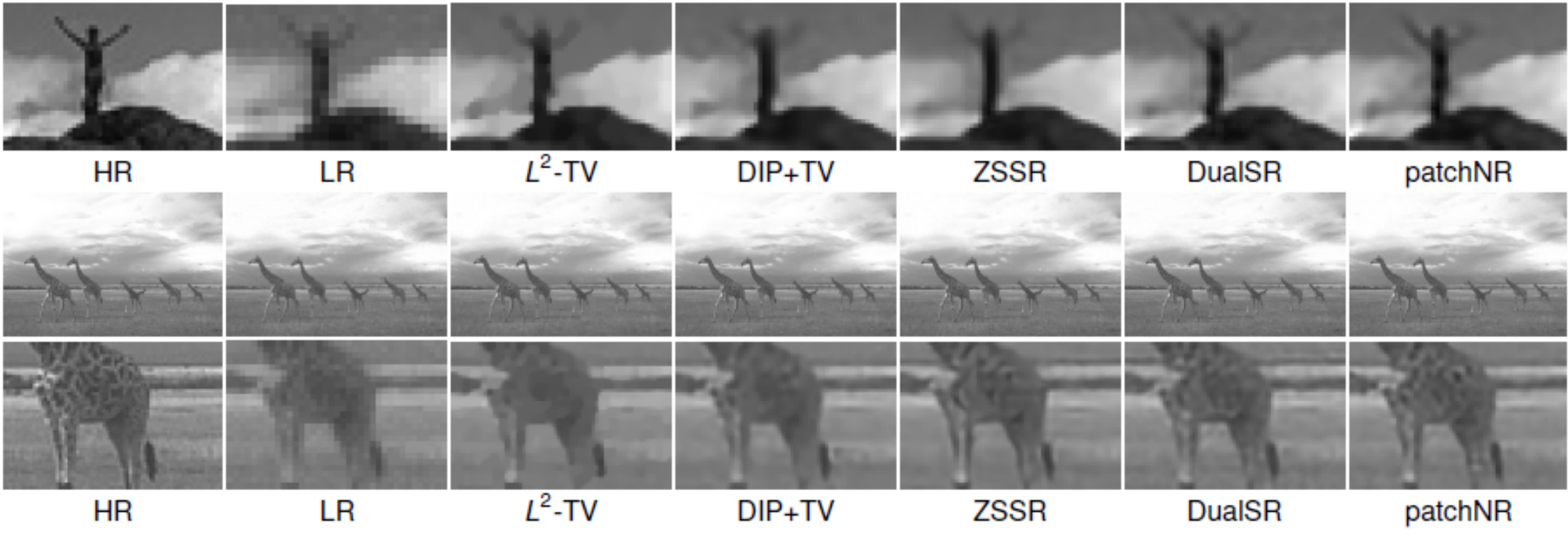



Ground truth
FBP
DIP+TV
EPLL
localAR
patchNR
FBP+UNet (s)

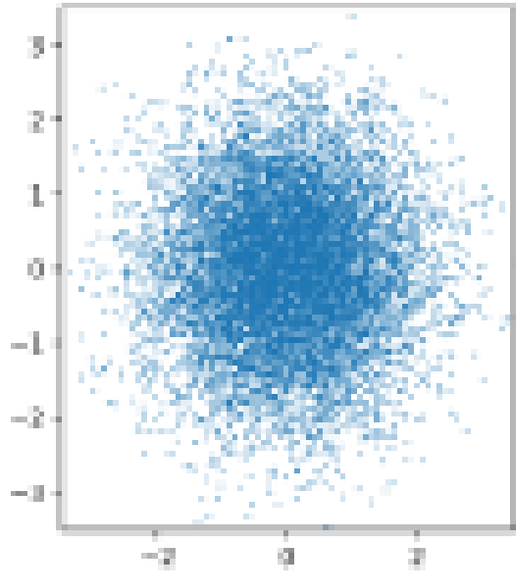
	FBP	DIP + TV	EPLL	localAR	patchNR	FBP+UNet (s)
PSNR	21.96 ± 2.25	32.57 ± 3.25	32.78 ± 3.46	31.06 ± 2.95	**33.20** ± 3.55	33.75 ± 3.58
LPIPS	0.305 ± 0.117	**0.191** ± 0.165	0.216 ± 0.175	0.222 ± 0.166	0.201 ± 0.176	0.171 ± 0.134
SSIM	0.531 ± 0.097	0.803 ± 0.146	0.801 ± 0.151	0.779 ± 0.142	**0.811** ± 0.151	0.820 ± 0.140

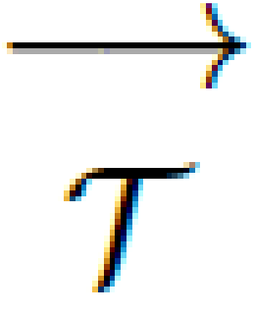

HR
LR
DIP + TV
EPLL
WPP
patchNR
ACNN
(supervised)

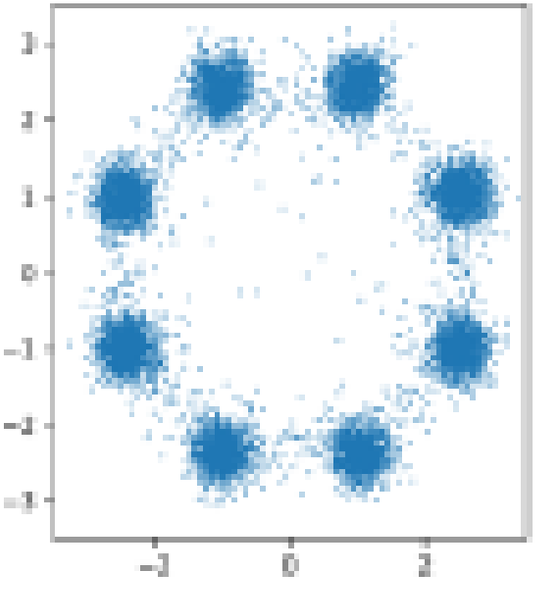
	bicubic (not shown)	DPIR (not shown)	DIP+TV	EPLL	WPP	patchNR	ACNN (supervised)
PSNR	25.63 $\pm$ 0.56	27.78 $\pm$ 0.53	27.99 $\pm$ 0.54	28.11 $\pm$ 0.55	27.80 $\pm$ 0.37	**28.53** $\pm$ 0.49	28.89 $\pm$ 0.53
LPIPS	0.406 $\pm$ 0.013	0.322 $\pm$ 0.015	0.191 $\pm$ 0.009	0.244 $\pm$ 0.012	0.167 $\pm$ 0.014	**0.159** $\pm$ 0.008	0.203 $\pm$ 0.011
SSIM	0.699 $\pm$ 0.012	0.770 $\pm$ 0.011	0.764 $\pm$ 0.007	0.779 $\pm$ 0.010	0.749 $\pm$ 0.011	**0.780** $\pm$ 0.008	0.804 $\pm$ 0.010

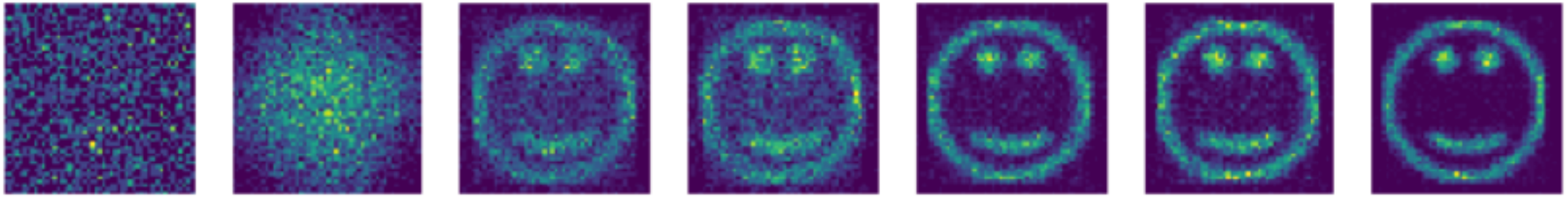

HR
LR
L^2-TV
DIP+TV
ZSSR
DualSR
patchNR
HR
LR
L^2-TV
DIP+TV
ZSSR
DualSR
patchNR

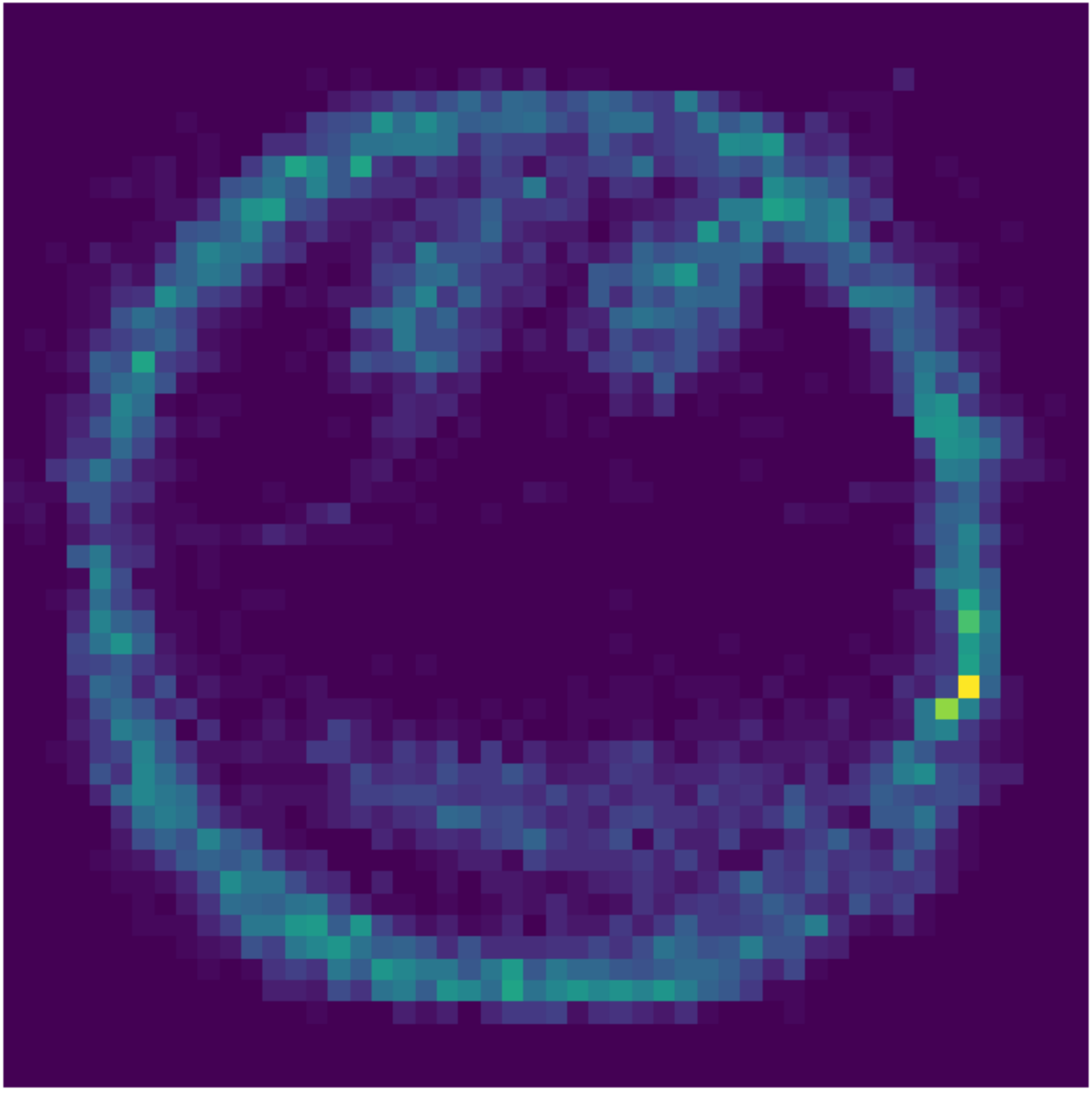
	L^2-TV	DIP+TV	ZSSR	DualSR	patchNR
PSNR	28.35 $\pm$ 3.55	28.44 $\pm$ 3.69	28.83 $\pm$ 3.57	28.64 $\pm$ 3.47	**29.08** $\pm$ 3.58
LPIPS	**0.184** $\pm$ 0.073	0.215 $\pm$ 0.086	0.224 $\pm$ 0.085	0.216 $\pm$ 0.074	0.202 $\pm$ 0.076
SSIM	0.820 $\pm$ 0.072	0.821 $\pm$ 0.087	0.834 $\pm$ 0.066	0.829 $\pm$ 0.061	**0.846** $\pm$ 0.061

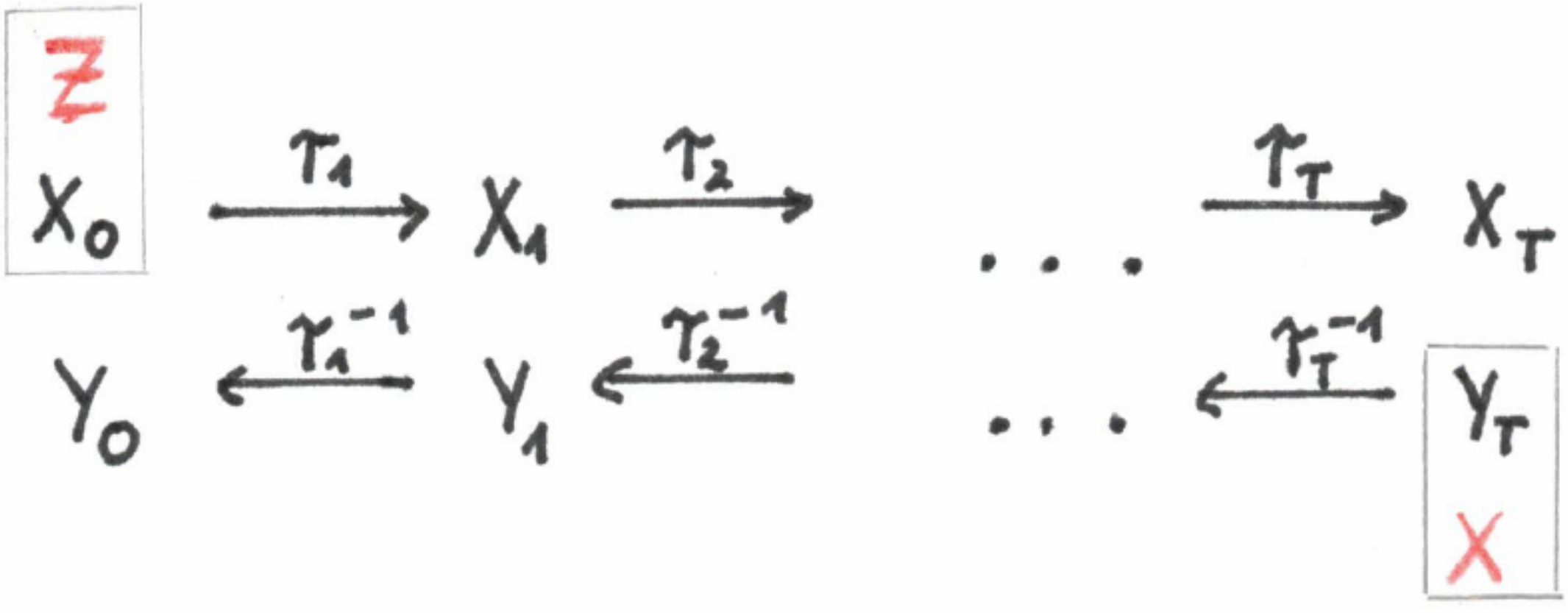

3
2
1
0
−1
−2
−3
−2
0
2

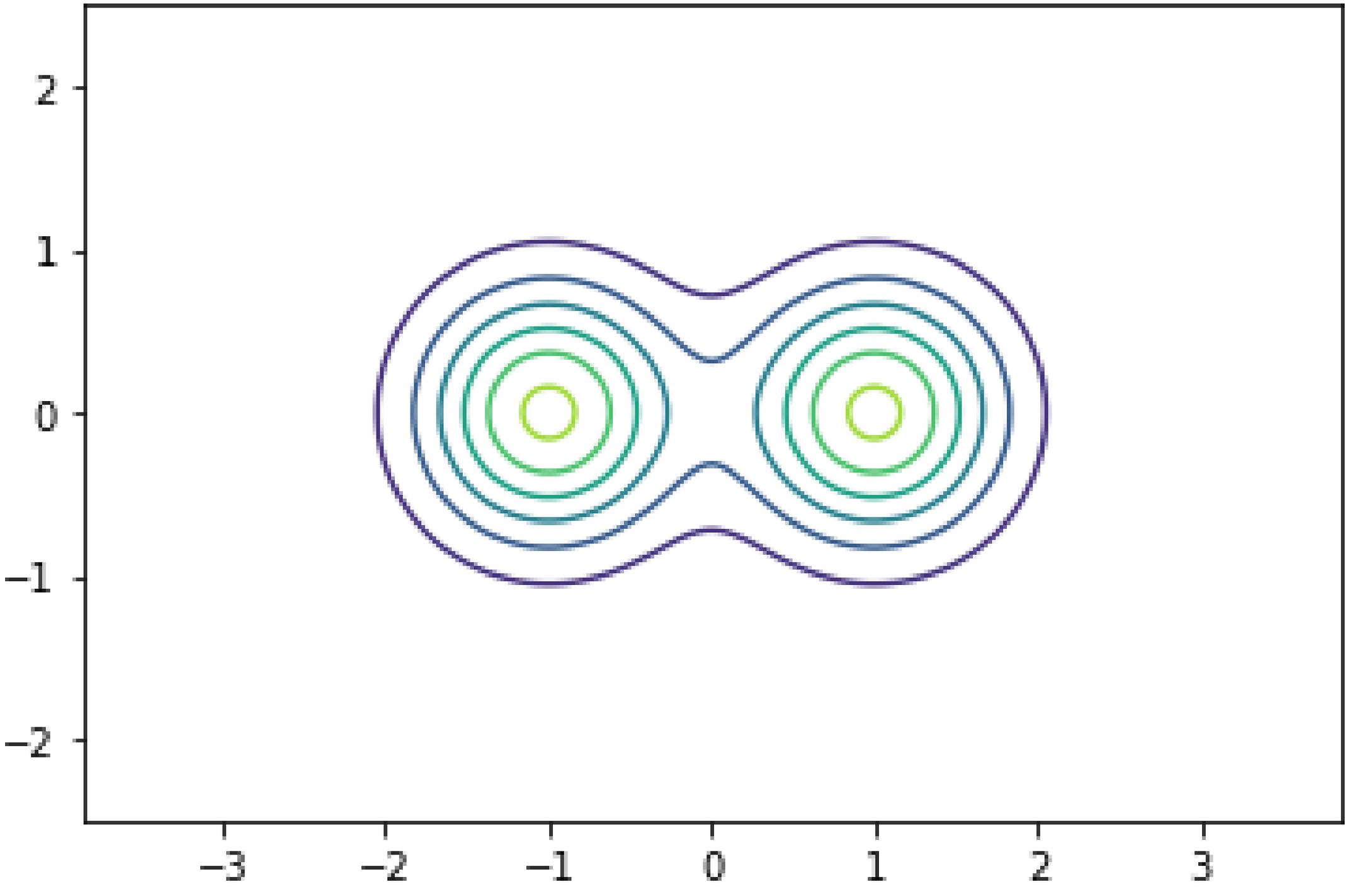

T

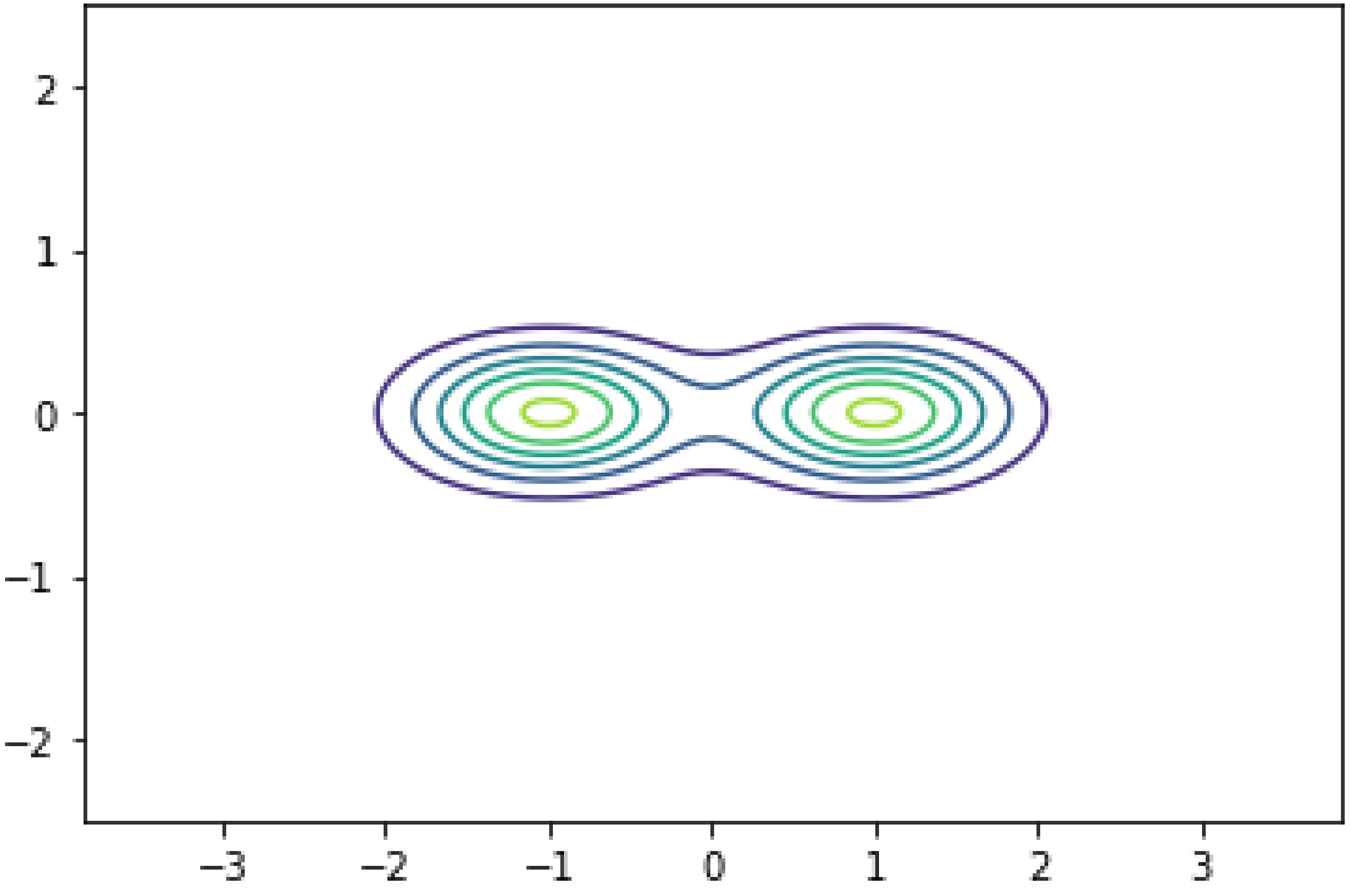

3
2
1
0
−1
−2
−3
−2
0
2

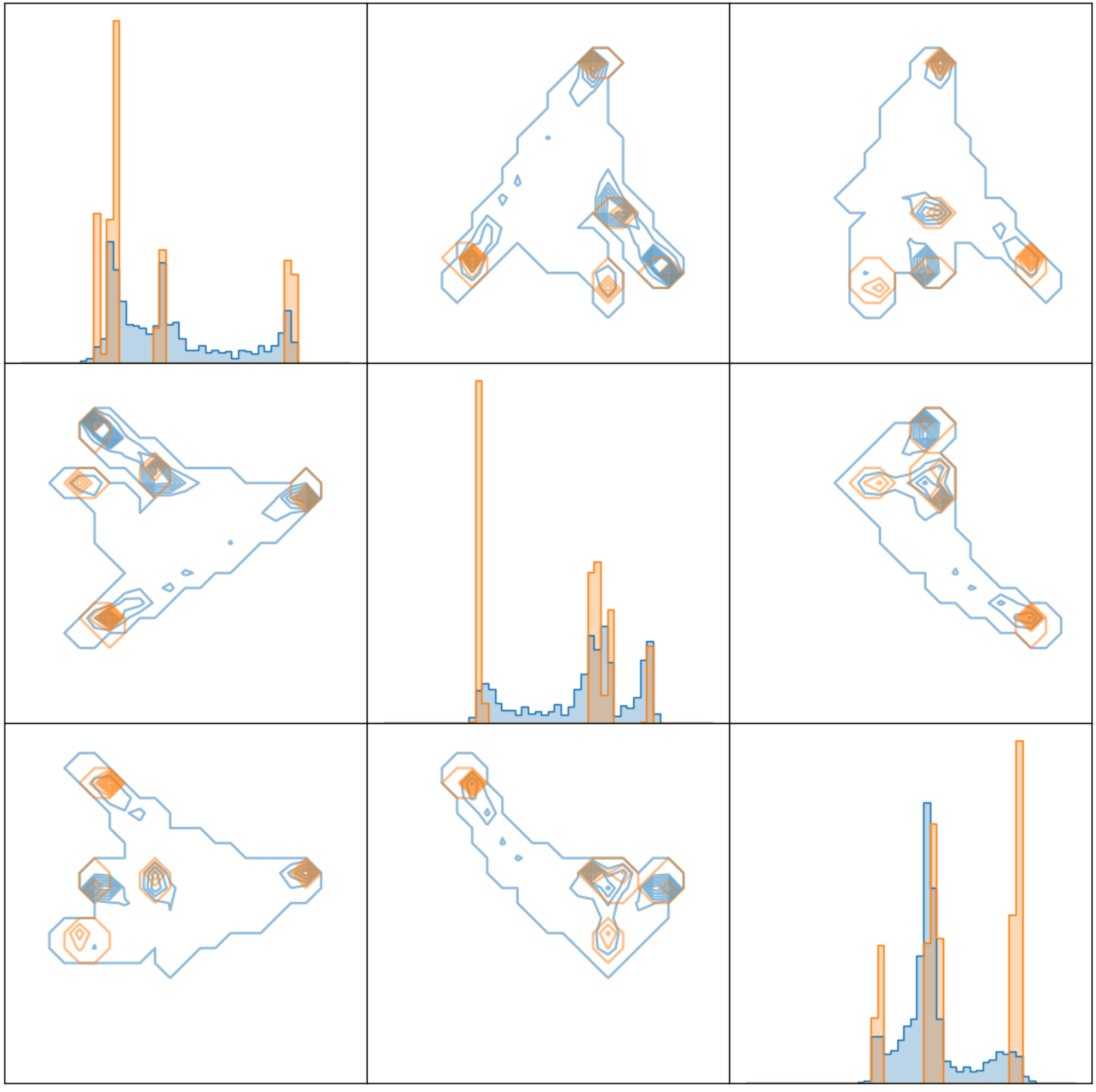


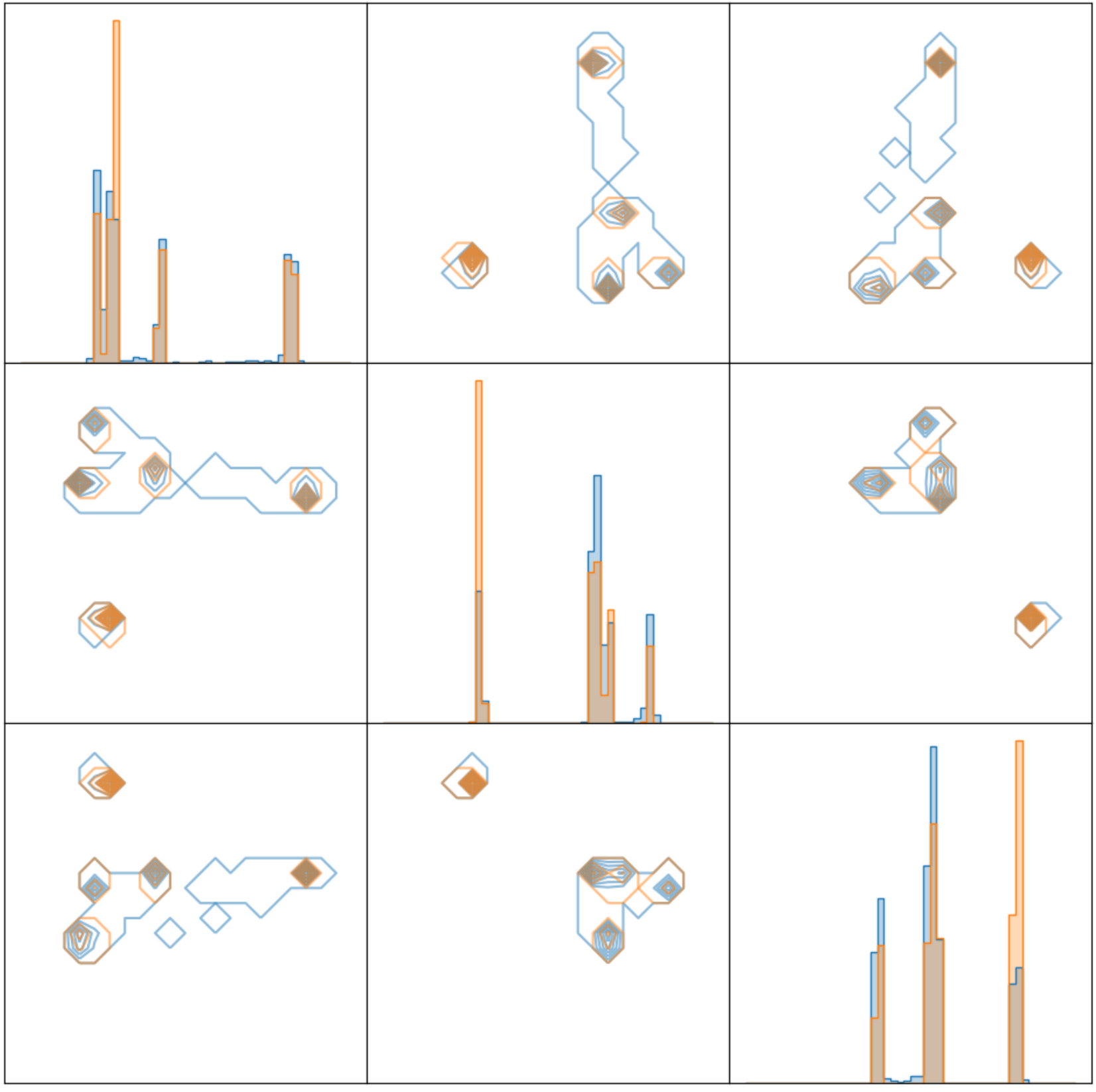


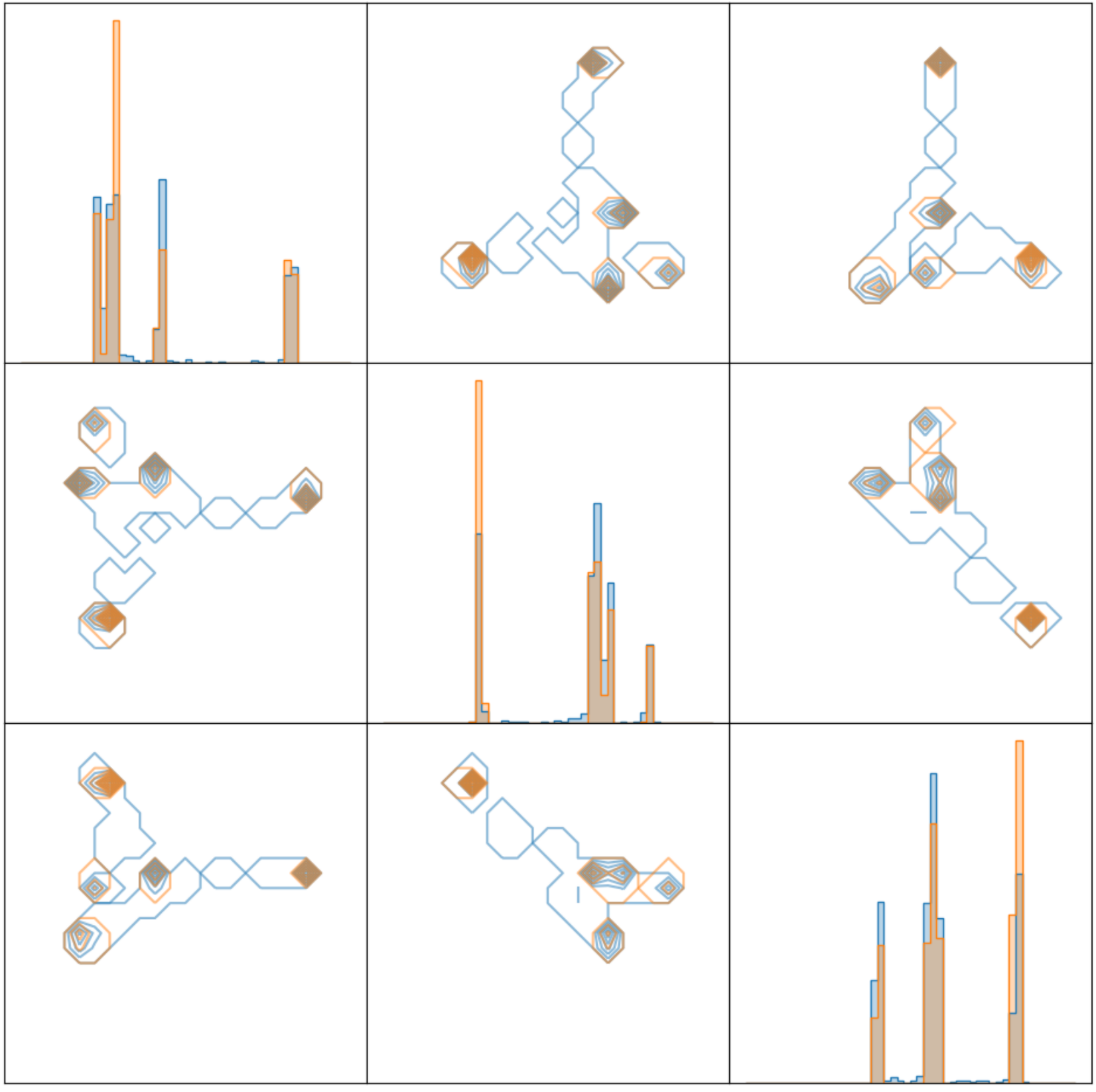

Z
X_0
τ_1
X_1
τ_2
. . .
τ_T
X_T
Y_0
τ_1^{-1}
Y_1
τ_2^{-1}
. . .
τ_T^{-1}
Y_T
X

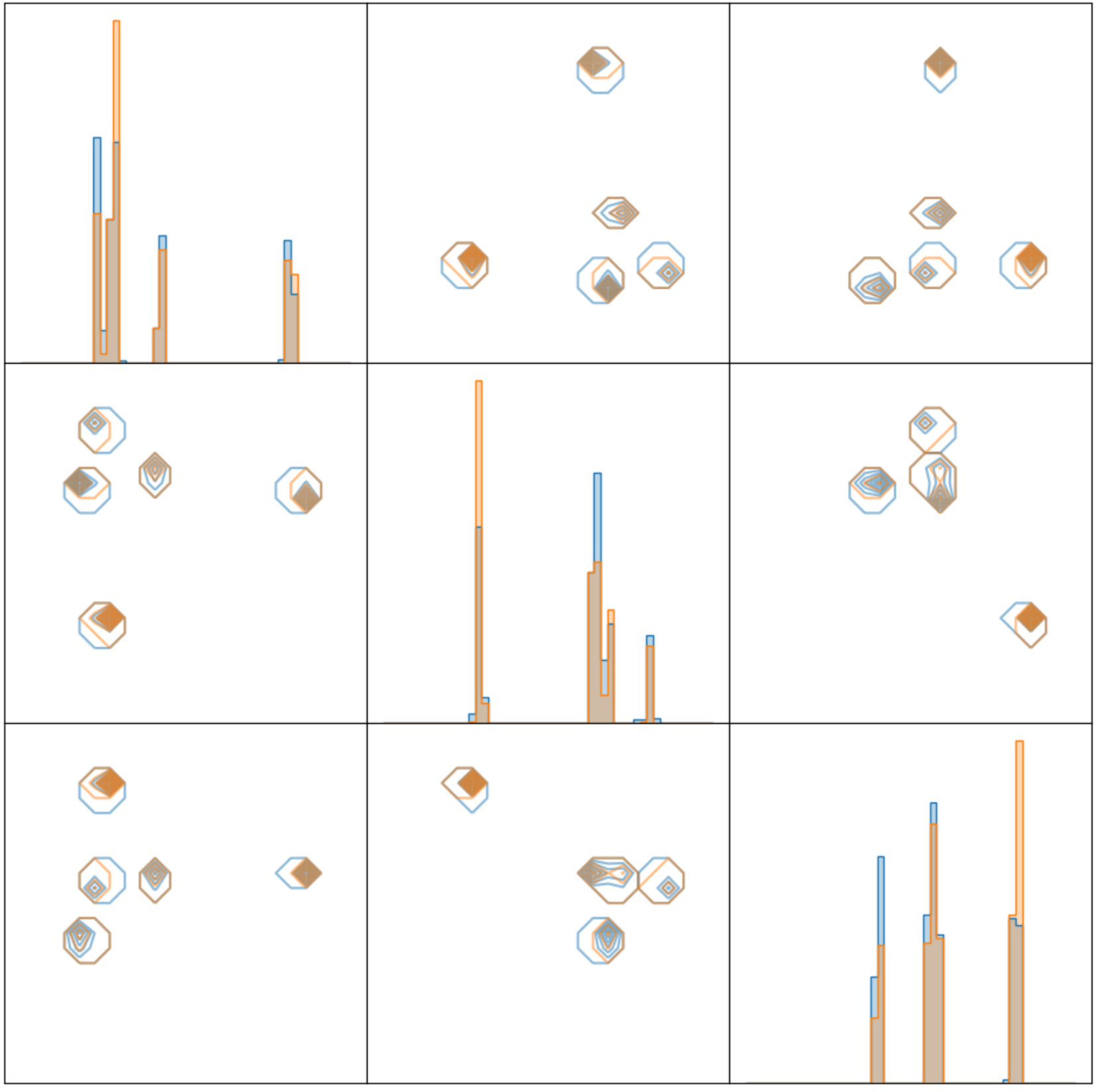

2
1
0
−1
−2
−3
−2
−1
0
1
2
3

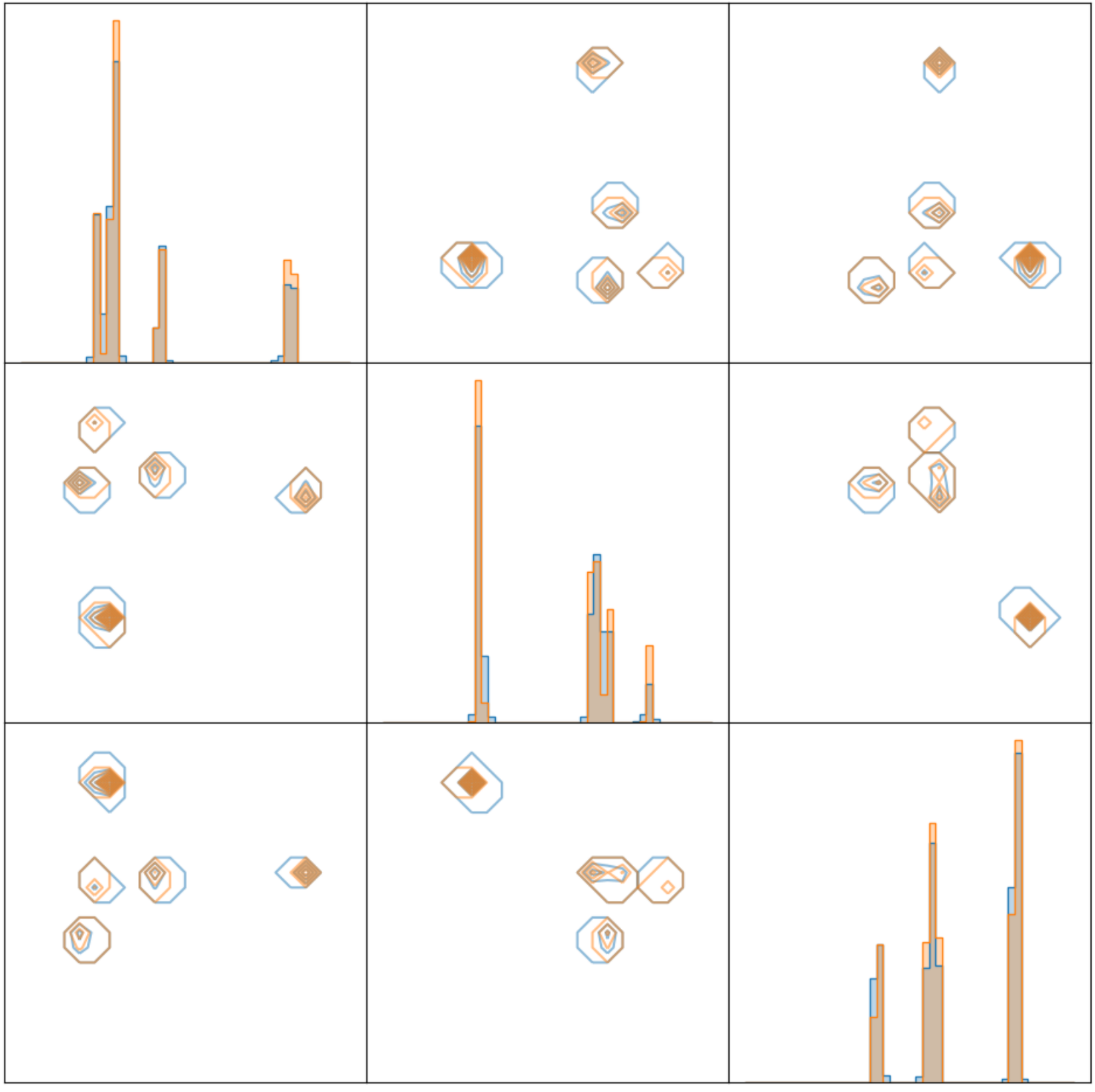

2
1
0
−1
−2
−3
−2
−1
0
1
2
3

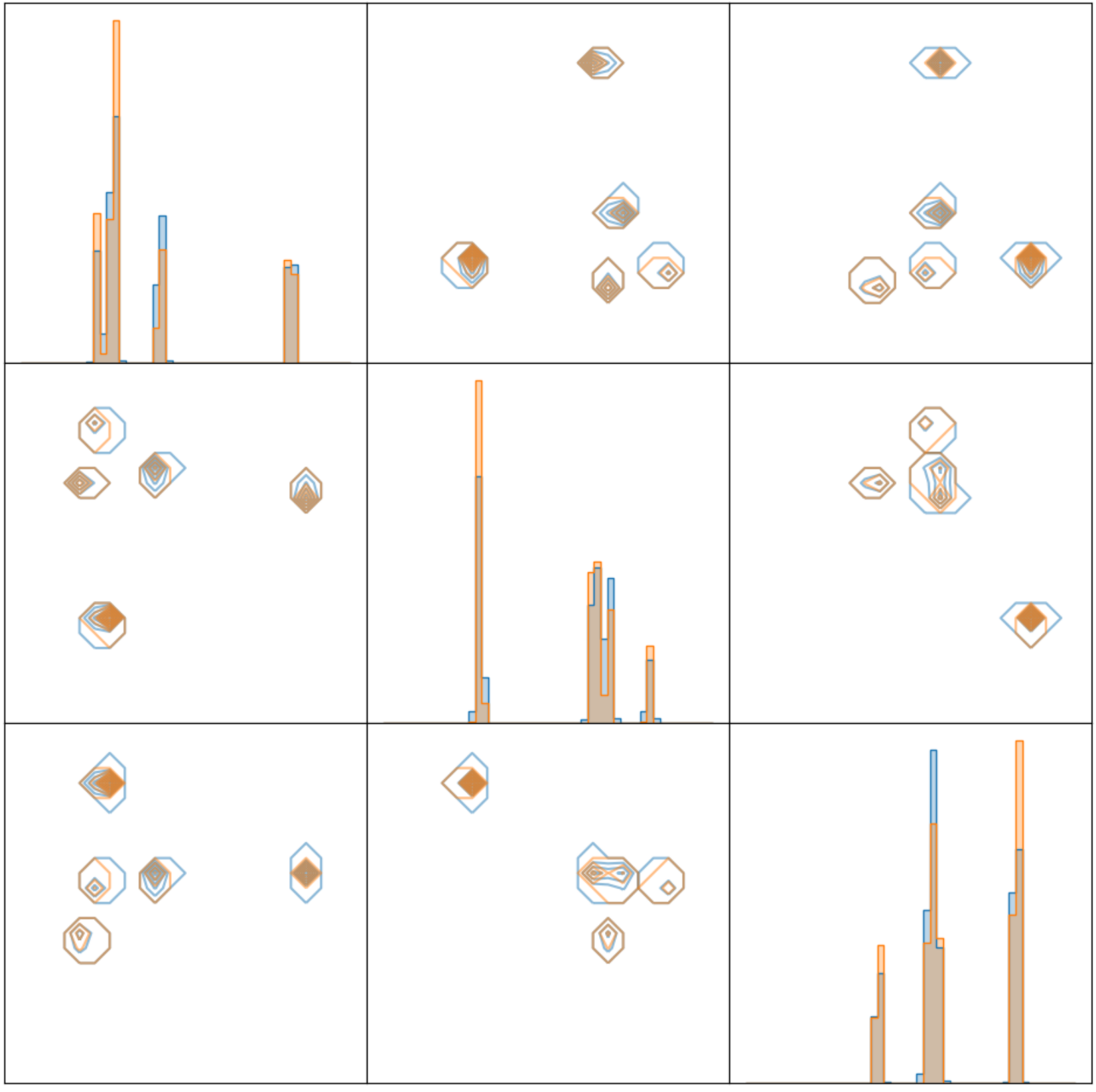


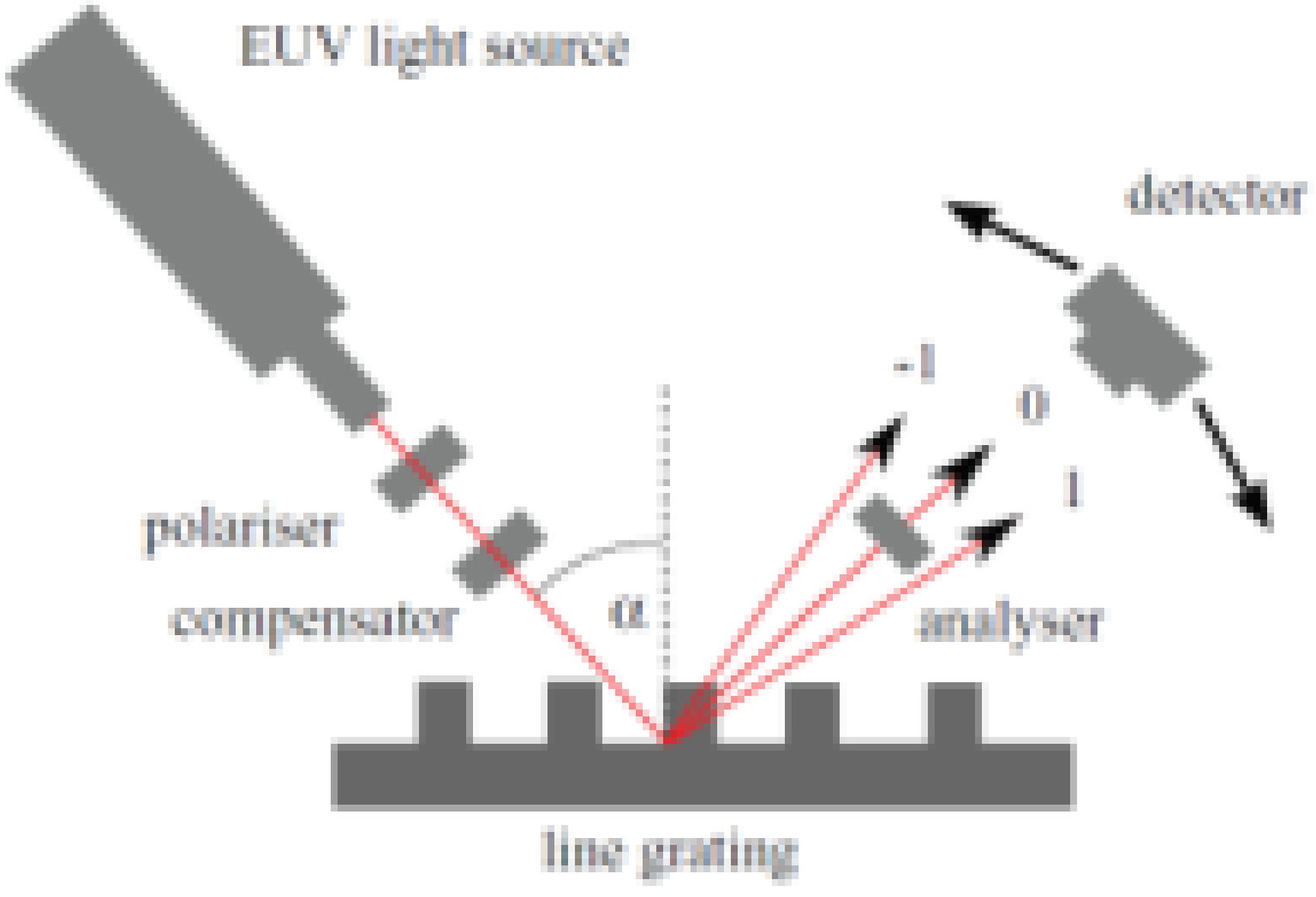


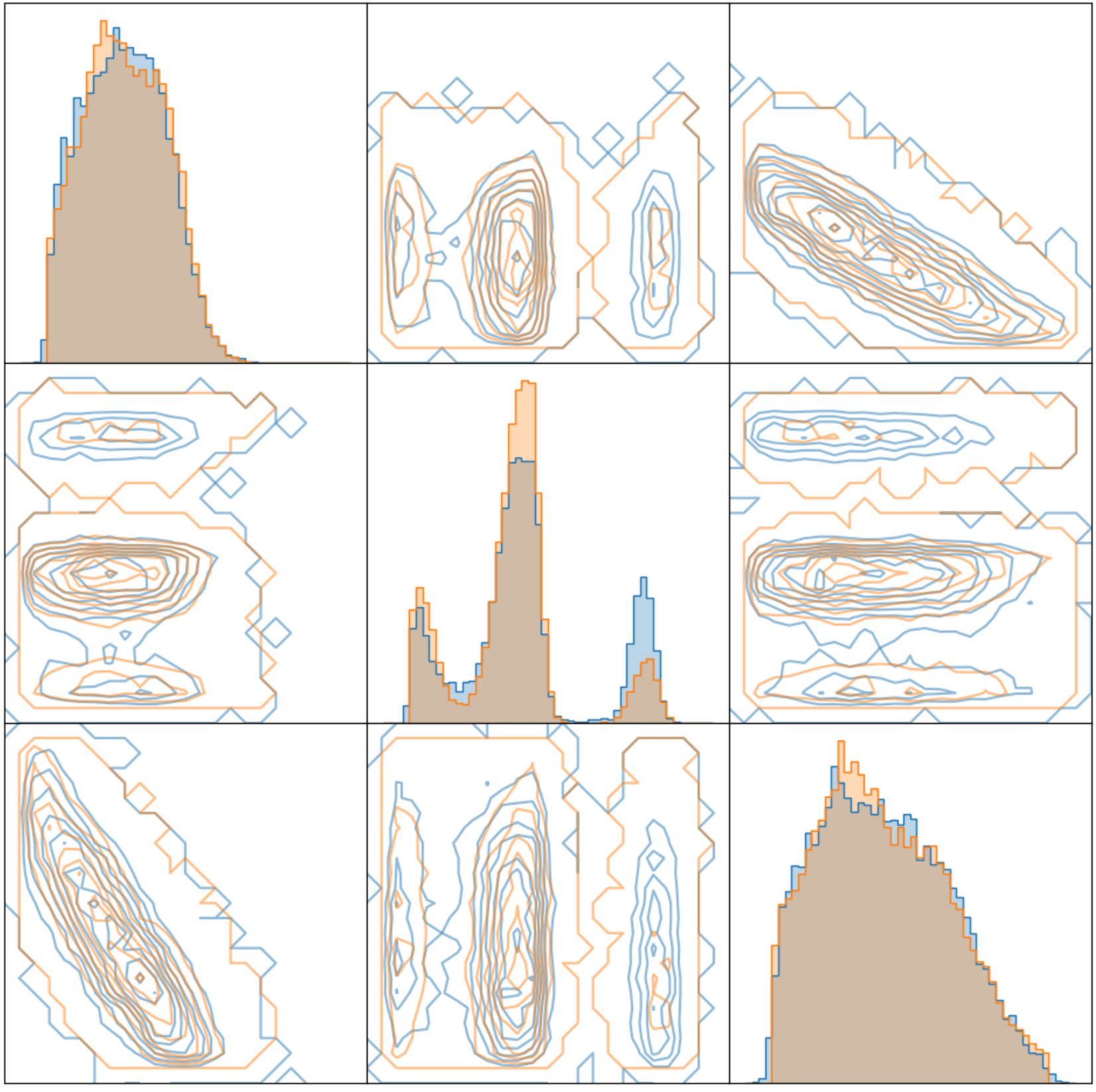


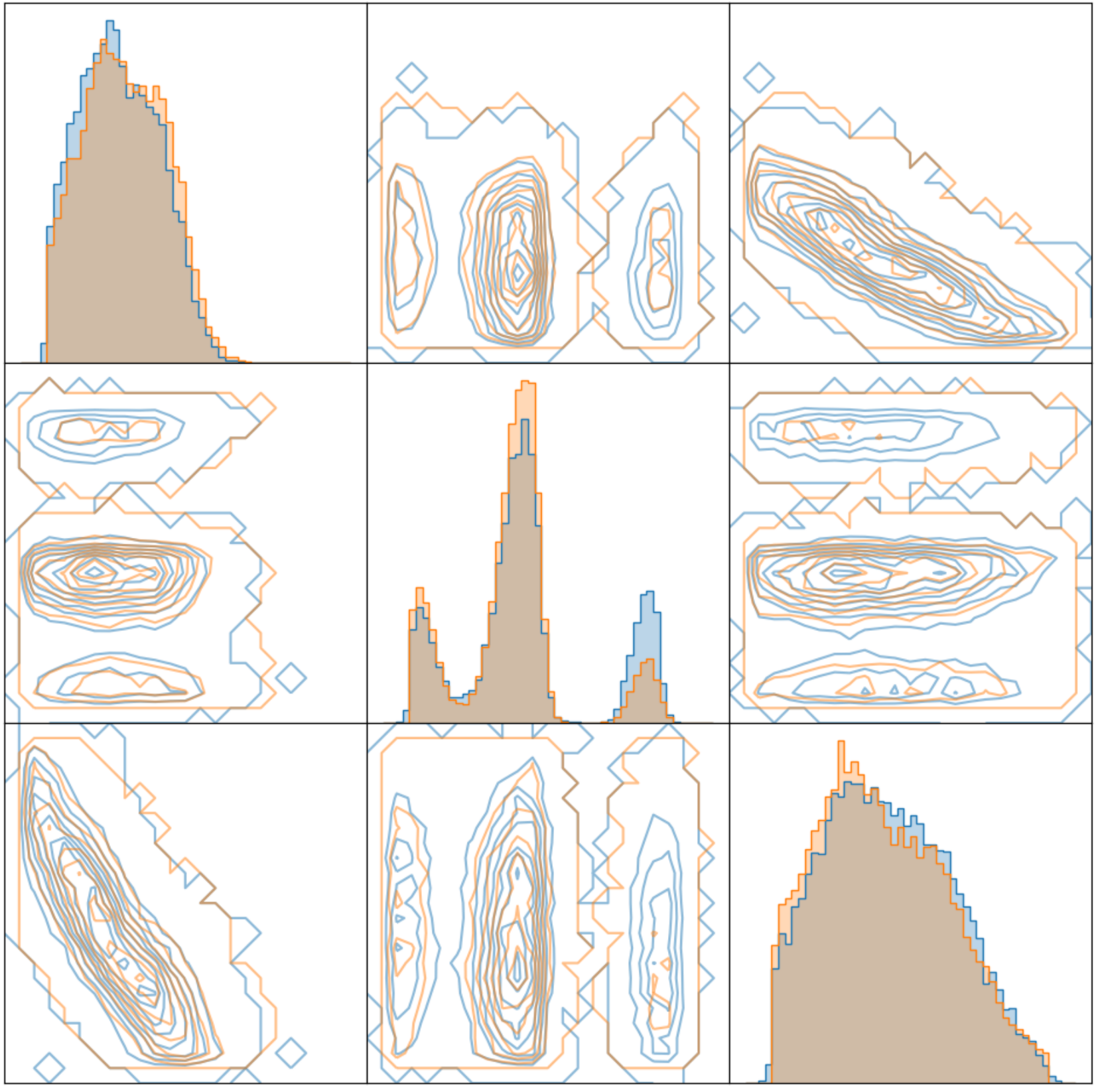


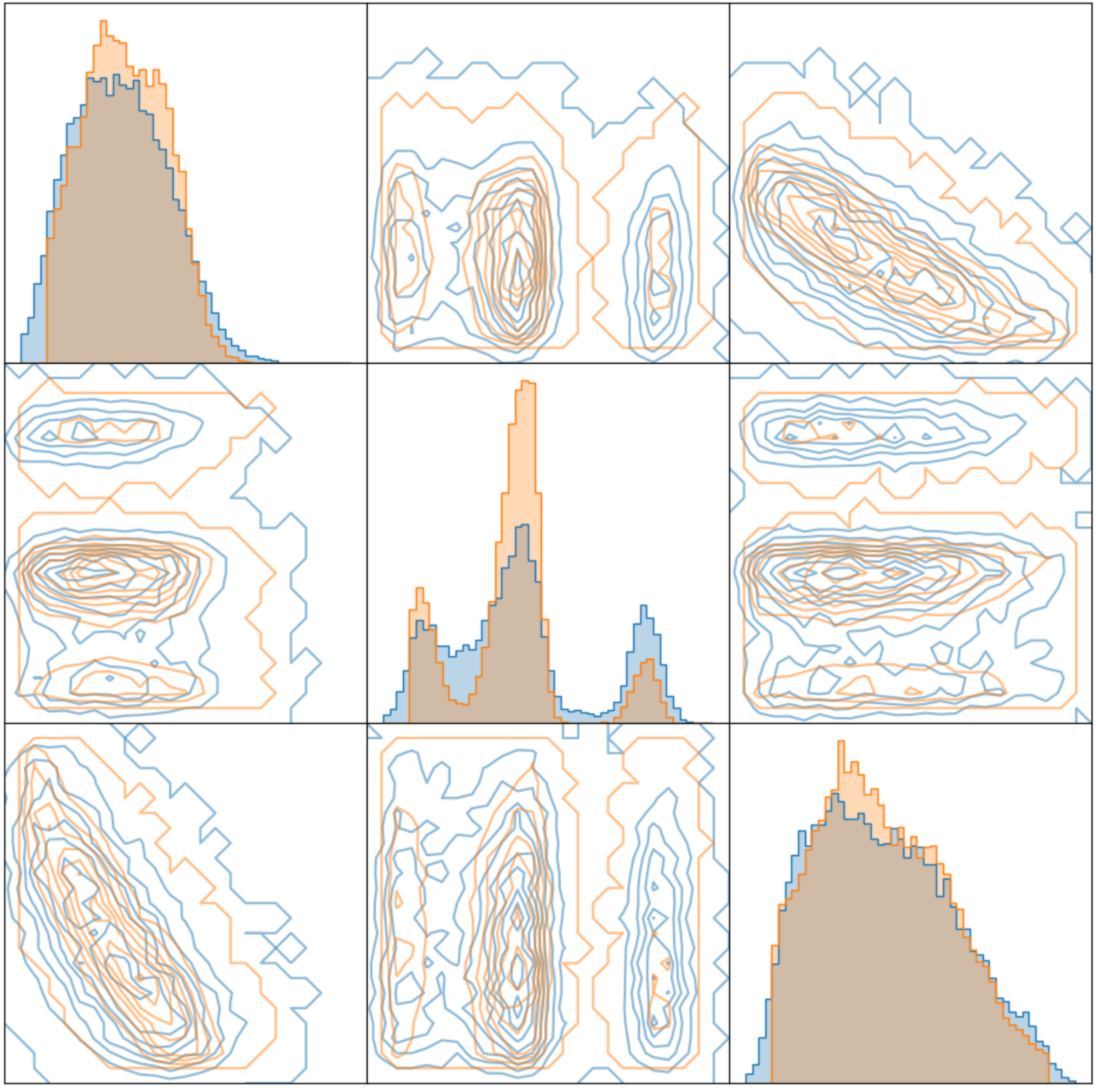


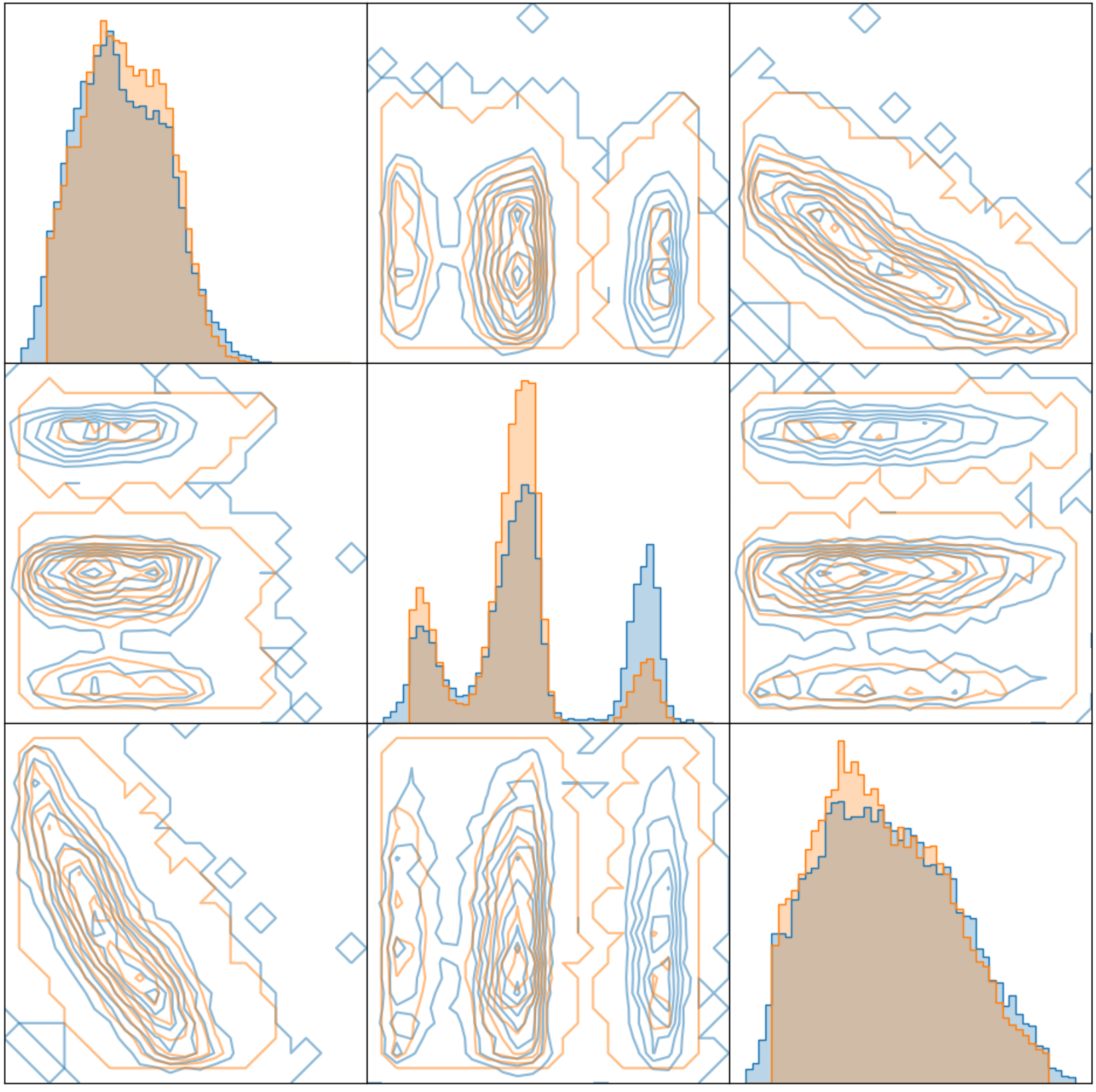



EUV light source
detector
-1
0
1
polariser
compensator
α
analyser
line grating





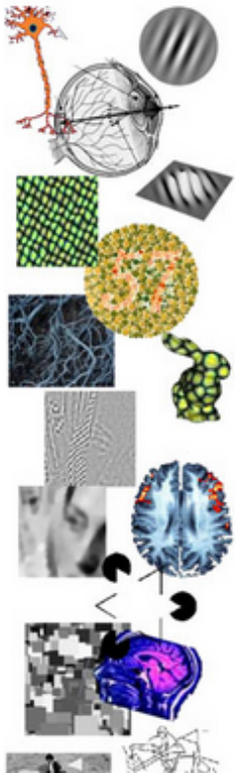




MIA 2023

Mathematics and Image Analysis

1-3 February 2023, Berlin, Germany



Aims and Scope

This is the next edition of the Mathematics and Image Analysis conference that will be held in Berlin in February 2023. It is organised by RT MIA from the French side and by Humboldt University Berlin, Technical University Berlin and Weierstrass Institute (WIAS) from the German side.

The conference follows a series of very successful, established MIA conferences. The first one took place in 2000 and it was subsequently held every two years at the Institute Henri Poincaré in Paris. Since 2014, German scientists have been involved in the organization of the conference and it was decided to organize it alternatingly in Paris and Berlin.

The conference will address a wide range of topics:

- Mathematics of novel imaging methods
- Inverse problems in imaging
- Mathematics of visualization
- Motion analysis
- Video processing
- Statistical and data science aspects in image processing
- PDEs and variational methods in image processing
- Deep and other machine learning methods in imaging

Previous MIA conferences:

MIA 2021 MIA 2018 MIA 2016 MIA 2014 MIA 2012 MIA 2009 MIA 2006 MIA 2004 MIA 2002 MIA 2000

Invited Speakers

- Michael Arbel (INRIA Grenoble)
- Mathieu Aubry (LIGM-Imagine, ENPC, Paris)
- Tatiana Bubba (University of Bath)
- Martin Burger (University of Erlangen-Nuremberg)
- Laetitia Chapel (Université de Bretagne Sud)
- Tom Goldstein (University of Maryland)
- Gloria Haro (University Pompeu Fabra)
- Ulugbek Kamilov (Washington University in St. Louis)
- Florian Knoll (University of Erlangen-Nuremberg)
- Zorah Lähner (University of Siegen)
- Serena Morigi (University of Bologna)
- Nelly Pustelnik (CNRS Lyon)
- Audrey Repetti (Heriot Watt University)
- Otmar Scherzer (University of Vienna)
- Julia Schnabel (TU Munich)
- Vladimir Spokoiny (Humboldt University and WIAS Berlin)
- Tomer Michaeli (Technion)
- Pierre Weiss (University of Toulouse)