Normalizing flows (cont'd)

- NFs as a compression (coding) method

	high compression $\Rightarrow$ small codes	good reconstruction $\hat{x} = g(f(x)) \approx x$	accurate distribution $\hat{p}(x) \sim p^*(x)$
AE	$\dim(z)$ hyperparameter	☺ (minimize loss)	☹ ($p_E(z) = f_\# p^*(x)$ unknown)
GAN	— " —	☺ (no encoder)	☺ (minimize loss)
AE with latent distrib. (*)	— " —	☺ ← trade-off →	☺
NF	☺ $\dim(z) = \dim(x)$	☺ $\hat{x} = g(f(x)) = x$ (invertible networks)	☺ (minimize loss)

3 goals of coding

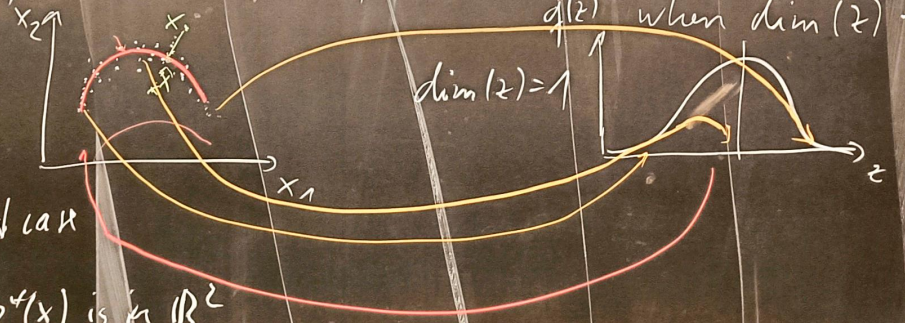
$z = f(x)$ $\quad \hat{x} = g(z)$

$z \sim p(z|x)$ $\quad \hat{x} \sim p(x|z)$

(*) AE with controlled latent distribution:
- VAE, - InfoVAE
- invertible GAN

- in AE: even if $q(z) = p_E(z)$, $\hat{p}(x) = p^*(x)$ is generally impossible

example: banana distribution
$\dim(x) = 2$

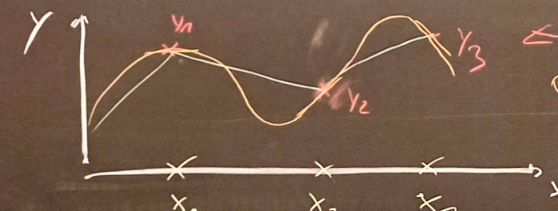


- imperfect reconstruction
$|\hat{x} - x| \neq$ noise in the best case
- imperfect generation
$\hat{p}(x)$ is on centerline, $p^*(x)$ is in $\mathbb{R}^2$

- change in VAE: $\mathbb{E}_{\hat{p}(\hat{x}|z=f(x))}[\hat{x}|x]$ is the centerline
$\hat{p}(\hat{x}|z=f(x))$ creates synthetic noise indistinguishable from true noise

Variants of NF architectures

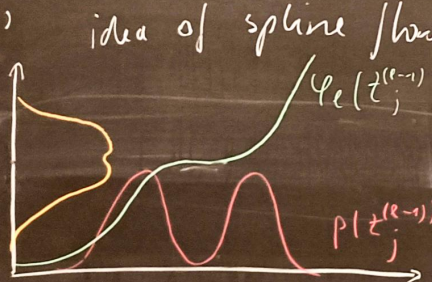
- coupling flows
  - volume-preserving (NICE, GIN) are not universal (cannot change the number of modes $\Rightarrow$ define $q(z) = \sum_k \pi_k \mathcal{N}(\mu_k, \Sigma_k)$ GMM
  - affine coupling $z^{(\ell)}_{>D} = s_\ell(z^{(\ell-1)}_{\leq D}) \cdot z^{(\ell-1)}_{>D} + t_\ell(z^{(\ell-1)}_{\leq D})$
    simplest universal NF models
  - spline coupling: $z^{(\ell)}_{>D} = \varphi_\ell(z^{(\ell-1)}_{>D}; z^{(\ell-1)}_{\leq D})$
    ↳ arbitrary function invertible in first arg.
    implement φ_ℓ as a spline ($\hat{=}$ piecewise polynomial or other basis fct.)
    spline: parameters at "knots" $y = \varphi(x)$ 1-1



$\Leftarrow$ knot values

optional: y'_k derivatives at knots ("Hermite spline")

spline: give knot parameters, interpolate to arbitrary x with an analytic rule (typically locally $\hat{=}$ only knot near the query x are relevant)



idea of spline flows: let θ_ℓ be the knot parameters in layer ℓ $\Rightarrow$ define a subnetwork: $\theta_\ell = \psi_\ell\left(z^{(\ell-1)}_{\leq \tilde{D}}\right)$

such that $\varphi_\ell\left(z^{(\ell-1)}_{>\tilde{D}}\,;\,\theta_\ell(z^{(\ell-1)}_{\leq\tilde{D}})\right)$ is invertible in $z^{(\ell-1)}_{>\tilde{D}}$

most popular: "Neural Spline Flow" rational quadratic splines

[NSF | Durkan et al. 2019]

neighbor knots

$$s = \frac{z - z_k}{z_{k+1} - z_k}$$

$$\frac{a_k s^2 + b_k s}{d_k s^2 + e_k s + f_k} + c_k$$

- other proposed NFs: inverse autoregressive flows, i-ResNets, MADE
  more complicated than coupling flows, but not better (according to our experiments)
- state-of-the-art in terms of generative quality: diffusion models } later
- brand new: free-form-flows (merge between NFs and AEs) }

---

conditional NFs: learn conditioned probab. densities $p(X \mid Y=y)$ (instead of $p(x)$)

- example: $Y \hat{=}$ observations, $X \hat{=}$ hidden system properties / states (e.g. class labels)

setting: $TS = \{(x_i, y_i)\}_{i=1}^{N}$, y_i arbitrary condition, $x_i \in \mathbb{R}^D$

use TS to learn $\hat{p}(x \mid y) \approx p^*(x \mid y)$

- easy to do using coupling flows: each layer implements a triangular map

$$z^{(\ell)} = \begin{pmatrix} f_{\ell,1}(z_1^{(\ell-1)}) \\ f_{\ell,2}(z_2^{(\ell-1)}, z_1^{(\ell-1)}) \\ \vdots \\ f_{\ell,D}(z_D^{(\ell-1)}, \dots, z_1^{(\ell-1)}) \end{pmatrix} = f_\ell(z^{(\ell-1)})$$ unconditional

$$z^{(\ell)} = \begin{pmatrix} f_{\ell,1}(z_1^{(\ell-1)}, y) \\ f_{\ell,2}(z_2^{(\ell-1)}, z_1^{(\ell-1)}, y) \\ \vdots \\ f_{\ell,D}(z_D^{(\ell-1)}, \dots, z_1^{(\ell-1)}, y) \end{pmatrix} = f_\ell(z^{(\ell-1)}, y)$$ conditional on y, but still triangular in $z^{(\ell-1)}$

$y \to$ [h] $\to$ [c|NN] $\leftrightarrow z$, $x \leftrightarrow$ [c|NN]

$\Rightarrow$ implement $s_\ell(z^{(\ell-1)}_{\le D}, y)$, $t_\ell(z^{(\ell-1)}_{\le D}, y)$

variant: when y is complicated (e.g. image): train feature detector $h(y)$ $\Rightarrow s_\ell(z^{(\ell-1)}_{\le D}, h(y))$, $t_\ell(z^{(\ell-1)}_{\le D}, h(y))$