Masked Autoregressive Flow for Density Estimation

Midterm 4

- **Problem**: estimating the joint density of samples taken from a set of observations $\mathscr{D} = \{x^1, \ldots, x^N\} \subset \mathbb{R}^D$
- **Neural Density Estimators** (NDE): exact density evaluation, unlike VAE and GAN
- Two families of NDE
  - **Autoregressive Models**: joint density $\longrightarrow$ product of conditionals
  - **Normalizing Flows**: base density $\xrightarrow{\{f_i\}_{i \leq K}}$ target (joint) density
- **Trick**: combine them together in a (deep) hierarchy
  - Differentiable randomness in generating data thanks to autoregression
  - Tractable Jacobians by design, often invertible $\equiv$ normalizing flow

Fabrizio De Castelli - 19/06/2024

Background

Autoregressive Density Estimation

- For a given sample x: joint density into product of one-dimensional conditionals

$$p(x) = p(x_1) \prod_{i=2}^{D} p(x_i \mid x_1, \ldots, x_{i-1})$$

- **Ordering sensitivity**
  - Factorial number of orderings
  - Wrong ordering $\implies$ wrong estimation
  - Need an ordering invariant model

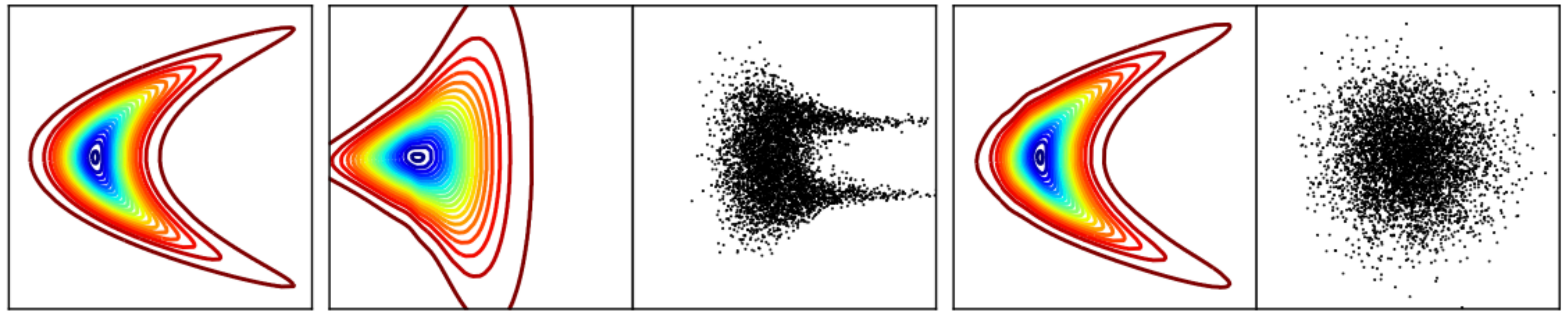
Normalizing Flows

- Given a base density $\pi_u(u)$, we can write

$$x = f(u), \quad u \sim \pi_u(u)$$

$$p(x) = \pi_u\left(f^{-1}(x)\right) \left| \det J_x\left(f^{-1}\right) \right|$$

- Need f **invertible** and tractable Jacobian
- If so, different functions f_i can be **composed** and preserve the property: choose $f = f_1 \circ \ldots \circ f_K$



(a) Target density (b) MADE with Gaussian conditionals (c) MAF with 5 layers

1a: The target density $p(x_1, x_2) = \mathcal{N}\left(x_2 \mid 0, 4\right) \mathcal{N}\left(x_1 \mid 0.25 \cdot x_2^2, 1\right)$

1b: Wrong estimation of Masked Autoencoder for Distribution Estimation (MADE)

1c: Correct estimation of a 5 layer MAF

Masked Autoregressive Flow (MAF)

- **Assumption**: all conditionals are parametrized by a single gaussian (x is our **single** sample)

$$p\left(x_i \mid x_1, \ldots, x_{i-1}\right) = \mathcal{N}\left(x_i \mid \mu_i, e^{\alpha_i^2}\right), \qquad \mu_i = f_{\mu_i}\left(x_1, \ldots, x_{i-1}\right), \qquad \alpha_i = f_{\alpha_i}\left(x_1, \ldots, x_{i-1}\right)$$

- **Autoregressive definition**: model i^{th} variable in terms of previously modeled variables $x_1, \ldots, x_{i-1}$
- **Forward Process**: $x_i = u_i\, e^{\alpha_i} + \mu_i$, with $u_i \sim \mathcal{N}\left(0,1\right)$
- **Inverse Process**: $u_i = \left(x_i - \mu_i\right) e^{-\alpha_i}$
- In this way we have a **tractable Jacobian**: $\left|\det J_x\left(f^{-1}\right)\right| = e^{-\sum_i \alpha_i}$
  - The Jacobian is lower triangular
  - It measures the rate of change of differential volume under a coordinate transformation
- f_{μ_i} and f_{α_i} are chosen with **masking**, following *MADE* $\Longrightarrow$ single forward pass to compute $p\left(x\right)$
- **Idea**: stack many layers to improve the fit. Each layer models the random numbers u_i of the next layer

MAF and Other Flows

- **MAF vs IAF**: in IAF f_{μ_i} and f_{α_i} are defined in terms of previous random numbers $u_1, \ldots, u_{i-1}$ (*MADE*)

$$\mu_i = f_{\mu_i}\left(u_1, \ldots, u_{i-1}\right), \quad \alpha_i = f_{\alpha_i}\left(u_1, \ldots, u_{i-1}\right), \quad p(u) = \pi_x\left(f(u)\right)\left|\det J_u\left(f\right)\right|$$

  - MAF more suitable for density estimation: one pass to get the density from an input x and D passes for sampling
  - IAF more suitable for generation: one pass to get generate and calculate the density of a sample and D passes for an external input
- **MAF vs Real-NVP**: in Real-NVP, coupling layers are used
  - Decide a split of size d, copy the first d elements and transform the remaining $D - d$

$$\begin{aligned} x_{1:d} &= u_{1:d} \\ x_{d+1:D} &= u_{d+1:D} \odot e^{\alpha} + \mu \end{aligned} \qquad \text{where} \qquad \begin{aligned} \mu &= f_\mu(u_{1:d}) \\ \alpha &= f_\alpha(u_{1:d}) \end{aligned}$$

  - MAF $\equiv$ generalization of Real-NVP with $\alpha_i = \mu_i = 0 \ \forall i \leq d$ and $\mu_i = f_{\mu_i}\left(x_{1:d}\right), \ \alpha_i = f_{\alpha_i}\left(x_{1:d}\right)$
- **Conditional MAF**: requires labelled inputs $\mathcal{D} = \{(x^n, y^n)\}_{n=1}^N$
  - Joint **conditional** density of a sample x is $p\left(x \mid y\right) = p\left(x_1 \mid y\right) \prod_{i=2}^{D} p\left(x_i \mid x_1, \ldots, x_{i-1}, y\right)$
  - Any ordering is valid, provided that y comes before $x \equiv$ in a layered architecture, y is an **extra input** for each layer

Key Catch

$$\max_\theta \log \mathscr{L}(\theta) = \max_\theta \sum_n \log p(x^n \mid \theta) \equiv \min_\theta D_{KL}\left(\pi_x(x) \parallel p_x(x)\right) = \min_\theta D_{KL}\left(p_u(u) \parallel \pi_u(u)\right)$$

- $\pi_x(x)$ is the **true density of the data**, $\pi_u(u)$ is the base density of MAF
- **Proof idea**: expand definition of KL, change variable $x \leftrightarrow u$ and use Monte Carlo sampling to get

$$\log \mathscr{L}(\theta) \approx \frac{1}{N}\sum_n \left(\log \pi_x(x^n) - \log p_x(x^n \mid \theta)\right) = -\frac{1}{N}\sum_n \log p_x(x^n \mid \theta) + \text{const}$$

- **Interpretation**
  - IAF: **encoder** with target density $\pi_u(u)$ and transformation f^{-1}, **trained with Stochastic Variational Inference** (SVI) and having base density $\pi_x(x)$ (the MAF target density)
  - **KL Divergences Equality**: due to the approximation $p_x(x)$ of $\pi_x(x)$ starting from prior $\pi_u(u)$ and in the reversed case, the approximation $p_u(u)$ of $\pi_u(u)$ starting from prior $\pi_x(x)$
- **Conclusion**: inverse direction of transformation f leads to equivalence between IAF and MAF. Training the latter as a density estimator of $\pi_x(x)$ is equivalent to perform SVI on an implicit IAF with posterior $\pi_u(u) \Longrightarrow$ MAF very **expressive**

Experimental Setup

- Comparison of MAF with MADE and Real-NVP in **unconditional** and **conditional** density estimation on different datasets. Other models' architectures are chosen following the state-of-the-art in the literature.
- **MADE**: mixture of C gaussians (conditionals). Inputs processing order is the same of the dataset.
    - ***MADE*** $\implies C = 1,$ ***MADE MoG*** $\implies C = 10$
- ***Real-NVP (N)***: two FFNNs of shared architecture as f_α (tanh) and f_μ (ReLU), N is the number of coupling layers with gaussian base density. Copying even and odd indexed elements in coupling layers.
- ***MAF (N)***, two versions with $N \in \{5,10\}$ MADE layers, $\pi_u(u) \sim \mathcal{N}(0,1)$ and one MoG version
    - ***MAF MoG (5)***: 5 autoregressive layers and $C = 10$ gaussians (≡ MAF (5) on top of MADE MoG and trained jointly)
    - First layer has standard input order, successive layers **alternatively reverse** the order (IAF guideline)
- **Batch normalization** after each autoregressive layer in MAF and after each coupling layer in Real-NVP
- **Adam** optimizer, **batch size** 100, **L2 weight decay** $= 10^{-6}$, **early stopping** with 30 epochs patience
- **Five** preprocessed **datasets** from UCI repository: *POWER, GAS, HEPMASS, MINIBOONE, BSDS300*

Results

- **Baseline**: simple Gaussian model (unconditional and class-conditional)
- Metric: **average test log likelihood** (in *nats*, for information entropy)

- **Unconditional** Density Estimation

	POWER	GAS	HEPMASS	MINIBOONE	BSDS300
Gaussian	-7.74 ± 0.02	-3.58 ± 0.75	-27.93 ± 0.02	-37.24 ± 1.07	96.67 ± 0.25
MADE	-3.08 ± 0.03	3.56 ± 0.04	-20.98 ± 0.02	-15.59 ± 0.50	148.85 ± 0.28
MADE MoG	$\mathbf{0.40 \pm 0.01}$	8.47 ± 0.02	$\mathbf{-15.15 \pm 0.02}$	-12.27 ± 0.47	153.71 ± 0.28
Real NVP (5)	-0.02 ± 0.01	4.78 ± 1.80	-19.62 ± 0.02	-13.55 ± 0.49	152.97 ± 0.28
Real NVP (10)	0.17 ± 0.01	8.33 ± 0.14	-18.71 ± 0.02	-13.84 ± 0.52	153.28 ± 1.78
MAF (5)	0.14 ± 0.01	9.07 ± 0.02	-17.70 ± 0.02	$\mathbf{-11.75 \pm 0.44}$	155.69 ± 0.28
MAF (10)	0.24 ± 0.01	$\mathbf{10.08 \pm 0.02}$	-17.73 ± 0.02	-12.24 ± 0.45	154.93 ± 0.28
MAF MoG (5)	0.30 ± 0.01	9.59 ± 0.02	-17.39 ± 0.02	$\mathbf{-11.68 \pm 0.44}$	$\mathbf{156.36 \pm 0.28}$

- **MAF is the best performing model** on 3/5 datasets, MADE on the other 2/5
- On BSDS300, MAF achieves almost equal performance to an **ensemble** of 32 Deep RNADEs
- Tie on MINIBOONE $\Longrightarrow$ statistical comparison $\Longrightarrow$ MAF MoG (5) **outperforms** MAF (5)

- **Conditional** Density Estimation

	MNIST		CIFAR-10	
	unconditional	conditional	unconditional	conditional
Gaussian	-1366.9 ± 1.4	-1344.7 ± 1.8	2367 ± 29	2030 ± 41
MADE	-1380.8 ± 4.8	-1361.9 ± 1.9	147 ± 20	187 ± 20
MADE MoG	$\mathbf{-1038.5 \pm 1.8}$	$\mathbf{-1030.3 \pm 1.7}$	-397 ± 21	-119 ± 20
Real NVP (5)	-1323.2 ± 6.6	-1326.3 ± 5.8	2576 ± 27	2642 ± 26
Real NVP (10)	-1370.7 ± 10.1	-1371.3 ± 43.9	2568 ± 26	2475 ± 25
MAF (5)	-1300.5 ± 1.7	-1302.9 ± 1.7*	2936 ± 27	2983 ± 26*
MAF (10)	-1313.1 ± 2.0	-1316.8 ± 1.8*	$\mathbf{3049 \pm 26}$	$\mathbf{3058 \pm 26}$*
MAF MoG (5)	-1100.3 ± 1.6	-1092.3 ± 1.7	2911 ± 26	2936 ± 26

- Uniform prior over labels $p(y) = 1/10$
- **Preprocessing**: pixels → add noise → logit space → augmentation (horizontal flips)
- **MNIST**: MADE is the best
- **CIFAR-10**: MAF is the best. MADE outperformed by the baseline

Final Conclusions

- MAF: **strong model** able to process all kinds of data (generalization of Real-NVP and NICE)
- **Idea**: improve MADE creating an autoregressive flow modeling internal random numbers with more layers
- **Strong** points
  - **Flexible**: stack many autoregressive models to compose a flow
  - **Efficient**: no recursion thanks to masking. $p(x)$ is computed in one single forward pass
  - Can model conditional densities with no additional costs
  - General purpose density estimation, independently from the domain knowledge
- **Weakness**
  - **Slow sampling**: cannot exploit parallelization when starting from noise (forward process) $\Longrightarrow$ more suitable for density estimation of input samples
- Possible future works: **deep** MAF, **regularization** techniques (not only L2), **dynamic masking** algorithms