Neural Autoregressive Flows

Нугаманов Эдуард

ВШЭ, 2018

Вариационный вывод

- Используется в: модели со скрытыми переменными, генерация изображений
- Приближаем апостериорную плотность в классе известных распределений, например, mean-field
- Даже асимптотически нет гарантии достижения правильного распределения (в отличие от MCMC)

Подходы

- Structured mean-field
- Приближать смесью распределений – требует вычислений лог-правдоподобия и его градиентов относительно каждой компоненты на шаг обновления параметров
- Normalizing Flows

Finite Normalizing Flows

$$z_K = f_K \circ \cdots \circ f_2 \circ f_1(z_0)$$

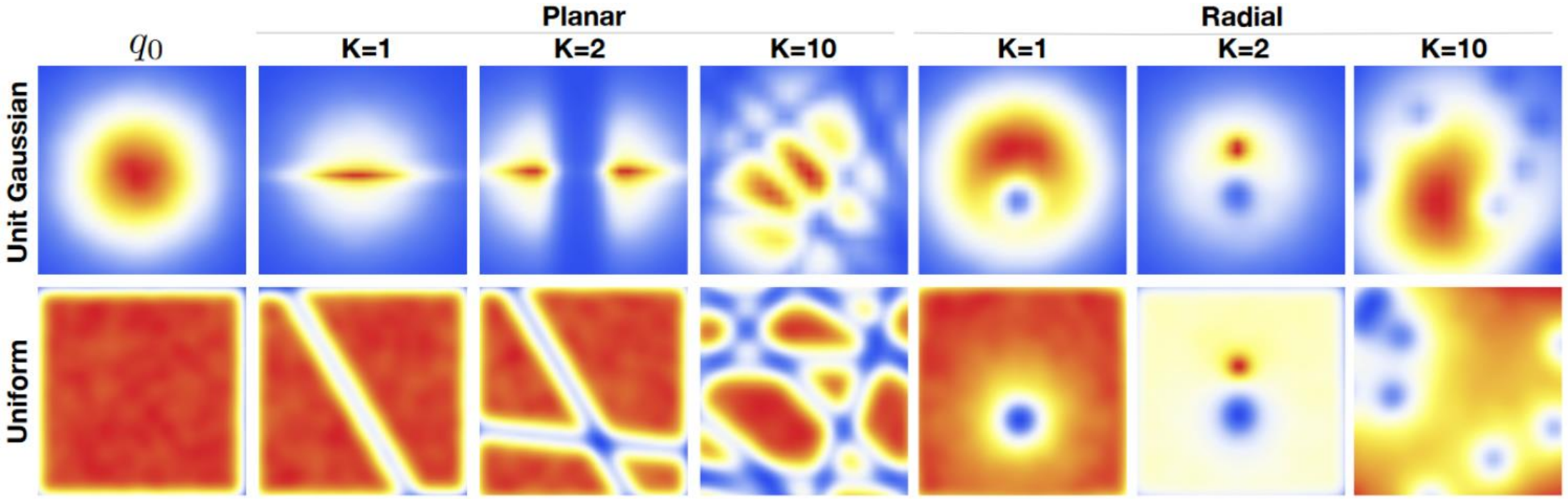
$$\ln q_K(z_K) = \ln q_0(z_0) - \sum_{k=1}^{K} \ln \left| \det \frac{\partial f_k}{\partial z_{k-1}} \right|$$

- Получаем из $z_0 \sim q_0(z)$ случайную величину $z_K \sim q_K(z)$ с помощью цепочки обратимых преобразований f_k
- Пользуемся теоремой о производной обратной функции

FNF: смысл

- Последовательность плотностей q_k - нормализующий поток
- Можно рассматривать как последовательность увеличений и уменьшений плотности в некоторых областях
- Можем начать с простых распределений и, применяя NF различной длины достичь сложных и мультимодальных распределений

FNF: примеры



Недостатки обычных NF

- Плохо масштабируются в пространствах высокой размерности
- Не используют топологию пространства скрытых переменных

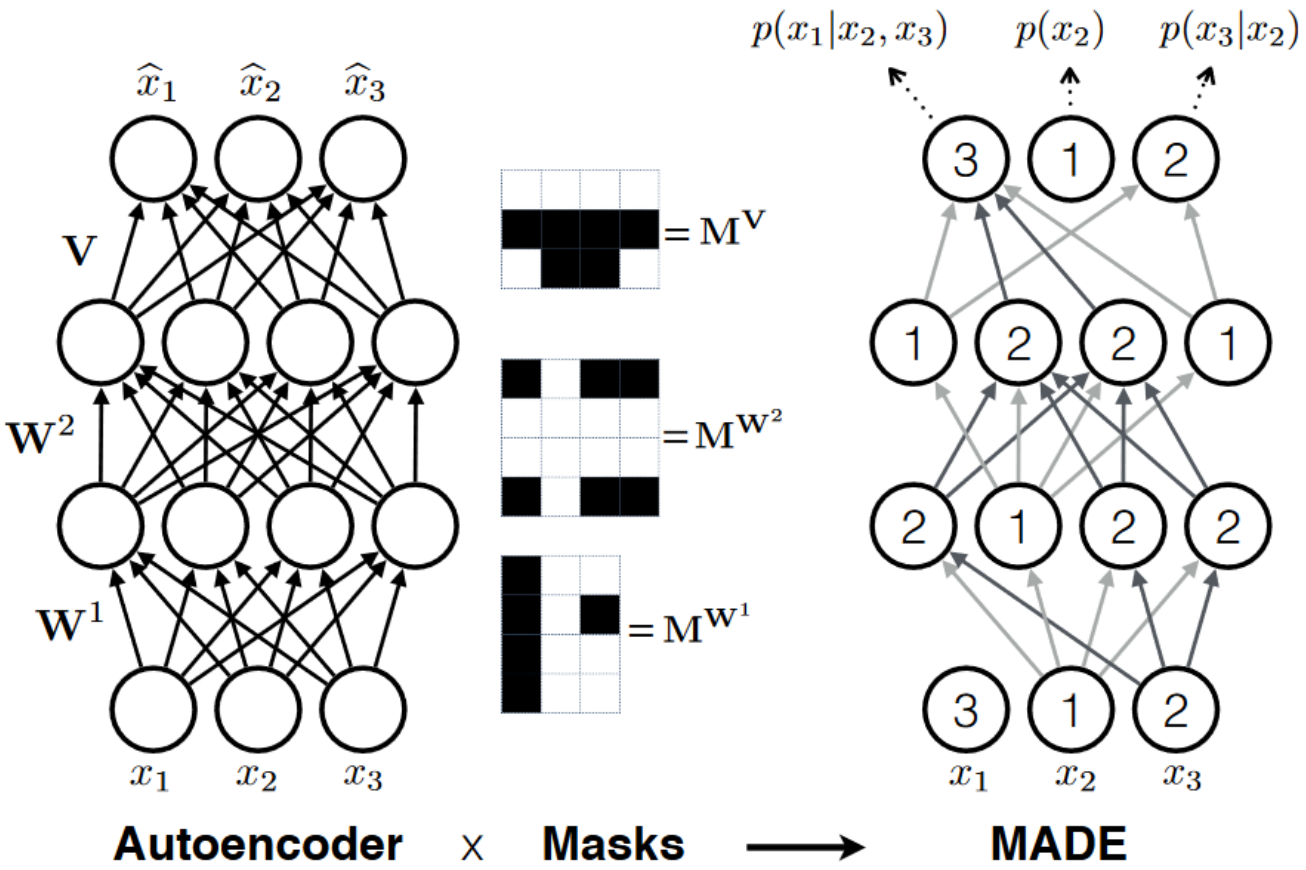
Авторегрессионные модели

- По правилу произведения можем разложить:

$$p(\mathbf{x}) = \prod_{d=1}^{D} p(x_d|\mathbf{x}_{<d})$$

- Каждый выход $\widehat{x_d} = p(x_d|\mathbf{x}_{<d})$ зависит только от $\mathbf{x}_{<d}$
- Таким образом получаем из выходов автоэнкодера валидные вероятности

Авторегрессионные модели: MADE



Affine Autoregressive Flows

- Улучшение стандартных NF
- Состоят из: *conditioner*, c, и *transformer,* τ:

$$y_t = f(x_{1:t}) = \tau(c(x_{1:t-1}), x_t)$$

- *Conditioner* – авторегрессионная модель, *transformer* – обратимая функция
- Эффективно вычисляем все выходы c за один проход с MADE

AAF: примеры трансформеров

- $\mu \in \mathbb{R}, \sigma > 0$ – выход из *conditioner*

- $\tau(\mu, \sigma, x_t) = \mu + \sigma x_t$, σ из экспоненциальной нелинейности
- $\tau(\mu, \sigma, x_t) = \sigma x_t + (1 - \sigma)\mu$, σ из сигмоидальной нелинейности

- Выразительность зависит полностью от сложности *conditioner* и стэкинга множества AAF

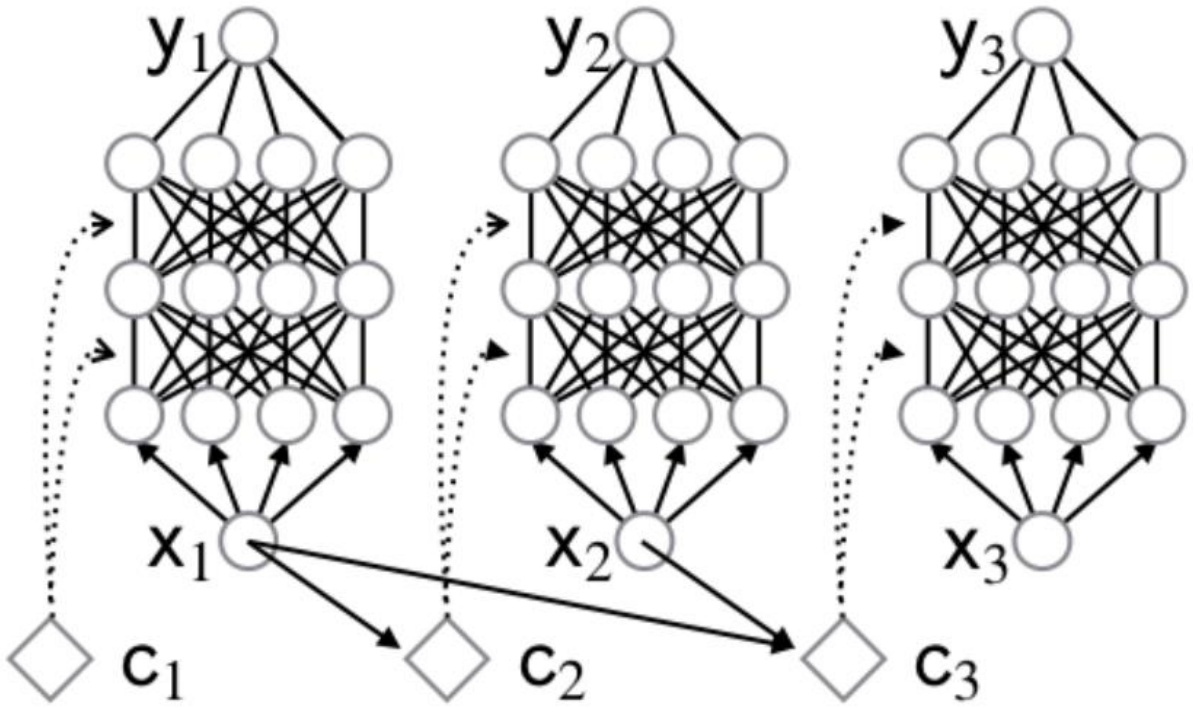
Neural Autoregressive Flows

- Заменим аффинный *transformer* на нейросеть:

$$\tau(c(x_{1:t-1}), x_t) = DNN(x_t; \phi = c(x_{1:t-1}))$$

- Принимает на вход скаляр x_t, выдает скаляр y_t
- Параметры для сети выдает *conditioner*

NAF: иллюстрация



(a) Neural autoregressive flows (NAF)

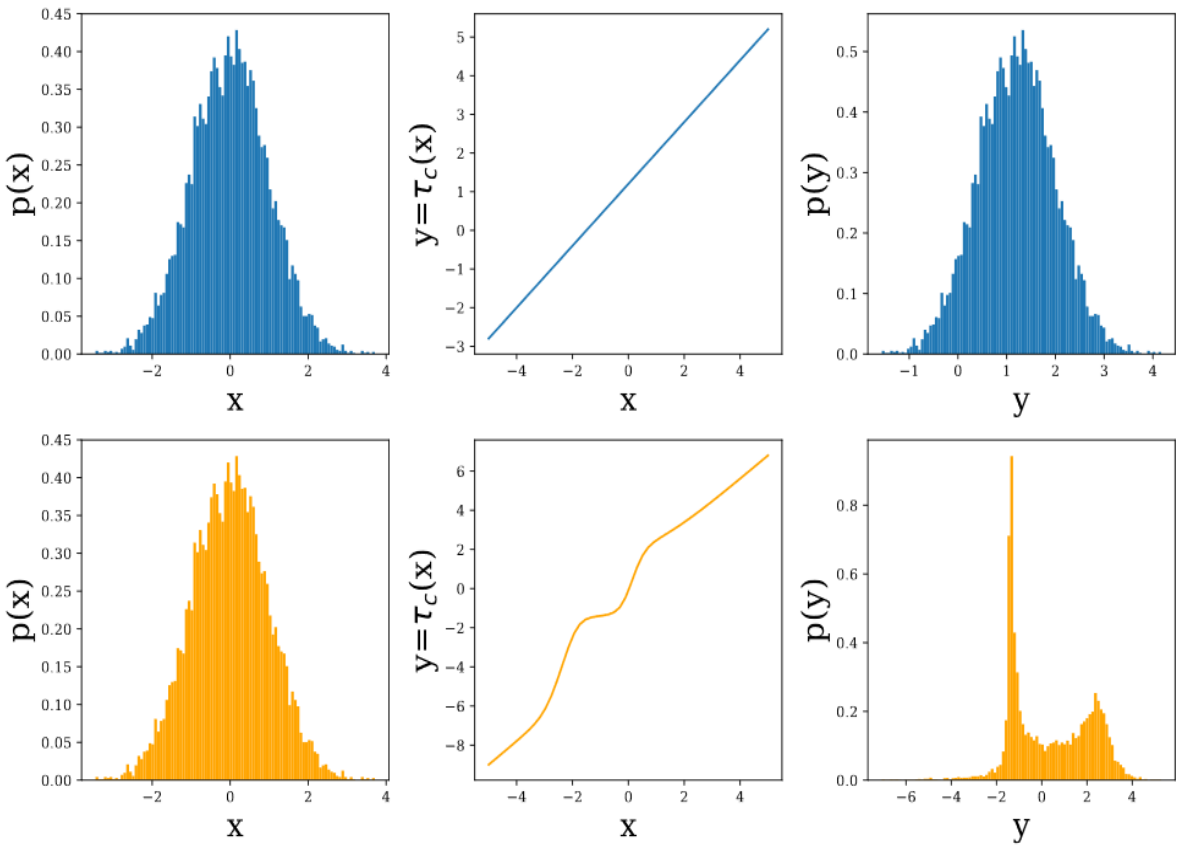
NAF: корректность

- Строго положительные веса + строго монотонные активации = строго монотонный (а значит, обратимый) NAF
- $\frac{dy_t}{dx_t}$ и градиенты по ϕ легко вычисляются с backpropagation
- Градиенты по ϕ проходят через *conditioner,* чтобы обучить и его

NAF: мультимодальность

- τ_c можно рассматривать как аналог CDF

- Точки изгиба τ_c соответствуют локальным минимумам и максимумам в PDF



NAF: примеры архитектур трансформера

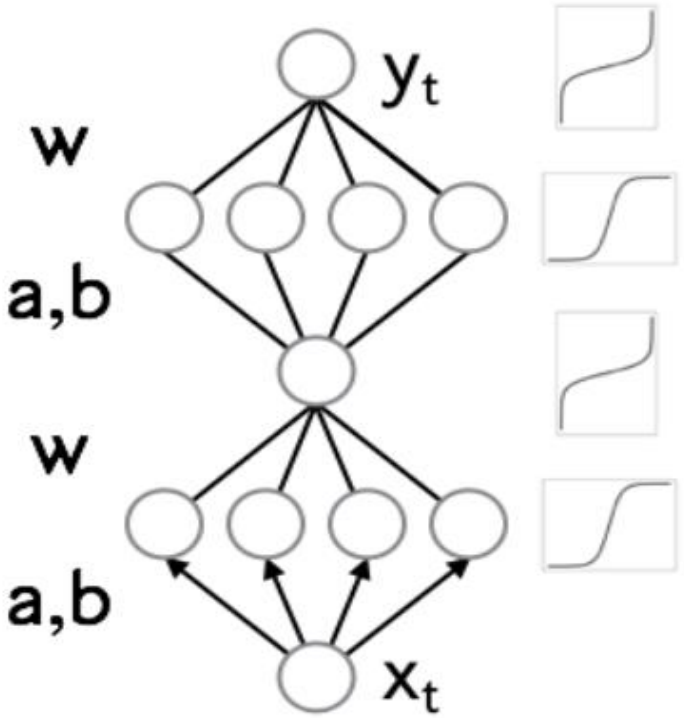
- Deep Sigmoidal Flows, самый простой вариант, но и его достаточно
- Deep Dense SF
- Другие, например, Leaky ReLU

Deep Sigmoidal Flow

$$y_t = \sigma^{-1}(w^T \cdot \sigma(a \cdot x + b)))$$

Где $0 < w_{i,j} < 1; \sum w_{i,j} = 1, a_{s,t} > 0.$

- Активации ведут в новое пространство
- Аффинные преобразования там = нелинейные в старом
- Возвращаемся в старое

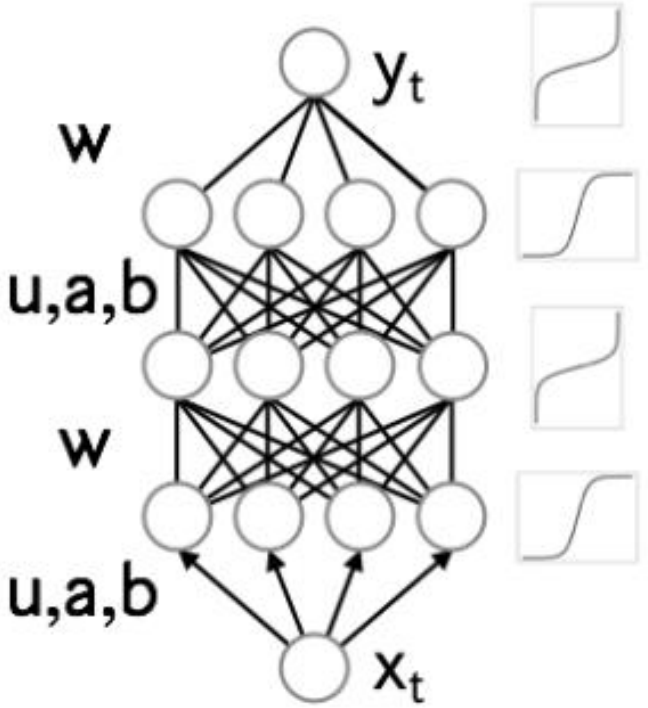


(b) DSF

Deep Dense Sigmoidal Flow

$$h^{(l+1)} =$$
$$= \sigma^{-1}\left(w^{(l+1)} \cdot \sigma(a^{(l+1)} \odot u^{(l+1)} \cdot h^{(l)} + b^{(l+1)})\right)$$

- $h_0 = x_t, h_L = y_t;\ \ d_0 = d_L = 1$
- $\sum_j w_{i,j} = 1; \sum_j u_{k,j} = 1$
- Все параметры положительны



(c) DDSF

Теорема о DSF (без доказательства)

Пусть X – случайный вектор на открытом множестве $U \in \mathbb{R}^m$. Пусть Y – случайный вектор в $\mathbb{R}^m$. Предположим, и X, и Y имеют положительную непрерывную плотность. Тогда существует последовательность функций $(K_n)_{n\geq 1}$:

$$K(x)_t = DSF\left(x_t, c(x_{1:t-1})\right)$$

Таких, что последовательность $Y_n = K_n(X)$ сходится по распределению к Y.

Проще: можем получить любое распределение из любого.

Toy energy fitting

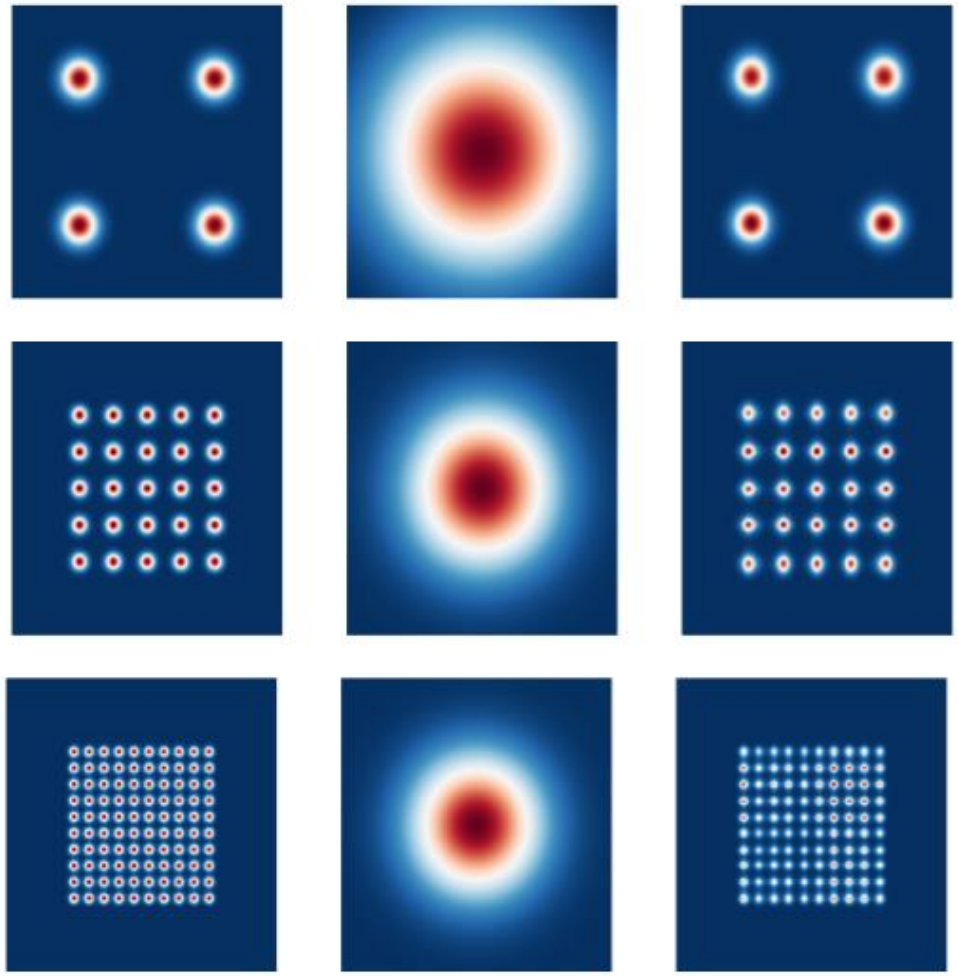


Figure 6. Fitting grid of Gaussian distributions using maximum likelihood. Left: true distribution. Center: affine autoregressive flow (AAF). Right: neural autoregressive flow (NAF)

Sine Wave

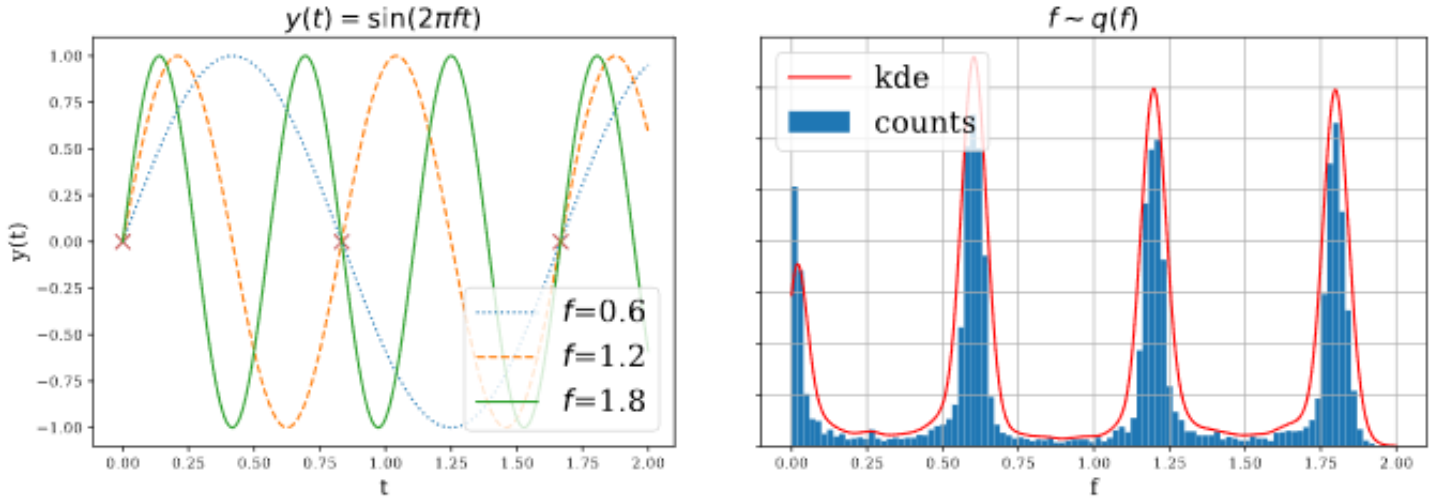


Figure 8. The DSF model effectively captures the true posterior distribution over the frequency of a sine wave. Left: The three observations (marked with red x's) are compatible with sine waves of frequency $f \in 0.0, 0.6, 1.2, 1.8$. Right: a histogram of samples from the DSF approximate posterior ("counts") and a Kernel Density Estimate of the distribution it represents (KDE).

Оценка плотности

Table 2. Test log-likelihood and error bars of 2 standard deviations on the 5 datasets (5 trials of experiments). Neural autoregressive flows (NAFs) produce state-of-the-art density estimation results on all 5 datasets. The numbers (5 or 10) in parantheses indicate the number of transformations which were stacked; for TAN (Oliva et al., 2018), we include their best results, achieved using different architectures on different datasets. We also include validation results to give future researchers a fair way of comparing their methods with ours during development.

Model	POWER	GAS	HEPMASS	MINIBOONE	BSDS300
MADE MoG	0.40 ± 0.01	8.47 ± 0.02	-15.15 ± 0.02	-12.27 ± 0.47	153.71 ± 0.28
MAF-affine (5)	0.14 ± 0.01	9.07 ± 0.02	-17.70 ± 0.02	-11.75 ± 0.44	155.69 ± 0.28
MAF-affine (10)	0.24 ± 0.01	10.08 ± 0.02	-17.73 ± 0.02	-12.24 ± 0.45	154.93 ± 0.28
MAF-affine MoG (5)	0.30 ± 0.01	9.59 ± 0.02	-17.39 ± 0.02	-11.68 ± 0.44	156.36 ± 0.28
TAN (various architectures)	0.48 ± 0.01	11.19 ± 0.02	-15.12 ± 0.02	-11.01 ± 0.48	157.03 ± 0.07
MAF-DDSF (5)	$\mathbf{0.62 \pm 0.01}$	11.91 ± 0.13	$\mathbf{-15.09 \pm 0.40}$	$\mathbf{-8.86 \pm 0.15}$	$\mathbf{157.73 \pm 0.04}$
MAF-DDSF (10)	0.60 ± 0.02	$\mathbf{11.96 \pm 0.33}$	-15.32 ± 0.23	-9.01 ± 0.01	157.43 ± 0.30
MAF-DDSF (5) valid	0.63 ± 0.01	11.91 ± 0.13	15.10 ± 0.42	-8.38 ± 0.13	172.89 ± 0.04
MAF-DDSF (10) valid	0.60 ± 0.02	11.95 ± 0.33	15.34 ± 0.24	-8.50 ± 0.03	172.58 ± 0.32

Заключение

- NAF – метод, способный моделировать сложные мультимодальные распределения и превосходящий иные существующие методы
- NAF производителен

Список литературы

- Variational Inference with Normalizing Flows
  https://arxiv.org/pdf/1505.05770.pdf
- MADE: Masked Autoencoder for Distribution Estimation
  https://arxiv.org/pdf/1502.03509.pdf
- Improved Variational Inference with Inverse Autoregressive Flow
  https://arxiv.org/pdf/1606.04934.pdf
- Neural Autoregressive Flows
  https://arxiv.org/pdf/1804.00779.pdf

Спасибо за внимание!