YOUR CLASSIFIER IS SECRETLY AN ENERGY BASED MODEL AND YOU SHOULD TREAT IT LIKE ONE

Bondarenko Nataliia, AMI171

Преимущества и недостатки генеративных моделей

- Одно решение для разных задач
- Можно использовать неразмеченные данные
- На самом деле процесс сосредоточен на улучшении данных или правдоподобия
- Фокус не на других задачах, к которым можно применять ганы
- Архитектуры расходятся с дискриминативными моделями
- Генеративные модели хуже дискриминативных в дискриминативных задачах

Энергетические модели

Плотность: $p_\theta(\mathbf{x}) = \frac{\exp(-E_\theta(\mathbf{x}))}{Z(\theta)}$

Данные: $\mathbf{x} \in \mathbb{R}^D$

Функция энергии: $E_\theta(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}$

Нормализующая константа: $Z(\theta) = \int_{\mathbf{x}} \exp(-E_\theta(\mathbf{x}))$

- Для большинства $E_\theta(\mathbf{x})$ сложно оценить $Z(\theta)$, отсюда максимизировать правдоподобие сложно
- Используем другие методы обучения

Оценка производной максимального правдоподобия

Для одного элемента:

$$\frac{\partial \log p_\theta(\mathbf{x})}{\partial \theta} = \mathbb{E}_{p_\theta(\mathbf{x}')}\left[\frac{\partial E_\theta(\mathbf{x}')}{\partial \theta}\right] - \frac{\partial E_\theta(\mathbf{x})}{\partial \theta}$$

- Используем марковские цепи монте-карло (MCMC), чтобы семплировать из $p_\theta(\mathbf{x})$

Последние результаты (SGLD): $p_0(\mathbf{x})$, α меняется полиномиально

$$\mathbf{x}_0 \sim p_0(\mathbf{x}), \qquad \mathbf{x}_{i+1} = \mathbf{x}_i - \frac{\alpha}{2}\frac{\partial E_\theta(\mathbf{x}_i)}{\partial \mathbf{x}_i} + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \alpha)$$

Оценка производной максимального правдоподобия

Для одного элемента:

$$\frac{\partial \log p_\theta(\mathbf{x})}{\partial \theta} = \mathbb{E}_{p_\theta(\mathbf{x}')}\left[\frac{\partial E_\theta(\mathbf{x}')}{\partial \theta}\right] - \frac{\partial E_\theta(\mathbf{x})}{\partial \theta}$$

- Используем марковские цепи монте-карло (MCMC), чтобы семплировать из $p_\theta(\mathbf{x})$

Последние результаты (SGLD): $p_0(\mathbf{x})$, α меняется полиномиально

$$\mathbf{x}_0 \sim p_0(\mathbf{x}), \qquad \mathbf{x}_{i+1} = \mathbf{x}_i - \frac{\alpha}{2}\frac{\partial E_\theta(\mathbf{x}_i)}{\partial \mathbf{x}_i} + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \alpha)$$

Что скрывает классификатор

Логиты: $f_\theta : \mathbb{R}^D \to \mathbb{R}^K$

Используются в софтмаксе: $p_\theta(y \mid \mathbf{x}) = \frac{\exp\left(f_\theta(\mathbf{x})[y]\right)}{\sum_{y'} \exp\left(f_\theta(\mathbf{x})[y']\right)}$

Переинтерпретируем их, чтобы получить совместную плотность:

$$p_\theta(\mathbf{x}, y) = \frac{\exp\left(f_\theta(\mathbf{x})[y]\right)}{Z(\theta)}$$

Здесь: $E_\theta(\mathbf{x}, y) = -f_\theta(\mathbf{x})[y]$

$Z(\theta)$ — неизвестная нормализующая константа

JEM

$$p_\theta(\mathbf{x}) = \sum_y p_\theta(\mathbf{x}, y) = \frac{\sum_y \exp\left(f_\theta(\mathbf{x})[y]\right)}{Z(\theta)}$$

- Сдвиг логитов на константу влияет на $\log p_\theta(\mathbf{x})$

Получим условную вероятность: $p_\theta(y|\mathbf{x}) = \frac{p_\theta(\mathbf{x}, y)}{p_\theta(\mathbf{x})}$

Заметим:

$$E_\theta(\mathbf{x}) = -\mathbf{LogSumExp}_y(f_\theta(\mathbf{x})[y]) = -\log \sum_y \exp(f_\theta(\mathbf{x})[y])$$

JEM

$$p_\theta(\mathbf{x}) = \sum_y p_\theta(\mathbf{x}, y) = \frac{\sum_y \exp\left(f_\theta(\mathbf{x})[y]\right)}{Z(\theta)}$$

- Сдвиг логитов на константу влияет на $\log p_\theta(\mathbf{x})$

Получим условную вероятность: $p_\theta(y|\mathbf{x}) = \frac{p_\theta(\mathbf{x}, y)}{p_\theta(\mathbf{x})}$

Заметим:

$$E_\theta(\mathbf{x}) = -\mathbf{LogSumExp}_y(f_\theta(\mathbf{x})[y]) = -\log \sum_y \exp(f_\theta(\mathbf{x})[y])$$

Оптимизация

Факторизация:

$$p_\theta(y \mid \mathbf{x}) = \frac{\exp\left(f_\theta(\mathbf{x})[y]\right)}{\sum_{y'} \exp\left(f_\theta(\mathbf{x})[y']\right)}$$

$$p_\theta(\mathbf{x}) = \sum_y p_\theta(\mathbf{x}, y) = \frac{\sum_y \exp\left(f_\theta(\mathbf{x})[y]\right)}{Z(\theta)}$$

$$\log p_\theta(\mathbf{x}, y) = \log p_\theta(\mathbf{x}) + \log p_\theta(y|\mathbf{x})$$

Оптимизация

- Оптимизируем $p(y|\mathbf{x})$ с помощью кросс-энтропии
- Оптимизируем $\log p(\mathbf{x})$ с помощью

$$\frac{\partial \log p_\theta(\mathbf{x})}{\partial \theta} = \mathbb{E}_{p_\theta(\mathbf{x}')}\left[\frac{\partial E_\theta(\mathbf{x}')}{\partial \theta}\right] - \frac{\partial E_\theta(\mathbf{x})}{\partial \theta}$$

Используем SGLD (Stochastic gradient Langevin dynamics).

Градиенты берем по $\mathbf{LogSumExp}_y(f_\theta(x)[y])$

Algorithm 1 JEM training: Given network f_θ, SGLD step-size α, SGLD noise σ, replay buffer B, SGLD steps η, reinitialization frequency ρ

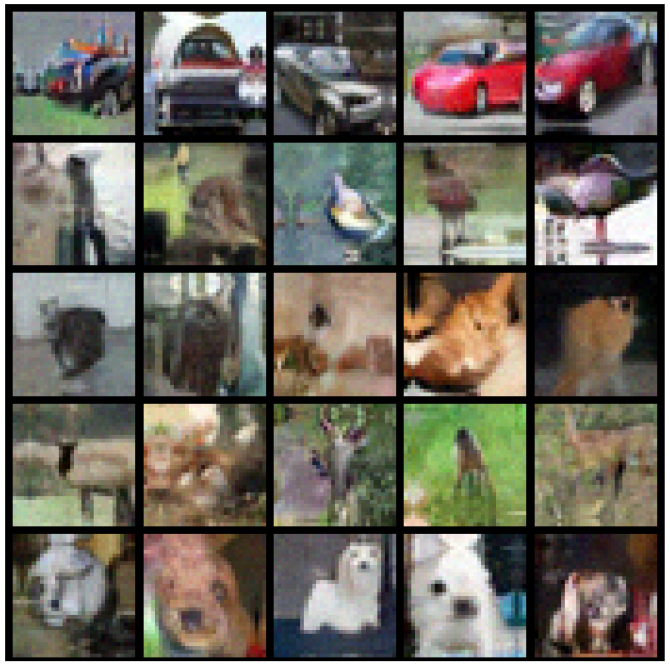
while not converged **do**
  Sample $\mathbf{x}$ and y from dataset
  $L_{\text{clf}}(\theta) = \text{xent}(f_\theta(\mathbf{x}), y)$
  Sample $\widehat{\mathbf{x}}_0 \sim B$ with probability $1 - \rho$, else $\widehat{\mathbf{x}}_0 \sim \mathcal{U}(-1, 1)$ ▷ Initialize SGLD
  for $t \in [1, 2, \ldots, \eta]$ **do** ▷ SGLD
    $\widehat{\mathbf{x}}_t = \widehat{\mathbf{x}}_{t-1} + \alpha \cdot \frac{\partial \text{LogSumExp}_{y'}(f_\theta(\widehat{\mathbf{x}}_{t-1})[y'])}{\partial \widehat{\mathbf{x}}_{t-1}} + \sigma \cdot \mathcal{N}(0, I)$
  end for
  $L_{\text{gen}}(\theta) = \text{LogSumExp}_{y'}(f(\mathbf{x})[y']) - \text{LogSumExp}_{y'}(f(\widehat{\mathbf{x}}_t)[y'])$ ▷ Surrogate for Eq 2
  $L(\theta) = L_{\text{clf}}(\theta) + L_{\text{gen}}(\theta)$
  Obtain gradients $\frac{\partial L(\theta)}{\partial \theta}$ for training
  Add $\widehat{\mathbf{x}}_t$ to B
end while

Результаты

	Accuracy
SVHN	96.7%
CIFAR100	72.2%

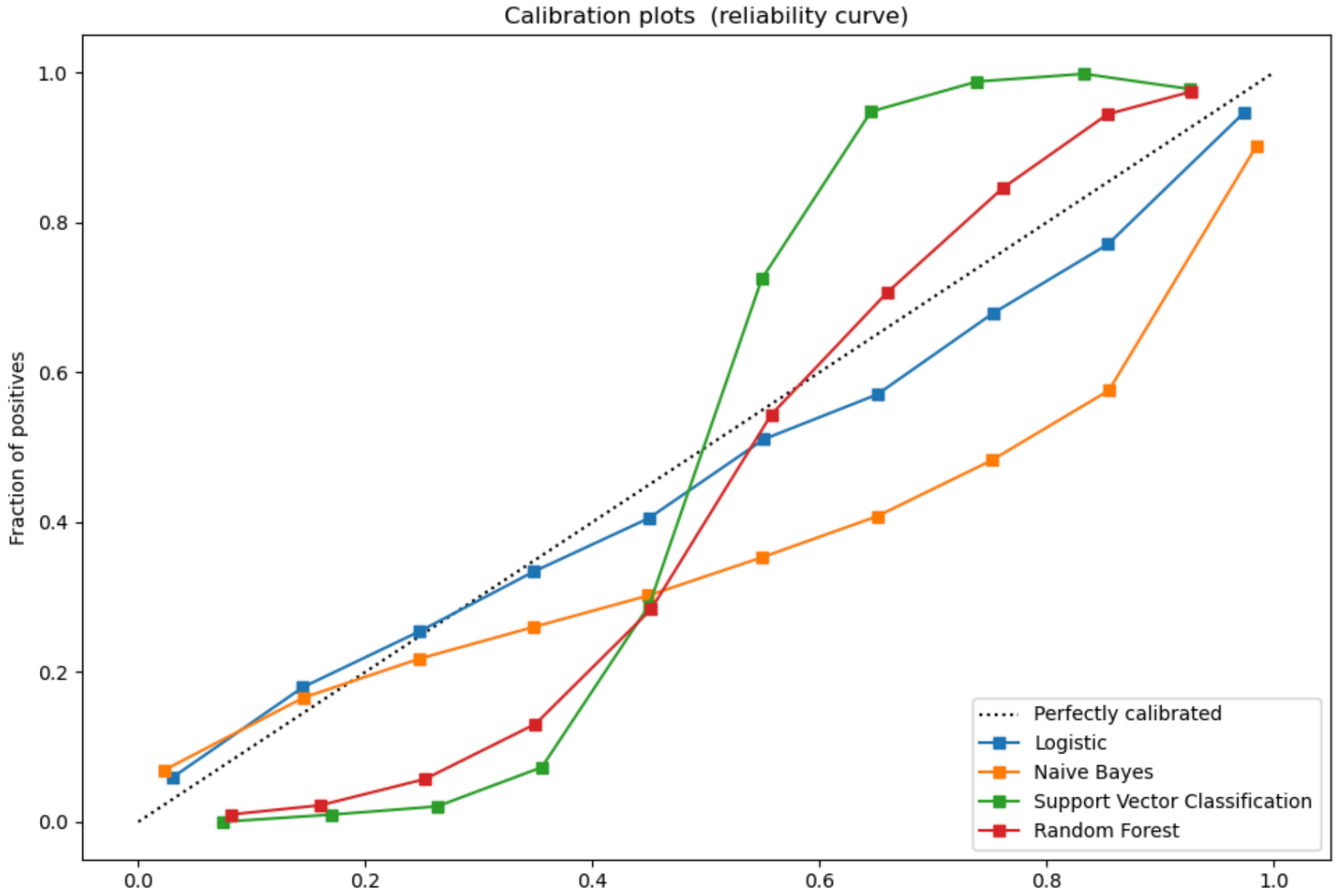
CIFAR10:

Class	Model	Accuracy% ↑	IS↑	FID↓
Hybrid	Residual Flow	70.3	3.6	46.4
	Glow	67.6	3.92	48.9
	IGEBM	49.1	8.3	**37.9**
	JEM $p(\mathbf{x}\|y)$ factored	30.1	6.36	61.8
	JEM (Ours)	**92.9**	**8.76**	38.4
Disc.	Wide-Resnet	95.8	N/A	N/A
Gen.	SNGAN	N/A	8.59	25.5
	NCSN	N/A	8.91	25.32

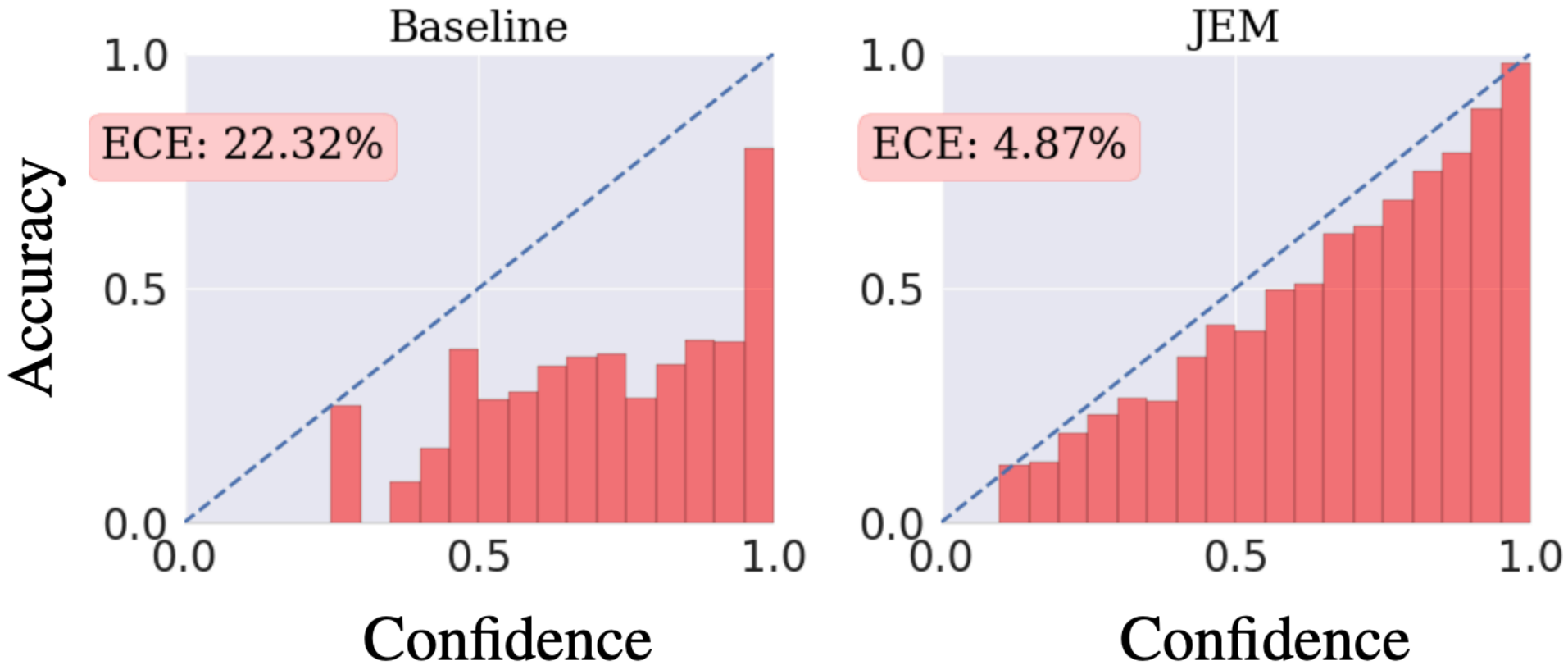


Калибровка

Откалиброванный классификатор полезнее точного



JEM точен и откалиброван



CIFAR100: точность – 72% (у ResNet-110 74,8%)

Out of distribution detection

$$s_\theta(\mathbf{x}) \in \mathbb{R}$$

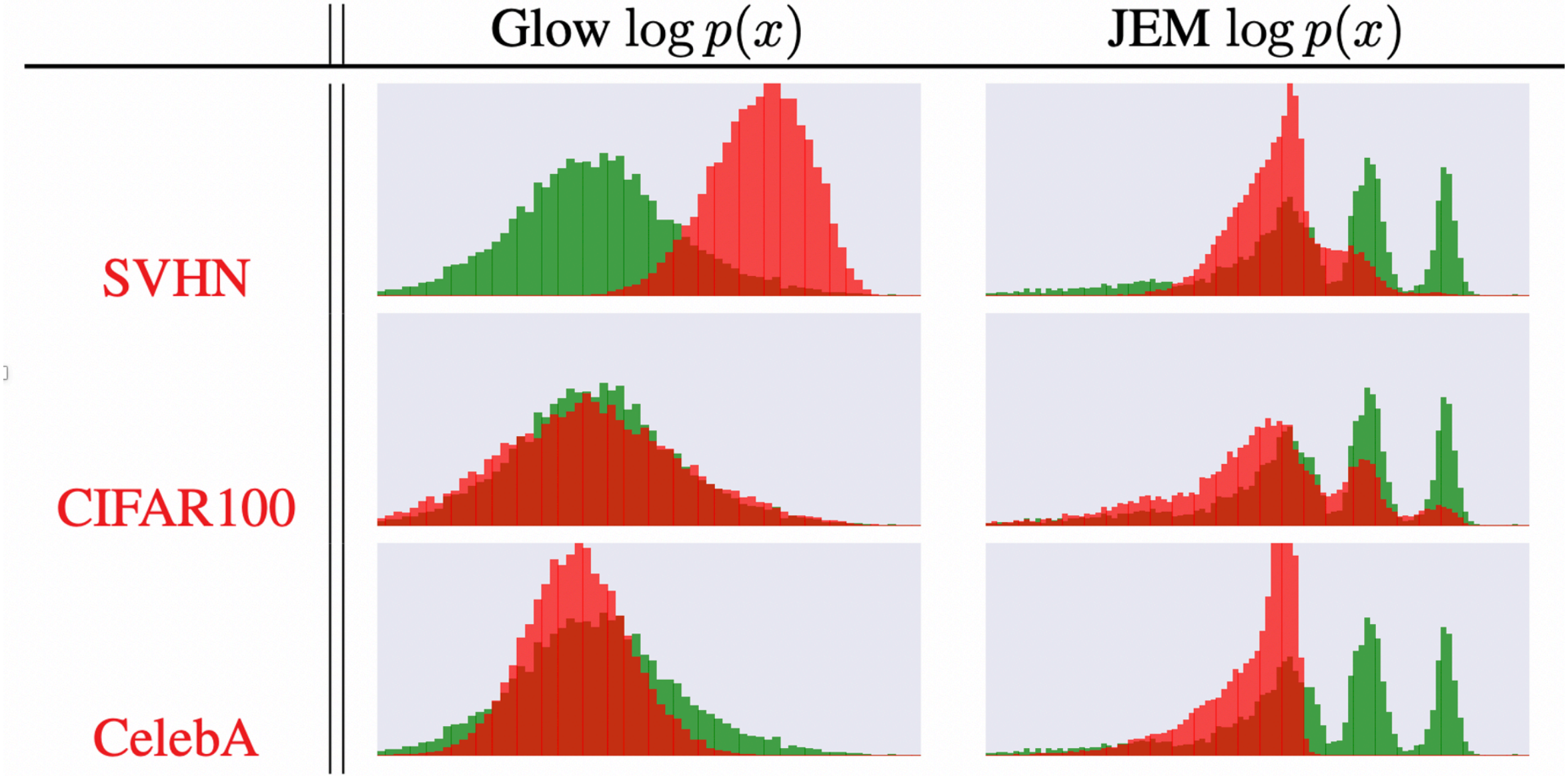
Выше для объектов из распределения

Ниже для объектов из других распределений

Out of distribution detection

- Подогнать плотность под данные. Низкая плотность = OOD
- $s_\theta(\mathbf{x}) = \max_y p_\theta(y|\mathbf{x})$
- Учитываем не только точку, но и окружение: $s_\theta(\mathbf{x}) = -\left|\left|\frac{\partial \log p_\theta(\mathbf{x})}{\partial \mathbf{x}}\right|\right|_2$

  Ожидаем, что если точка – случайный пик, то плотность вокруг нее быстро будет снижаться, поэтому смотрим на норму градиента


Glow $\log p(x)$
JEM $\log p(x)$
SVHN
CIFAR100
CelebA

$s_\theta(\mathbf{x})$	Model	CIFAR10 SVHN	CIFAR10 Interp	CIFAR100	CelebA
$\log p(\mathbf{x})$	Unconditional Glow	.05	.51	.55	.57
	Class-Conditional Glow	.07	.45	.51	.53
	IGEBM	.63	**.70**	.50	.70
	JEM (Ours)	**.67**	.65	**.67**	**.75**
$\max_y p(y\|\mathbf{x})$	Wide-ResNet	**.93**	**.77**	.85	.62
	Class-Conditional Glow	.64	.61	.65	.54
	IGEBM	.43	.69	.54	.69
	JEM (Ours)	.89	.75	**.87**	**.79**
$\left\|\left\|\frac{\partial \log p(\mathbf{x})}{\partial \mathbf{x}}\right\|\right\|$	Unconditional Glow	**.95**	.27	.46	.29
	Class-Conditional Glow	.47	.01	.52	.59
	IGEBM	.84	.65	.55	.66
	JEM (Ours)	.83	**.78**	**.82**	**.79**

Adversarial attacks

Пробовали атаки:

- white-box PGD-атаку (с доступом к градиентам)
- gradient-free black-box атаку (без доступа к градиентам)
- the boundary attack
- the brute-force pointwise attack

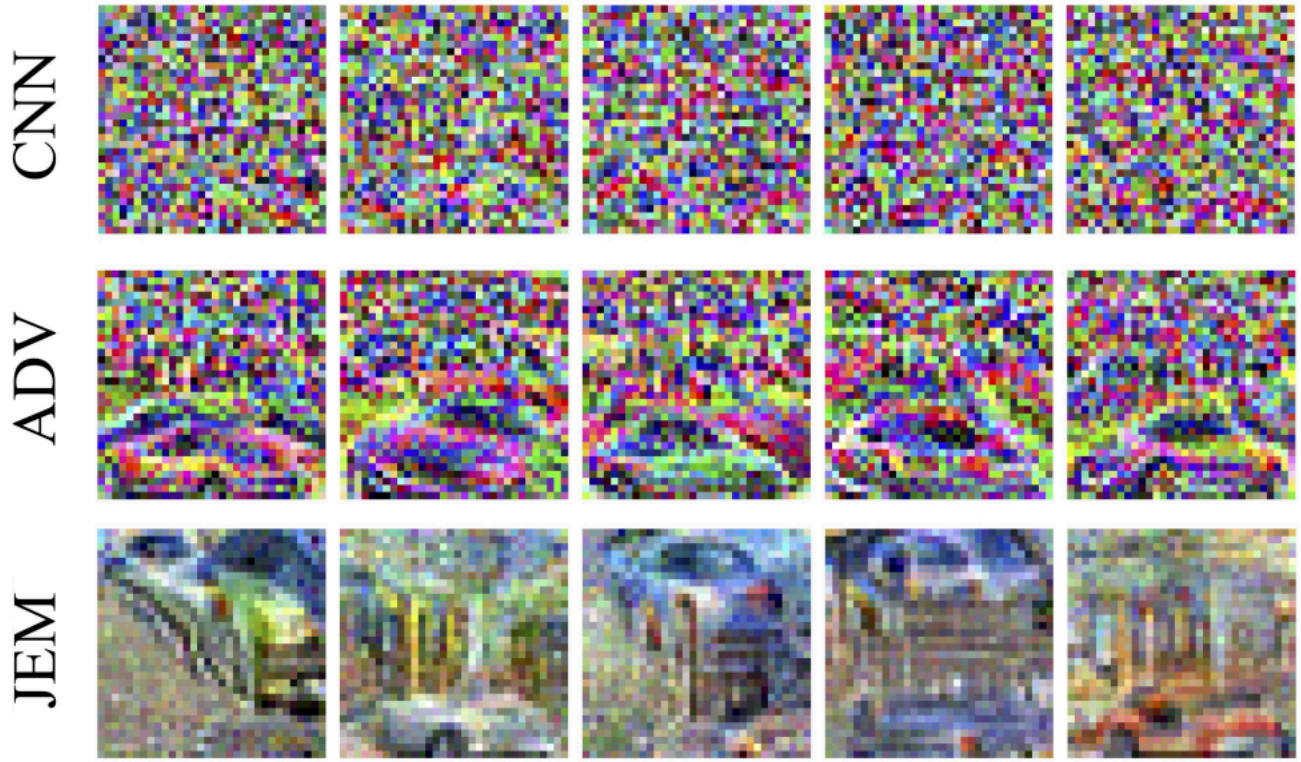
Относительно норм L_∞ и L_2

JEM лучше базовой модели во всех случаях

Еще одна проблема

Некоторые модели могут уверенно классифицировать бессмысленные данные

Максимизируем $p(y = \text{“car”}|\mathbf{x})$, начиная со случайного шума



Сложности

- Нет нормализованных вероятностей – сложно проверить, что вообще идет обучение. Картинки можно нарисовать, но это не общая стратегия
- Оценки градиента нестабильны и будут расходиться без грамотной настройки гиперпараметров

Есть, куда расти!

Вопросы

- Что и как оптимизируют авторы статьи?
- Какие есть проблемы у JEM?
- Как получить совместную плотность $p_\theta(\mathbf{x}, y)$ и $p_\theta(\mathbf{x})$