Semi-supervised Learning with Deep Generative Models

Хайруллин Ринат

Московский физико-технический институт

План доклада

- Вариационный вывод и SVGB.
- Deep Generative Models для задачи SSL.
- Обучение модели.
- Модификация DGM (ADGM).
- Результаты.

Байесовский вывод

$p(\theta)$ – априорные знания о величине θ. $\mathbf{x} = \{x_i\}_1^n$ – серия наблюдений, при разных θ наблюдения $\mathbf{x}$ определяются правдоподобием $p(x|\theta)$.

$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{\int p(x|\theta)p(\theta)d\theta}$, при этом θ – может быть векторозначной. При большой размерности θ численное интегрирование с требуемой точностью может быть невозможно.

Сопряженные распределения:
$p(\theta) \sim \mathcal{A}(\alpha_0)$, $p(x|\theta) \sim \mathcal{B}(\beta)$ – сопряженные, если $p(\theta|x) \sim \mathcal{A}(\alpha_1)$.

Пример сопряженных распределений: $p(m|n,q) \sim \mathcal{B}(m|n,q)$, $p(q|a,b) \sim Beta\ (q|a,b)$.

Если априорное распределение выбрано из класса распределений, сопряженных правдоподобию, то апостериорное распределение можно выписать в явном виде.

Вывод в вероятностных моделях

$p(X, Y, Z)$ – вероятностная модель.
X – известные переменные.
Y – требуется оценить.
Z – неизвестные переменные.

Задача вывода:
$p(Y|X) = \frac{p(Y,X)}{P(X)} = \frac{\int p(X,Y,Z)dZ}{\int p(X,\tilde{Y},Z)d\tilde{Y}dZ}$

Точечные оценки:
$\hat{Y} = \underset{Y}{\text{argmax}} p(Y|X)$ –мода
$\hat{Y} = \mathbb{E}[Y|X]$ –мат.ожидание
Интегралы возникающие при вычислении апостериорного распределения, часто невозможно вычислить $\Rightarrow$ требются приближенные методы для осуществления байесовского вывода.

- Методы Монте-Карло.
- Вариационный подход.

Вариационноый подход

$p(X, Y)$ – вероятностная модель.
$q(Y)$ – произвольное распределение.

Нижнаяя оценка обоснованности:

$$logp(X) = \underbrace{\int log\frac{p(X,Y)}{q(Y)}q(Y)dY}_{\mathcal{L}(q)} \underbrace{- \int \frac{p(Y|X)}{q(Y)}q(Y)dY}_{KL(q\|p(T|X))}$$

Допустим $p(Y|X)$ – не вычисляется, хотим найти приближение $q(Y)$ для $p(Y|X)$.

$$KL(q \parallel p(T|X)) \to \min_q \Leftrightarrow \mathcal{L}(q) \to \max_q$$

Оптимизация

$q(Y)$ – семейство факторизованных распределений, т.е.

$$q(Y) = \prod_{j=1}^{J} q_j(Y_j)$$

где $Y = \bigcup_j Y_j$.
Задача оптимизации:

$$\mathcal{L}(q) = \int log \frac{p(X,Y)}{\prod_{j=1}^{J} q_j(Y_j)} \prod_{j=1}^{J} q_j(Y_j) dY_j \to \max_{q_1,\ldots,q_J}$$

Покоординатный подъем: фиксируем $q_i(Y_i)$, и рассмотрим оптимизацию по одной компоненте.

Покоординатный подъем

$$\mathcal{L}(q) = \int logp(X,Y) \prod_{j=1}^{J} q_j(Y_j) dY_j - \sum_{j=1}^{J} \int logq_j(Y_j) q_j(Y_j) dY_j =$$

$$= \int \left(\int logp(X,Y) \prod_{j \neq i}^{J} q_j(Y_j) dY_j \right) q_i(Y_i) dY_i - \int logq_i(Y_i) q_i(Y_i) dY_i + const$$

Рассмотрим распределение:

$$r(Y_i) = \frac{1}{Z} \exp \left(\int logp(X,Y) \prod_{j \neq i}^{J} q_j(Y_j) dY_j \right)$$

Покоординатный подъем

Прологорифмируем $r(Y_i)$ и подставим в выражение для $\mathcal{L}(q)$.

$$\mathcal{L}(q) = \int log r(Y_i) q_j(Y_j) dY_j - \int log q_i(Y_i) q_i(Y_i) dY_i + const =$$

$$= \int log \frac{r_i(Y_i)}{q_i(Y_i)} q_i(Y_i) dY_i + const = -KL(q_i \parallel r) + const$$

В итоге:

$$q_i(Y_i) = \frac{\exp(\int log p(X, Y) \prod_{j \neq i}^{J} q_j(Y_j) dY_j)}{\int \exp(\int log p(X, Y) \prod_{j \neq i}^{J} q_j(Y_j) dY_j) dY_i}$$

Для исходной задачи получим:
$p(Y|X) = \int p(Y, Z|X) dZ \simeq \int q_Y(Y) q_Z(Z) dZ = q_Y(Y)$

Стохастический вариационный вывод

z – скрытые переменные, полученные из априорного распределения $p_{\theta^*}(z)$.

$\mathbf{x} = \{x_i\}_{i=1}^N$ – i.i.d выборка полученная из распределения $p_{\theta^*}(x|z)$.

$p_{\theta^*}(z) \sim p_\theta(z)$, $p_{\theta^*}(x|z) \sim p_\theta(x|z)$

$p_\theta(z)$, $p_\theta(x|z)$ – дифференцируемы по θ и z почти всюду.

θ^* и z – скрытые параметры.

Задача:

- Оценить праметр θ.
- Оценить $p_\theta(z|x)$.
- Оценить $p_\theta(x)$.

Стохастический вариационный вывод

Пусть $q_\phi(z|x)$ некоторое приближение для $p_\theta(z|x)$.
$q_\phi(z|x)$ – вероятностный энкодер, $p_\theta(x|z)$ – вероятностный декодер.

Маргинальное распределение:
$logp_\theta(\mathbf{x}) = \sum_{i=1}^{N} logp_\theta(x_i)$,

$$logp_\theta(x_i) = KL(q_\phi(z|x_i) \parallel p_\theta(z|x_i)) + \mathcal{L}(q_\phi(z|x_i), \theta)$$

$$\mathcal{L}(q_\phi(z|x_i), \theta) = -KL(q_\phi(z|x_i) \parallel p_\theta(z)) + \mathbb{E}_{q_\phi(z|x_i)}[logp_\theta(x_i|z)]$$

Задача:

$$\mathcal{L}(q_\phi(z|x_i), \theta) \to \max_{\theta,\phi}$$

Решение:

- $\nabla_\phi \mathbb{E}_{q_\phi(z)}[f(z)] = \mathbb{E}_{q_\phi(z)}[f(z)\nabla_{q_\phi(z)} logq_\phi(z)] \simeq$
  $\frac{1}{L}\sum_{\ell=1}^{L} f(z)\nabla_{q_\phi(z_\ell)} logq_\phi(z_\ell)$
- SVGB

Алгоритм SVGB

Параметризуем $\tilde{z} = g_\phi(\epsilon, x)$, $\epsilon \sim p(\epsilon)$, $g_\phi(\epsilon, x)$ – дифференцируемая функция.

$\mathbb{E}_{q_\phi(z|x_i)}[f(z)] = \mathbb{E}_{p(\epsilon)}[f(g_\phi(\epsilon, x_i))] \simeq \frac{1}{L}\sum_{\ell=1}^{L} f(g_\phi(\epsilon_\ell, x_i))$, $\epsilon_\ell \sim p(\epsilon)$

$KL(q_\phi(z|x_i) \parallel p_\theta(z))$ – может быть проинтегрирована аналитически.

Приближение для нижней оценки обоснованности:

$\tilde{\mathcal{L}}(q_\phi(z|x_i), \theta) = -KL(q_\phi(z|x_i) \parallel p_\theta(z)) + \frac{1}{L}\sum_{\ell=1}^{L}(logp_\theta(x_i|z_{i,\ell}))$

$z_{i,\ell} = g_\phi(\epsilon_{i,\ell}, x_i)$, $\epsilon_\ell \sim p(\epsilon)$

Для подвыборок размера M, $X^M = \{x_i\}_{i=1}^M$:

$\mathcal{L}(q_\phi(z|x_i), \theta) \simeq \tilde{\mathcal{L}}^M(q_\phi(z|x_i), \theta) = \frac{N}{M}\sum_{i=i}^{M} \tilde{\mathcal{L}}(q_\phi(z|x_i), \theta)$

Репараметризация $z = g_\phi(\epsilon, x)$

$g_\phi(\cdot)$ – вектор-функция, с параметром ϕ.

Пусть $z = g_\phi(\epsilon, x)$.

$q_\phi(z|x) \prod_i dz_i = p(\epsilon) \prod_i d\epsilon_i$.

$\int q_\phi(z|x) f(z) dz = \int p(\epsilon) f(z) d\epsilon = \int p(\epsilon) f(g_\phi(\epsilon, x)) d\epsilon$

$\Rightarrow \int q_\phi(z|x) f(z) dz \simeq \frac{1}{L} \sum_{\ell=1}^{L} f(g_\phi(\epsilon_\ell, x))$, $\epsilon_\ell \sim p(\epsilon)$.

Пример: $z \sim N(\mu, \sigma^2)$
Репараметризация: $z = \mu + \sigma\epsilon$, где $\epsilon \sim N(0, 1)$
Тогда $\mathbb{E}_{N(z,\mu,\sigma)}[f(z)] = \mathbb{E}_{N(0,1)}[f(\mu + \sigma\epsilon)] \simeq \frac{1}{L} \sum_{\ell=1}^{L} f(\mu + \sigma\epsilon_i)$

Задача:
Необходимо подобрать дифференцируемое преобразование $g_\phi(\cdot)$ для $q_\phi(z|x)$.

Алгоритм SVGB

Algorithm 1 Minibatch SVGB

1: θ, $\phi \leftarrow$ Выбираем начальное приближение.
2: **repeat**
3: $X^M \leftarrow$ Случайная выборка размера M из X
4: $\epsilon \leftarrow$ Случайная выборка из $p(\epsilon)$
5: $g \leftarrow \nabla_{\theta,\phi} \tilde{\mathcal{L}}^M(q_\phi(z|x_i), \theta)$
6: $\theta, \phi \leftarrow$ Обновляем параметры на основании градиента g (Adagrad)
7: **until** θ, ϕ не сойдутся
8: **return** θ, ϕ

Пояснение: выбираем $\epsilon_i \longrightarrow$ вычисляем $z_{i,\ell} = g_\theta(\epsilon_i, x_i) \longrightarrow$ вычисляем $logp_\theta(x_i|z_{i,\ell})$

Общая постановка задачи SSL

Дано:
$\mathcal{X}$ - множество объектов.
$\mathcal{Y}$ - множество классов.

$(\mathcal{X}, \mathcal{Y}) = \{(x_i, y_i)\}_{i=1}^{\ell}$ – размеченная выборка.

$(\mathcal{X}) = \{(x_i)\}_{i=\ell+1}^{\ell+u}$ – неразмеченная выборка.

Варианты постановки задач:

- Частичное обучение.
  Построить алгоритм классификации $a : \mathcal{X} \to \mathcal{Y}$.
- Трансдуктивное обучение.
  Получить метки классов $(\mathcal{Y})^u = \{(y_i)\}_{i=\ell+1}^{\ell+u}$ на неразмеченной части выборки $(\mathcal{X})^u$.

Применение:
В задачах, для которых много неразмеченных данных и при этом разметка достаточного количества данных либо слишком дорогая либо получить метки классов для всего набора данных не представляется возможным.

SSL in deep generative models(DGM).

$(\mathcal{X}, \mathcal{Y})$ – размеченная часть выборки.
$\mathcal{X}$ – неразмеченная часть выборки.
$x_i \in R^d$
$y_i \in \{1, ..., L\}$
z_i – некоторый дополнительный набор скрытых переменных.

Описание модели

- Latent-feature discriminative model (M1).
- Generative semi-supervised model (M2).
- Stacked generative semi-supervised model (M1+M2).

Описание модели

Latent-feature discriminative model (M1)
$p(z) = N(z|0, I)$, $p_\theta(x|z) = f(x; z, \theta)$
$p(z|x)$ будем использовать в качестве признаков для обучения алгоритма классификации $a : X \to Y$.

Generative semi-supervised model (M2)
$p(y) = Cat(y|\pi)$, $p(z) = N(z|o, I)$, $p_\theta(x|y, z) = f(x; y, z, \theta)$
$Cat(y|\pi)$ – мультиномиальное распределение.
Метки классов y на неразмеченной части выборки будем считать скрытыми переменными в дополнение к z.

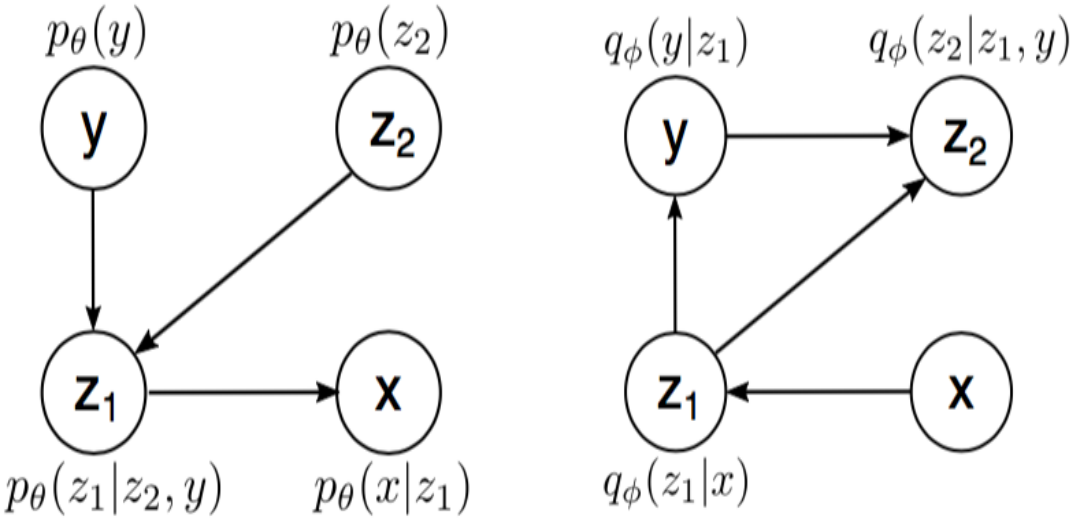
Stacked generative semi-supervised model (M1+M2)

- обучение модели M1 $\longrightarrow$ вектор параметров z_1
- $z_1 \longrightarrow$ обучение модели M2

Результат: модель: $p(x, y, z_1, z_2) = p(y)p(z_2)p_\theta(z_1|y, z_2)p_\theta(x|z_1)$

$f(\cdot)$ – некоторая нелинейная функция правдоподобия (например можно выбрать функции распределения Бернулли или Гаусса).

- **M1**: $p_\theta(x|z_1) = f(x, z_1, \theta)$
- **M2**: $p(y) = Cat(y|\pi)$,
  $p(z_2) = N(z_2|0, I), p_\theta(z_1|y, z_2) = f(z_1, y, z_2, \theta)$



Вариационный вывод для DGM

- Введем приближение $q_\phi(z|x)$ с параметром ϕ, для $p(z|x)$.
- **M1**: $q_\phi(z|x) = N(z|\mu_\phi(x), diag(\sigma^2_\phi(x)))$
- $q_\phi(z, y|x) = q_\phi(z|x)q_\phi(y|x)$
- **M2**: $q_\phi(z|y, x) = N(z|\mu_\phi(y, x), diag(\sigma^2_\phi(x, y)))$,
  $q_\phi(y|x) = Cat(y|\pi_\phi(x))$

  $\sigma_\phi(x)$ – вектор стандартных отклонений

  $\pi_\phi(x)$ – вектор вероятностей

  $\mu_\phi(x), \sigma_\phi(x), \pi_\phi(x)$ –MLP's.

Вариационный вывод для DGM

- **M1**:
  $logp_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)}[logp_\theta(x|z)] - KL(q_\phi(z|x) \parallel p_\theta(z)) = -\mathcal{J}(x)$
- **M2**:
  Для размеченной части выборки:
  $logp_\theta(x,y) \geq \mathbb{E}_{q_\phi(z|x,y)}[logp_\theta(x|y,z) + logp_\theta(y) + logp(z) - logq_\phi(z|x,y)] = -\mathcal{L}(x,y)$

  Для неразмеченной части выборки:
  $logp_\theta(x) \geq$
  $\mathbb{E}_{q_\phi(y,z|x)}[logp_\theta(x|y,z) + logp_\theta(y) + logp(z) - logq_\phi(y,z|x)] = -\mathcal{U}(x)$

  Для всего входного набора данных:
  $\mathcal{J} = \sum_{(x,y)\in(\mathcal{X},\mathcal{Y})} \mathcal{L}(x,y) + \sum_{x\in\mathcal{X}} \mathcal{U}(x)$
  Добавим потерю при классификации на размеченной части выборки:
  $\mathcal{J}^\alpha = \mathcal{J} + \alpha\, \mathbb{E}_{(\mathcal{X},\mathcal{Y})}[-logq_\phi(y|x)]$

Оптимизация модели M1

Algorithm 2 Learning in model M1

while *generativeTraining*() **do**
  $\mathcal{D} \leftarrow$ случайная подвыборка из множества объектов выборки.
  $z_i \sim q_\phi(z_i|x_i), \forall x_i \in \mathcal{D}$
  $\mathcal{J} \leftarrow \sum_n \mathcal{J}(x_i)$
  $(g_\theta, g_\phi) \leftarrow (\frac{\partial \mathcal{J}}{\partial \theta}, \frac{\partial \mathcal{J}}{\partial \phi})$
  $(\theta, \phi) \leftarrow (\theta, \phi) + \Gamma(g_\theta, g_\phi)$
while *discriminativeTraining*() **do**
  $\mathcal{D} \leftarrow$ случайная подвыборка из множества размеченных объектов выборки.
  $z_i \sim q_\theta(z_i|x_i), \forall \{x_i, y_i\} \in \mathcal{D}$
  trainClassifier$(\{z_i, y_i\})$

Оптимизация модели M2

Algorithm 3 Learning in model M2

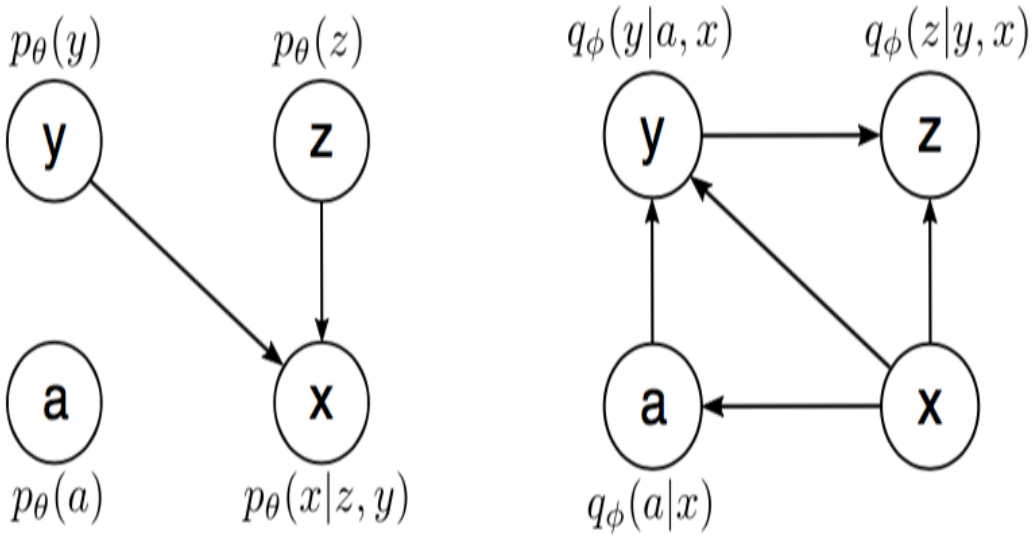
while *training*() **do**
  $\mathcal{D} \leftarrow$ случайная подвыборка из множества объектов.
  $y_i \sim q_\phi(y_i|x_i), \forall\{x_i, y_i\} \notin \mathcal{O}$
  $z_i \sim q_\phi(z_i|y_i, x_i)$
  $\mathcal{J}^\alpha \leftarrow \mathcal{J} + \alpha\, \mathbb{E}_{(\mathcal{X},\mathcal{Y})}[-log q_\phi(y|x)]$
  $(g_\theta, g_\phi) \leftarrow (\frac{\partial \mathcal{L}^\alpha}{\partial \theta}, \frac{\partial \mathcal{L}^\alpha}{\partial \phi})$
  $(\theta, \phi) \leftarrow (\theta, \phi) + \Gamma(g_\theta, g_\phi)$

$$\mathbb{E}_{q_\phi(z|x)}[log p_\theta(x|z)] = \mathbb{E}_{N(\epsilon|0,I)}[log p_\theta(x|\mu_\phi(x) + \sigma_\phi(x) \odot \epsilon)]$$

$$\nabla_{\theta,\phi}\, \mathbb{E}_{q_\phi(z|x)}[log p_\theta(x|z)] = \mathbb{E}_{N(\epsilon|0,I)}[\nabla_{\theta,\phi} log p_\theta(x|\mu_\phi(x) + \sigma_\phi(x) \odot \epsilon)]$$

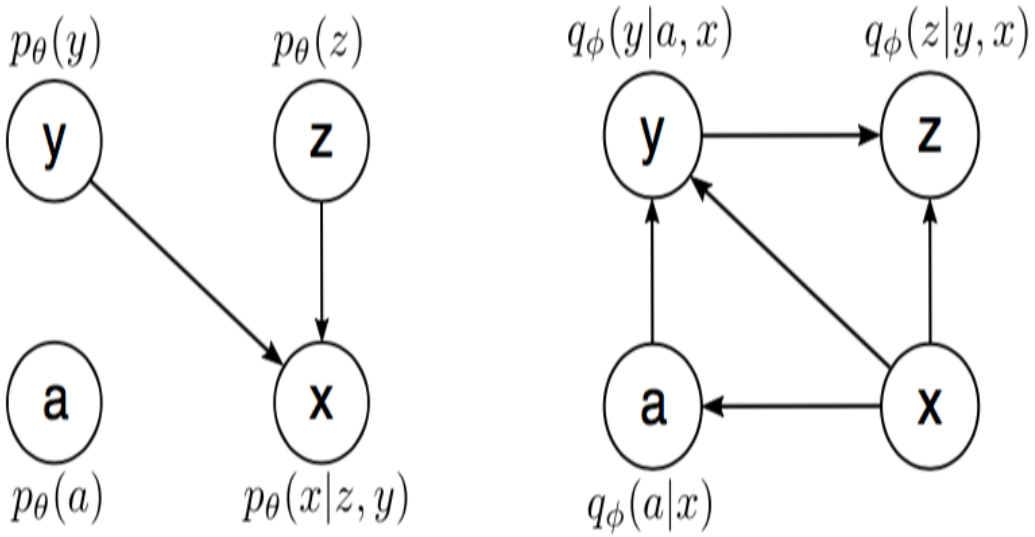
Auxiliary deep generative model(ADGM)

- $p_\theta(x|z,y) = f(x,z,y,\theta), p(z) = N(z|0,I)$
- $p(y) = Cat(y|\pi), p(a) = N(a|0,I)$
- $q_\phi(a|x) = N(a|\mu_\phi(x), diag(\sigma^2_\phi(x)))$
- $q_\phi(z|y,x) = N(z|\mu_\phi(y,x), diag(\sigma^2_\phi(y,x)))$
- $q_\phi(y|a,x) = Cat(y|\pi_\phi(a,x))$



Auxiliary deep generative model(ADGM)

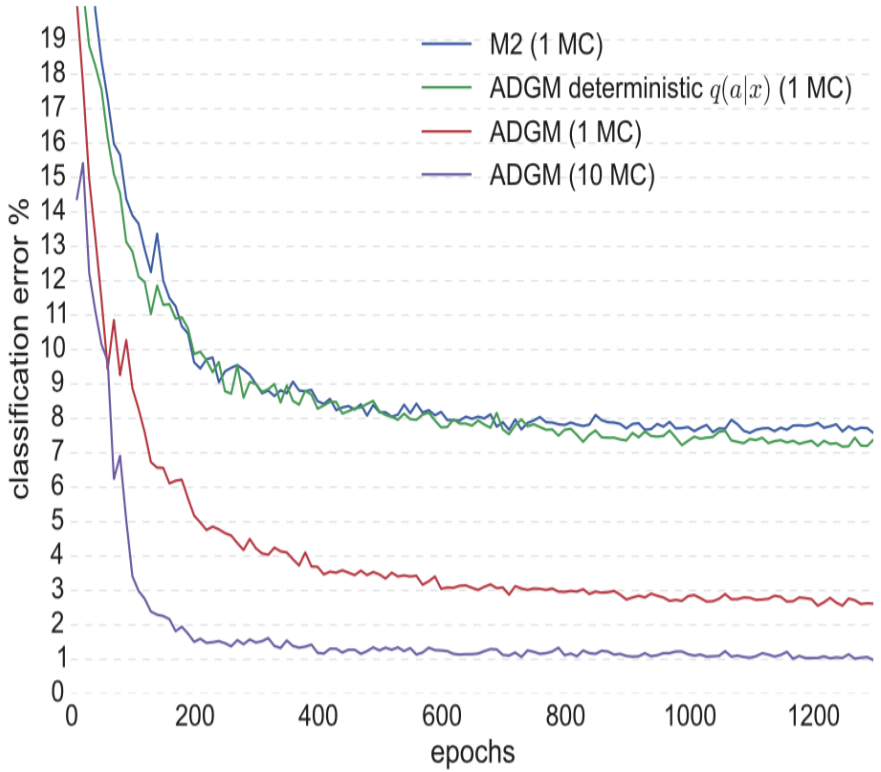
- $p_\theta(x|z,y) = f(x,z,y,\theta), p(z) = N(z|0,I)$
- $p(y) = Cat(y|\pi), p(a) = N(a|0,I)$
- $q_\phi(a|x) = N(a|\mu_\phi(x), diag(\sigma^2_\phi(x)))$
- $q_\phi(z|y,x) = N(z|\mu_\phi(y,x), diag(\sigma^2_\phi(y,x)))$
- $q_\phi(y|a,x) = Cat(y|\pi_\phi(a,x))$



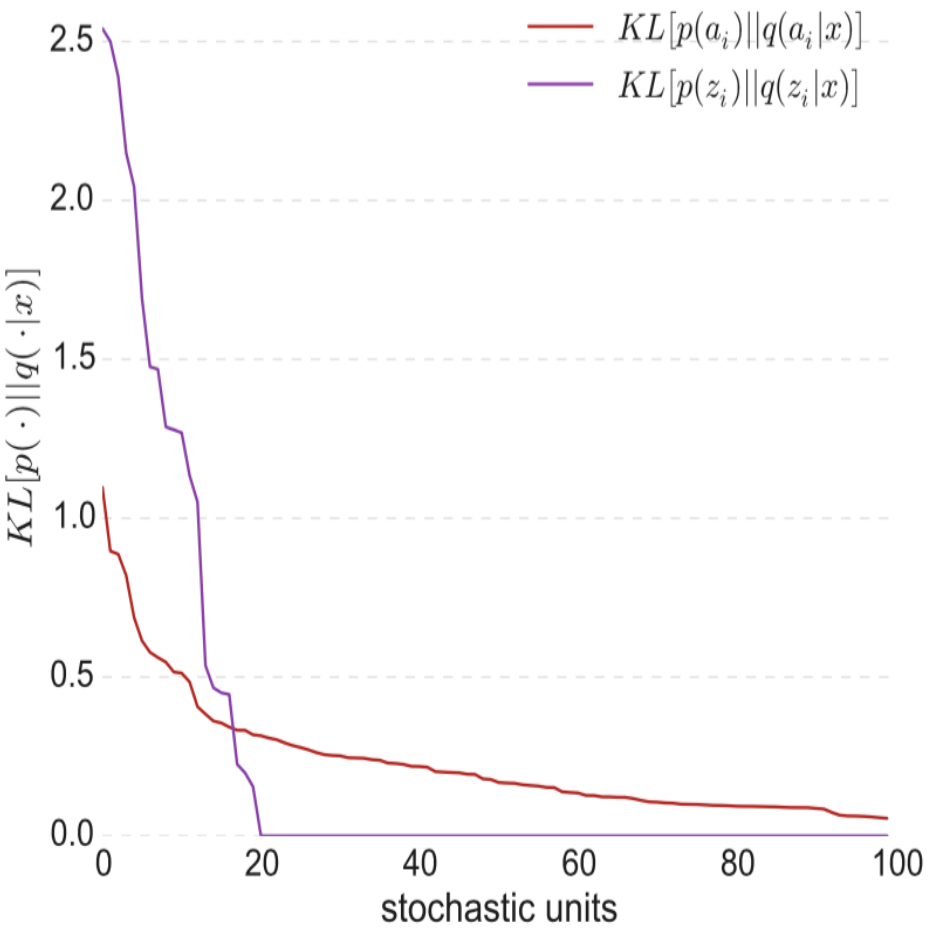
Результаты

	100 labels
AtlasRBF (Pitelis et al., 2014)	8.10% (±0.95)
Deep Generative Model (M1+M2) (Kingma et al., 2014)	3.33% (±0.14)
Virtual Adversarial (Miyato et al., 2015)	2.12%
Ladder (Rasmus et al., 2015)	1.06% (±0.37)
Auxiliary Deep Generative Model (1 MC)	2.25% (± 0.08)
Auxiliary Deep Generative Model (10 MC)	**0.96% (± 0.02)**

Результаты


M2 (1 MC)
ADGM deterministic $q(a|x)$ (1 MC)
ADGM (1 MC)
ADGM (10 MC)
classification error %
epochs
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
0
200
400
600
800
1000
1200

Результаты


$KL[p(a_i)||q(a_i|x)]$
$KL[p(z_i)||q(z_i|x)]$
2.5
2.0
1.5
1.0
0.5
0.0
$KL[p(\cdot)||q(\cdot|x)]$
0
20
40
60
80
100
stochastic units

Ссылки

- Байесовские методы машинного обучения (курс лекций) / 2016 (machinelearning.ru)
- Kingma, Diederik P; Welling, M. (2013). Auto-Encoding Variational Bayes. arXiv preprint arXiv:1312.6114.
- Kingma, D. P., Rezende, D. J., Mohamed, S., and Welling, M. (2014). Semi-Supervised Learning with Deep Generative Models. arXiv preprint arXiv:1406.5298.
- Lars Maaløe, Casper Kaae Sønderby, Søren Kaae Sønderby, Ole Winther Auxiliary Deep Generative Models arXiv:1602.05473

Спасибо за внимание.