Semi-supervised Learning with Deep Generative Models

Ольга Гребенькова

Байесовское мультимоделирование

27 октября 2021 г.

О чем пойдет речь?

Обучение с частичным привлечением учителя

Обучающий датасет содержит как размеченные, так и неразмеченные данные. Этот метод особенно полезен, когда трудно извлечь из данных важные признаки или разметить все объекты – трудоемкая задача.

Где встречается?

Очень много применений: анализ медицинских изображений, анализ речи, парсинг текста и т.д

Варианты решений

Самообучение, совместное обучение, полуавтоматические опорные вектора (S3VM), генеративные сети

1. данные
$$(\mathbf{X}, \mathbf{Y}) = (\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N),$$
где $\mathbf{x}_i \in \mathbb{R}^D$, а $y_i \in \{1, \dots, L\}$.
2. скрытая переменная и распределение вектора переменных в скрытом пространстве
$$\mathbf{z}_i$$
$$p(\mathbf{z}) = N(\mathbf{z}|0, \mathbf{I});$$
3. распределение итоговых наблюдений при условии скрытой перемнной
$$p_\theta(\mathbf{x}|\mathbf{z}) = f(\mathbf{x}; \mathbf{z}, \theta),$$
где $f(\mathbf{x}; \mathbf{z}, \theta)$ является подходящей функцией правдоподобия. Вероятности формируются нелинейным преобразованием с параметрами θ из набора скрытых переменных $\mathbf{z}$.
4. сэмплы из $p(\mathbf{z}|\mathbf{x})$ рассматриваем как признаки для обучения классификатора (SVM, многклассовая регрессия).

1. распределение вектора переменных в скрытом пространстве
$$p(\mathbf{z}) = N(\mathbf{z}|0, \mathbf{I});$$
2. $$p(y) = Cat(y|\pi),$$
где $Cat(y|\pi)$ мультиномиальное распределение.
3. распределение итоговых наблюдений при условии скрытых переменных
$$p_\theta(\mathbf{x}|y, \mathbf{z}) = f(\mathbf{x}; y, \mathbf{z}, \theta),$$
где $f(\mathbf{x}; y, \mathbf{z}, \theta)$ является подходящей функцией правдоподобия.
4. из апостериорного распределения $p(y|\mathbf{x})$ получаем классы неразмеченных данных.
5. M1+M2 сначала обучаем скрытое представление $\mathbf{z}_1$ с помощью модели 1. Далее обучаем модель 2 используя эмбеддинги из $\mathbf{z}_1$ вместо $\mathbf{x}$. Итоговая модель:
$$p_\theta(\mathbf{x}, y, \mathbf{z}_1, \mathbf{z}_2) = p(y)p(\mathbf{z}_2)p_\theta(\mathbf{z}_1|y, \mathbf{z}_2)p_\theta(\mathbf{x}|\mathbf{z}_1).$$

Масштабируемый вариационный вывод

- Введем распределение фиксированной формы аппроксимирующее неизвестное апостериорное $p(\mathbf{z}|\mathbf{x})$:
$$q_\phi(\mathbf{z}|\mathbf{x}).$$
-
$$M1 : q_\phi(\mathbf{z}|\mathbf{x}) = N(\mathbf{z}|\mu_\phi(\mathbf{x}), diag(\sigma^2_\phi(\mathbf{x})))$$
$$M2 : q_\phi(\mathbf{z}|y, \mathbf{x}) = N(\mathbf{z}|\mu_\phi(y, \mathbf{x}), diag(\sigma^2_\phi(\mathbf{x}))) \quad q_\phi(y|\mathbf{x}) = Cat(y|\pi_\phi(x))$$
При этом:
$$q_\phi(\mathbf{z}, y|\mathbf{x}) = q_\phi(\mathbf{z}|\mathbf{x})q_\phi(y|\mathbf{x})$$
- Будем расматривать в качестве лосса вариационную нижнюю границу правдоподобия модели. Для первой модели она выглядит следующим образом:
$$\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - KL[q_\phi(\mathbf{z}|\mathbf{x})||p_\theta(\mathbf{z})] = -J(\mathbf{x})$$

- Если мы знаем класс объекта:

$$\log p_\theta(\mathbf{x}, y) \geq \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x},y)}[\log p_\theta(\mathbf{x}|y,\mathbf{z}) + \log p_\theta(y) + \log p(\mathbf{z}) - \log q_\phi(\mathbf{z}|\mathbf{x},y)] = -L(\mathbf{x}, y)$$

- Если класс объекта неизвестен

$$\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\phi(y,\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|y,\mathbf{z}) + \log p_\theta(y) + \log p(\mathbf{z}) - \log q_\phi(y,\mathbf{z}|\mathbf{x})] =$$

$$= \sum_y q_\phi(y|\mathbf{x})(-L(\mathbf{x}, y)) + H(q_\phi(y|\mathbf{x})) = -U(\mathbf{x}).$$

- Тогда функция для всего датасета:

$$J = \sum_{(\mathbf{x},y)\sim p_l} L(\mathbf{x}, y) + \sum_{\mathbf{x}\sim p_u} U(\mathbf{x})$$

- Расширенная лосс функция:

$$J_\alpha = J + \alpha \mathbb{E}_{p_l(\mathbf{x},y)}[-\log q_\phi(y|\mathbf{x})].$$

Algorithm 1 Learning in model M1

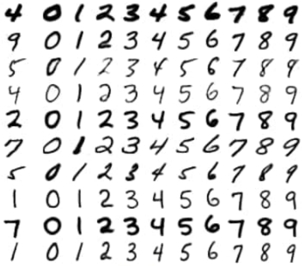
while generativeTraining() **do**
  $\mathcal{D} \leftarrow$ getRandomMiniBatch()
  $\mathbf{z}_i \sim q_\phi(\mathbf{z}_i|\mathbf{x}_i) \quad \forall \mathbf{x}_i \in \mathcal{D}$
  $\mathcal{J} \leftarrow \sum_n \mathcal{J}(\mathbf{x}_i)$
  $(\mathbf{g}_\theta, \mathbf{g}_\phi) \leftarrow (\frac{\partial \mathcal{J}}{\partial \theta}, \frac{\partial \mathcal{J}}{\partial \phi})$
  $(\boldsymbol{\theta}, \boldsymbol{\phi}) \leftarrow (\boldsymbol{\theta}, \boldsymbol{\phi}) + \mathbf{\Gamma}(\mathbf{g}_\theta, \mathbf{g}_\phi)$
end while
while discriminativeTraining() **do**
  $\mathcal{D} \leftarrow$ getLabeledRandomMiniBatch()
  $\mathbf{z}_i \sim q_\phi(\mathbf{z}_i|\mathbf{x}_i) \quad \forall \{\mathbf{x}_i, y_i\} \in \mathcal{D}$
  trainClassifier($\{\mathbf{z}_i, y_i\}$)
end while

Algorithm 2 Learning in model M2

while training() **do**
  $\mathcal{D} \leftarrow$ getRandomMiniBatch()
  $y_i \sim q_\phi(y_i|\mathbf{x}_i) \quad \forall \{\mathbf{x}_i, y_i\} \notin \mathcal{O}$
  $\mathbf{z}_i \sim q_\phi(\mathbf{z}_i|y_i, \mathbf{x}_i)$
  $\mathcal{J}^\alpha \leftarrow$ eq. (9)
  $(\mathbf{g}_\theta, \mathbf{g}_\phi) \leftarrow (\frac{\partial \mathcal{L}^\alpha}{\partial \theta}, \frac{\partial \mathcal{L}^\alpha}{\partial \phi})$
  $(\boldsymbol{\theta}, \boldsymbol{\phi}) \leftarrow (\boldsymbol{\theta}, \boldsymbol{\phi}) + \mathbf{\Gamma}(\mathbf{g}_\theta, \mathbf{g}_\phi)$
end while

Результаты

(a) Handwriting styles for MNIST obtained by fixing the class label and varying the 2D latent variable **z**



(b) MNIST analogies

(c) SVHN analogies