Байесовы нейронные сети

Басов Дмитрий Константинович

1 Аннотация

$N(\mu, \sigma^2)$ — нормальное распределение

$\mathcal{L}$ — Evidence Lower Bound (ELBO)

$KL(q||p) = \int q(\mathbf{Z}) \cdot \ln \frac{q(\mathbf{Z})}{p(\mathbf{Z})} d\mathbf{Z}$ — расстояние Кульбака — Лейблера

$\mathbf{x}$ — вектор признаков

$\mathbf{y}$ — таргет

D — датасет — пары значений $\{\mathbf{x_i}, \mathbf{y_i}\}$, где $i = 1, \ldots, L$

$\mathbf{W}$ — параметры модели — случайная величина размерности M

$p(D|\mathbf{W}) = \prod_{i=1}^{L} p(\mathbf{y_i}|\mathbf{x_i}, \mathbf{W})$ — правдоподобие (likelihood)

$p(\mathbf{W})$ — априорное распределение параметров модели (prior)

$p(\mathbf{W}|D)$ — апостериорное распределение параметров модели (posterior)

$p(D)$ — маргинальная вероятность датасета (evidence)

$q(\mathbf{W})$ — аппроксимация апостериорного распределения параметров модели

$p(\mathbf{W}, D) = p(D|\mathbf{W}) \cdot p(\mathbf{W}) = p(\mathbf{W}|D) \cdot p(D)$ — совместная вероятность параметров и данных

2 Постановка задачи

Постановка задачи следующая — у нас есть датасет D и наша цель — смоделировать распределение $p(\mathbf{y}|\mathbf{x}, D)$. То есть мы хотим получить распределение вероятностей таргета $\mathbf{y}$ для неразмеченных $\mathbf{x}$ используя датасет D. Сделаем следующие преобразования:

$$p(\mathbf{y}|\mathbf{x}, D) = \int p(\mathbf{y}, \mathbf{W}|\mathbf{x}, D) d\mathbf{W} = \int p(\mathbf{y}|\mathbf{W}, \mathbf{x}, D) \cdot p(\mathbf{W}|\mathbf{x}, D) d\mathbf{W} = \int p(\mathbf{y}|\mathbf{W}, \mathbf{x}) \cdot p(\mathbf{W}|D) d\mathbf{W}$$

Получим выражение для $p(\mathbf{W}|D)$, используя формулу Байеса:

$$p(\mathbf{W}|D) = \frac{p(\mathbf{W}, D)}{p(D)} = \frac{p(\mathbf{W}, D)}{\int p(\mathbf{W}, D) d\mathbf{W}} = \frac{p(D|\mathbf{W}) \cdot p(\mathbf{W})}{\int p(D|\mathbf{W}) \cdot p(\mathbf{W}) d\mathbf{W}}$$

Для аппроксимации распределения ответов модели можно воспользоваться методом Монте — Карло — взять сэмпл весов $\hat{\mathbf{W}}$ из $p(\mathbf{W}|D)$, прогнать их через модель и получить $\hat{\mathbf{y}}$. Однако для этого необходимо уметь сэмплировать из распределения $p(\mathbf{W}|D)$.

Получить аналитическое решение можно только в очень ограниченном числе случаев. Существует возможность семплировать из $p(\mathbf{W}|D)$, используя методы Монте — Карло для марковских цепей (MCMC). Однако для больших датасетов и большого числа параметров это становится технически сложно. Альтернативный подход к решению таких задач — аппроксимация распределения $p(\mathbf{W}|D)$ распределением $q(\mathbf{W})$, из которого сэмплировать намного проще.

3 Вариационный вывод для нейронной сети

Запишем выражение ELBO для распределения $q(\mathbf{W})$ и преобразуем его используя тождество $p(\mathbf{W}, D) = p(\mathbf{W}|D) \cdot p(D)$:

$$\mathcal{L}(q(\mathbf{W})) = \int q(\mathbf{W}) \cdot \ln \frac{p(\mathbf{D}, \mathbf{W})}{q(\mathbf{W})} d\mathbf{W} = \int q(\mathbf{W}) \cdot \ln \frac{p(\mathbf{W}|D) \cdot p(D)}{q(\mathbf{W})} d\mathbf{W} = \ln p(D) \cdot \int q(\mathbf{W}) d\mathbf{W} -$$
$$\int q(\mathbf{W}) \cdot \ln \frac{q(\mathbf{W})}{p(\mathbf{W}|D)} d\mathbf{W} = \ln p(D) - KL(q(\mathbf{W})||p(\mathbf{W}|D))$$

Из равенства $\mathcal{L}(q(\mathbf{W})) = ln(p(D)) - KL(q(\mathbf{W})||p(\mathbf{W}|D))$ видно, что максимизируя $\mathcal{L}(q(\mathbf{W}))$, мы не только максимизируем $\ln p(D)$, но и минимизируем $KL(q(\mathbf{W})||p(\mathbf{W}|D))$. То есть распределение $q(\mathbf{W})$ будет приближаться к распределению $p(\mathbf{W}|D)$.

Будем максимизировать $\mathcal{L}(q(\mathbf{W}))$. Преобразуем выражение для $\mathcal{L}(q(\mathbf{W}))$, используя тождество $p(\mathbf{W}, D) = p(D|\mathbf{W}) \cdot p(\mathbf{W})$:

$$\mathcal{L}(q(\mathbf{W})) = \int q(\mathbf{W}) \cdot \ln \frac{p(\mathbf{D}, \mathbf{W})}{q(\mathbf{W})} d\mathbf{W} = \int q(\mathbf{W}) \cdot \ln \frac{p(D|\mathbf{W}) \cdot p(\mathbf{W})}{q(\mathbf{W})} d\mathbf{W} = \int q(\mathbf{W}) \cdot \ln p(D|\mathbf{W}) d\mathbf{W} -$$
$$\int q(\mathbf{W}) \cdot \ln \frac{q(\mathbf{W})}{p(\mathbf{W})} = \int q(\mathbf{W}) \cdot \ln p(D|\mathbf{W}) d\mathbf{W} - KL(q(\mathbf{W})||p(\mathbf{W}))$$

Для дальнейшнего вывода положим, что распределения $p(\mathbf{W})$ и $q(\mathbf{W})$ являются нормальными с диагональными матрицами ковариации:

$p(\mathbf{W}) = N(\mathbf{W}|\mathbf{0}, \boldsymbol{\sigma_{p(W)}}^2 \cdot \mathbf{I})$, где $\boldsymbol{\sigma_{p(W)}}$ — вектор длины M

$q(\mathbf{W}) = N(\mathbf{W}|\boldsymbol{\theta}, \boldsymbol{\sigma_{q(W)}}^2 \cdot \mathbf{I})$, где $\boldsymbol{\theta}$ и $\boldsymbol{\sigma_{q(W)}}$ — вектора длины M

Так как распределения $p(\mathbf{W})$ и $q(\mathbf{W})$ являются нормальными, то $KL(q(\mathbf{W})||p(\mathbf{W}))$ можно посчитать аналитически:

$$KL(q(\mathbf{W})||p(\mathbf{W})) = \frac{1}{2} \sum_{k=1}^{M} (\frac{\sigma^2_{q(W)_k}}{\sigma^2_{p(W)_k}} + \frac{\theta_k^2}{\sigma^2_{p(W)_k}} - \ln \frac{\sigma^2_{q(W)_k}}{\sigma^2_{p(W)_k}} - 1)$$

Априорное распределение параметров модели определяется параметром $\boldsymbol{\sigma_{p(W)}}$. Воспользуемся техникой эмперического Байеса — нахождения параметров априорного распределения из данных. Посчитаем $\frac{d\mathcal{L}(q(\mathbf{W}))}{d(\boldsymbol{\sigma}^{-2}_{\boldsymbol{p(W)}})}$:

$$\frac{d\mathcal{L}(q(\mathbf{W}))}{d(\boldsymbol{\sigma}^{-2}_{\boldsymbol{p(W)}})} = \frac{d(\int q(\mathbf{W}) \cdot \ln p(D|\mathbf{W}) d\mathbf{W} - KL(q(\mathbf{W})||p(\mathbf{W})))}{d(\boldsymbol{\sigma}^{-2}_{\boldsymbol{p(W)}})} = -\frac{d(KL(q(\mathbf{W})||p(\mathbf{W})))}{d(\boldsymbol{\sigma}^{-2}_{\boldsymbol{p(W)}})}$$

$$\frac{d\mathcal{L}(q(\mathbf{W}))}{d(\boldsymbol{\sigma}^{-2}_{\boldsymbol{p(W)}})} = -\frac{1}{2} \sum_{k=1}^{M} (\sigma^2_{q(W)_k} + \theta_k^2 - \sigma^2_{p(W)_k})$$

Приравняв производную к нулю, получим:

$\boldsymbol{\sigma^2_{p(W)}} = \boldsymbol{\theta^2} + \boldsymbol{\sigma^2_{q(W)}}$

Подставив полученное выражение в $KL(q(\mathbf{W})||p(\mathbf{W}))$, получим:

$$KL(q(\mathbf{W})||p(\mathbf{W})) = \frac{1}{2} \sum_{k=1}^{M} \ln(1 + \frac{\theta_k^2}{\sigma^2_{q(W)_k}})$$

Чтобы избежать неопределенности $\frac{0}{0}$ и переписать выражение в векторном виде, сделаем следующую замену переменных:

$\boldsymbol{\theta} = \boldsymbol{\gamma} \cdot \boldsymbol{\sigma_{q(W)}}$

$\boldsymbol{\nu} = \ln(1 + \boldsymbol{\gamma}^2)$

Так как обучение модели будет производится с помощью градиентных методов, сделаем следующую замену переменных, чтобы $\boldsymbol{\sigma_{q(W)}}$ была всегда положительна:

$\boldsymbol{\sigma_{q(W)}} = \ln(1 + \exp(\boldsymbol{\rho})) = Softplus(\boldsymbol{\rho})$

Таким образом, функция потерь будет иметь следующий вид:

$Loss(\boldsymbol{\rho}, \boldsymbol{\gamma}) = -\frac{\mathcal{L}(q(\mathbf{W}))}{L} = \int N(\mathbf{W}|\boldsymbol{\theta}, \boldsymbol{\sigma_{q(\mathbf{W})}}^2 \cdot \mathbf{I}) \sum_{i=1}^{L} \frac{-\ln p(\mathbf{y_i}|\mathbf{x_i}, \mathbf{W})}{L} d\mathbf{W} + \frac{\boldsymbol{\nu}^T \boldsymbol{\nu}}{2L}$, где:

$\boldsymbol{\nu} = \ln(1 + \boldsymbol{\gamma}^2)$

$\boldsymbol{\theta} = \boldsymbol{\gamma} \cdot \boldsymbol{\sigma_{q(\mathbf{W})}}$

$\boldsymbol{\sigma_{q(\mathbf{W})}} = Softplus(\boldsymbol{\rho})$

4 Алгоритм обучения

Задаем шаг градиентного спуска α и инициализируем параметры распределения $q(\mathbf{W})$ — $\boldsymbol{\rho} \leftarrow \mathbf{1}$ и $\boldsymbol{\gamma} \leftarrow \mathbf{0}$. Затем повторяем, пока не достигнем критерия остановки:

1. $\boldsymbol{\sigma} \leftarrow Softplus(\boldsymbol{\rho})$
2. $\boldsymbol{\theta} \leftarrow \boldsymbol{\gamma} \cdot \boldsymbol{\sigma}$
3. $\boldsymbol{\nu} \leftarrow \ln(1 + \boldsymbol{\gamma}^2)$
4. $\hat{\mathbf{W}} \leftarrow N(0, 1)$ — сэмплируем случайные веса
5. $\hat{\mathbf{W}} \leftarrow \hat{\mathbf{W}} \cdot \boldsymbol{\sigma} + \boldsymbol{\theta}$ — репараметризация
6. $l \leftarrow \frac{\boldsymbol{\nu}^T \boldsymbol{\nu}}{2L} - \sum_{i=1}^{L} \frac{\ln p(\mathbf{y_i}|\mathbf{x_i}, \hat{\mathbf{W}})}{L}$ — считаем функцию потерь
7. $\boldsymbol{\rho} \leftarrow \boldsymbol{\rho} - \alpha \frac{dl}{d\boldsymbol{\rho}}$
8. $\boldsymbol{\gamma} \leftarrow \boldsymbol{\gamma} - \alpha \frac{dl}{d\boldsymbol{\gamma}}$

Если моя задумка верна, то лишние веса модели должны выпилиться, то есть соответсвующие им θ и σ должны занулиться.