

Вычисление оператора Лапласа для многослойного перцептрона

11 октября 2022 г.

1 Математическое описание многослойного перцептрона

Пусть на вход перцептрону подаётся вектор $\mathbf{x} \in \mathbb{R}^n$, он последовательно линейно преобразуется каждым слоем и в результате на выходе перцептрона имеется вектор $\mathbf{y} \in \mathbb{R}^m$. Пусть l — число внутренних слоёв нейронной сети, а на k -ом слое имеется n_k нейронов ($n_0 = n$, $n_{l+1} = m$). Через $\mathbf{h}^{(k)} \in \mathbb{R}^{n_k}$ будет обозначаться вектор на выходе с k -ого слоя. Используя введённые обозначения можно связать \mathbf{x} и \mathbf{y} следующим образом:

$$\begin{aligned} h_i^{(0)} &= x_i, \quad i = \overline{1, n_0}; \\ h_i^{(1)} &= f^{(1)} \left(\sum_{j=1}^{n_0} W_{i,j}^{(1)} h_j^{(0)} + b_i^{(1)} \right), \quad i = \overline{1, n_1}; \\ &\dots \\ h_i^{(k)} &= f^{(k)} \left(\sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} h_j^{(k-1)} + b_i^{(k)} \right), \quad i = \overline{1, n_k}; \\ &\dots \\ h_i^{(l)} &= f^{(l)} \left(\sum_{j=1}^{n_{l-1}} W_{i,j}^{(l)} h_j^{(l-1)} + b_i^{(l)} \right), \quad i = \overline{1, n_l}; \\ y_i = h_i^{(l+1)} &= f^{(l+1)} \left(\sum_{j=1}^{n_l} W_{i,j}^{(l+1)} h_j^{(l)} + b_i^{(l+1)} \right), \quad i = \overline{1, n_{l+1}}. \end{aligned} \tag{1}$$

В предыдущей записи функция $f^{(k)}$ называется функцией активации k -го слоя, матрица $\hat{W}^{(k)}$ — матрицей весов размера $n_k \times n_{k-1}$ (первый индекс — номер строки, второй — номер столбца), а вектор $b^{(k)}$ — вектором сдвижки.

2 Первая производная многослойного перцептрона

Продифференцируем перцептрон по x_t , $t = \overline{1, n_0}$. Из (1) и правил взятия сложной производной следует такая схема расчёта

$$\begin{aligned}\frac{\partial y_i}{\partial x_t} &= \frac{\partial h_i^{(l+1)}}{\partial x_t}, \quad i = \overline{1, n_{l+1}}; \\ \frac{\partial h_i^{(k)}}{\partial x_t} &= f^{(k)'} \left(\sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} h_j^{(k-1)} + b_i^{(k)} \right) \cdot \sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} \frac{\partial h_j^{(k-1)}}{\partial x_t}, \quad i = \overline{1, n_k}, \quad k = \overline{1, l+1}; \\ \frac{\partial h_i^{(0)}}{\partial x_t} &= \delta_{i,t}, \quad i = \overline{1, n_0}.\end{aligned}\tag{2}$$

Можно заметить, что для вычисления производной разумно идти от нижних слоёв к верхним, так как вычисление каждой последующей $\partial h_i^{(k)} / \partial x_t$ зависит от значения таких производных на предыдущих слоях.

3 Лапласиан многослойного перцептрона

Основываясь на предыдущих результатах и полагая, что вектор \mathbf{x} дан в n -мерной декартовой системе координат, нетрудно получить схему вычисления лапласиана перцептрона, которая имеет следующий вид:

$$\nabla^2 y_i = \sum_{t=1}^{n_0} \frac{\partial^2 y_i}{\partial x_t^2} = \sum_{t=1}^{n_0} \frac{\partial^2 h_i^{(l+1)}}{\partial x_t^2}, \quad i = \overline{1, n_{l+1}};\tag{3}$$

$$\begin{aligned}\frac{\partial^2 h_i^{(k)}}{\partial x_t^2} &= f^{(k)''} \left(\sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} h_j^{(k-1)} + b_i^{(k)} \right) \cdot \left\{ \sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} \frac{\partial h_j^{(k-1)}}{\partial x_t} \right\}^2 \\ &+ f^{(k)'} \left(\sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} h_j^{(k-1)} + b_i^{(k)} \right) \cdot \sum_{j=1}^{n_{k-1}} W_{i,j}^{(k)} \frac{\partial^2 h_j^{(k-1)}}{\partial x_t^2}, \quad i = \overline{1, n_k}, \quad k = \overline{1, l+1};\end{aligned}\tag{4}$$

$$\frac{\partial^2 h_i^{(0)}}{\partial x_t^2} = 0, \quad i = \overline{1, n_0}.\tag{5}$$

При вычислении лапласиана, как и при вычислении градиента, следует начинать с нижних слоёв.

4 Программная реализация и верификация

Алгоритм вычисления градиента и лапласиана для многослойного перцептрона был реализован на языке Python с использованием библиотеки PyTorch. Вариант реализации и его верификация, описанная ниже, доступны по ссылке.

Для оценки верности работы алгоритмов был рассмотрен пример перцептрона с одним внутренним слоем, содержащим n нейронов. На вход такой нейронной сети подавался вектор $\mathbf{x} \in \mathbb{R}^n$, что позволило выбрать веса следующими

$$\hat{W}^{(1)} = \hat{I}, \quad \hat{W}_i^{(2)} = 1 \quad \forall i.\tag{6}$$

Тогда нетрудно получить выражение результата работы перцептрона по входному вектору

$$y(\mathbf{x}) = \tanh \left(\sum_{k=1}^n \tanh(x_k) \right). \quad (7)$$

Отсюда можно получить выражение для градиента перцептрона

$$\nabla y(\mathbf{x}) = \sum_{j=1}^n \cosh^{-2} \left(\sum_{k=1}^n \tanh(x_k) \right) \cosh^{-2}(x_j) \mathbf{e}_j, \quad (8)$$

где вектора $\{\mathbf{e}_j\}_{j=1}^n$ образуют стандартный базис в пространстве \mathbb{R}^n . Выражение для лапласиана функции $y(\mathbf{x})$ удаётся записать, используя предыдущие производные

$$\nabla^2 y(\mathbf{x}) = \sum_{i=1}^n \frac{\partial y}{\partial x_i}(\mathbf{x}) \left(\tanh(x_i) + \frac{y(\mathbf{x})}{\cosh^2(x_i)} \right). \quad (9)$$

Результаты тестирования доступны по ссылке. Из них можно заключить, что значение нейронной сети, её градиента и лапласиана вычисляются верно.