Методы оптимизации. Семинар 3. Autodiff.

Корнилов Никита Максимович

МФТИ ФИВТ

16 сентября 2024г

Example (Логистическая регрессия)

Найдите первый и второй дифференциал $df(x), d^2f(x)$, а также градиент $\nabla f(x)$ и гессиан $\nabla^2 f(x)$ функции

$$f(x) = \ln(1 + \exp(\langle a, x \rangle)),$$

где $a \in \mathbb{R}^n$.

Example (Логарифм определителя)

Найдите первый и второй дифференциалы $df(X)$ и $d^2f(X)$, а также градиент $\nabla f(X)$ функции $f(X)$

$$f(X) = \ln(\det(X))$$

заданной на множестве $X \in \mathbb{S}^n_{++}$ в пространстве $\mathbb{S}^n$.

Граф вычислений

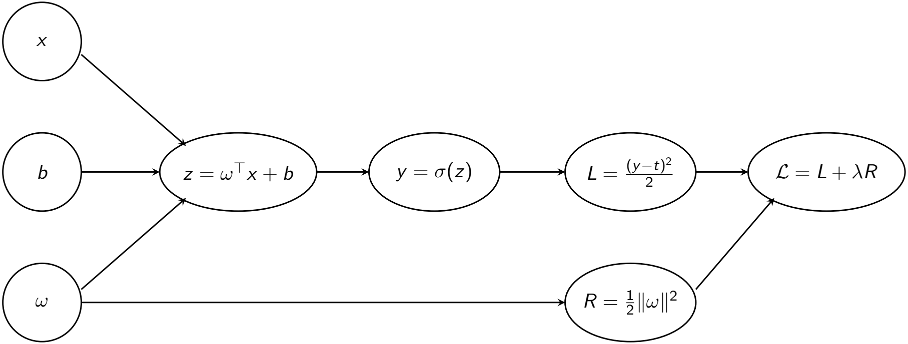
Функции представляют собой последовательность (дифференцируемых) параметрических преобразований.
Строим вычислительный граф (computational graph), где промежуточным вершинам соответствуют преобразования, входящим стрелкам – входные переменные, а выходным стрелкам – результат преобразования.

Граф вычислений

$$f(x, \omega, b) = \frac{(\sigma(\omega^\top x + b) - t)^2}{2} + \lambda \cdot \frac{1}{2}\|\omega\|_2^2, \quad x, \omega \in \mathbb{R}^n, b \in \mathbb{R}.$$

Граф вычислений

$$f(x, \omega, b) = \frac{(\sigma(\omega^\top x + b) - t)^2}{2} + \lambda \cdot \frac{1}{2}\|\omega\|_2^2, \quad x, \omega \in \mathbb{R}^n, b \in \mathbb{R}.$$



Подсчёт значения $f(x)$ по графу называется forward pass.

Производные в графе

Мы хотим найти градиент по переменной $x \in \mathbb{R}^n$ финальной функции $f(x) : \mathbb{R}^n \to \mathbb{R}$.

Back Propagation

Идти от детей к родителям и в вершине u считать $\frac{\partial f}{\partial u}$.
Для вершины u с детьми $v_1, \ldots, v_d$ мы имеем $f(u(x)) = f(v_1(u(x)), \ldots, v_d(u(x)))$ и по правилу дифф функции нескольких переменных:

$$\frac{\partial f}{\partial u}(x) = \sum_{j=1}^{d} \frac{\partial f}{\partial v_j}(x) \cdot \frac{\partial v_j}{\partial u}(x), \quad \frac{\partial f}{\partial v_j}(x) = (\nabla_{v_j} f(x))^\top.$$

В конце мы доходим до начальной вершины x и получаем $\frac{\partial f}{\partial x}$.

Back Propagation

Отсортируем все m вершин от детей к родителям и перенумеруем их от 1 до m.
Обозначим производную функции f по вершине u_i как

$$\overline{u_i} = \frac{\partial f}{\partial u_i}.$$

Общий алгоритм действий выглядит так

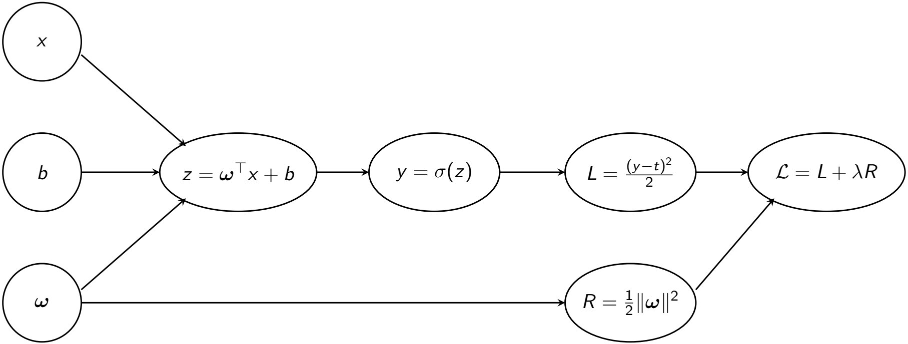
1. Произвести forward pass и сохранить все значения u_i как функции от их родителей.
2. Положить $\overline{u_m} = 1$ и для всех $i = m-1, \ldots, 1$ посчитать

$$\overline{u_i} = \sum_{j \in \text{дети}(u_i)} \overline{u_j} \frac{\partial u_j}{\partial u_i}.$$

Пример

$$f(u) = f(v_1(u), \dots, v_d(u))$$

$$\frac{\partial f}{\partial u} = \sum_{j=1}^{d} \frac{\partial f}{\partial v_j} \cdot \frac{\partial v_j}{\partial u}.$$



Обсуждение Back Propagation

$$f(u) = f(v_1(u), \ldots, v_d(u))$$

$$\frac{\partial f}{\partial u} = \sum_{j=1}^{d} \frac{\partial f}{\partial v_j} \cdot \frac{\partial v_j}{\partial u}.$$

1. Необходимо хранить ВСЕ промежуточные значения u_i.

Обсуждение Back Propagation

$$f(u) = f(v_1(u), \ldots, v_d(u))$$

$$\frac{\partial f}{\partial u} = \sum_{j=1}^{d} \frac{\partial f}{\partial v_j} \cdot \frac{\partial v_j}{\partial u}.$$

1. Необходимо хранить ВСЕ промежуточные значения u_i.
2. Важно уметь быстро превращать градиент по выходу в градиент по входу (умножать якобиан $\frac{\partial v_j}{\partial u}$ на строку слева).
3. Операции с якобианами эффективно разработаны в рамках библиотек автоматического дифференцирования.

Обсуждение Back Propagation

$$f(u) = f(v_1(u), \ldots, v_d(u))$$

$$\frac{\partial f}{\partial u} = \sum_{j=1}^{d} \frac{\partial f}{\partial v_j} \cdot \frac{\partial v_j}{\partial u}.$$

1. Необходимо хранить ВСЕ промежуточные значения u_i.
2. Важно уметь быстро превращать градиент по выходу в градиент по входу (умножать якобиан $\frac{\partial v_j}{\partial u}$ на строку слева).
3. Операции с якобианами эффективно разработаны в рамках библиотек автоматического дифференцирования.
4. Каждая вершина может быть запрограммирована как отдельная сущность, умеющая внутри себя делать forward и backward pass.

Обсуждение Back Propagation

$$f(u) = f(v_1(u), \ldots, v_d(u))$$

$$\frac{\partial f}{\partial u} = \sum_{j=1}^{d} \frac{\partial f}{\partial v_j} \cdot \frac{\partial v_j}{\partial u}.$$

1. Необходимо хранить ВСЕ промежуточные значения u_i.
2. Важно уметь быстро превращать градиент по выходу в градиент по входу (умножать якобиан $\frac{\partial v_j}{\partial u}$ на строку слева).
3. Операции с якобианами эффективно разработаны в рамках библиотек автоматического дифференцирования.
4. Каждая вершина может быть запрограммирована как отдельная сущность, умеющая внутри себя делать forward и backward pass.
5. Можно строить граф вычисления производной.

Умножение якобиана на строку

Для функции вида $f : \mathbb{R}^n \to \mathbb{R}^m$ и вектора $y \in \mathbb{R}^m$ считаем значение $y^\top J_f(x)$ в точке x, а именно

$$d\langle y, f(x)\rangle = \langle y, J_f dx\rangle = \langle J_f^\top y, dx\rangle = \langle \nabla g(x), dx\rangle,$$

где $g(x) = \langle f(x), y\rangle$.
Вместо полного гессиана можно найти градиент функции g. Именно поэтому вычисления из Back Propagation можно эффективно реализовать на практике.

Умножение гессиана на вектор

Дважды непрерывно дифференцируемая функция $f : \mathbb{R}^n \to \mathbb{R}$, вектор $y \in \mathbb{R}^n$.
Покажем, как можно эффективно считать

$$\nabla^2 f(x) \cdot y$$

$$d(\langle \nabla f(x), y \rangle) = d(\nabla f)^\top y = dx^\top \nabla^2 f(x) \cdot y = dx^\top \nabla g(x),$$

где $g(x) = \langle \nabla f(x), y \rangle$.
Вместо полного гессиана можно найти градиент функции g.

Forward Propagation

Идти от родителей к детям и в вершине u считать $\frac{\partial u}{\partial x}$.
Для вершины u с родителями $p_1, \ldots, p_k$ мы имеем $u(x) = u(p_1(x), \ldots, p_k(x))$ и по правилу дифф функции нескольких переменных:

$$\frac{\partial u}{\partial x}(x) = \sum_{j=1}^{k} \frac{\partial u}{\partial p_j}(x) \cdot \frac{\partial p_j}{\partial x}(x).$$

В конце мы доходим до конечной вершины f и получаем $\frac{\partial f}{\partial x}$.

Обсуждение Forward Propagation

$$u = u(p_1, \ldots, p_k)$$
$$\frac{\partial u}{\partial x} = \sum_{j=1}^{k} \frac{\partial u}{\partial p_j} \cdot \frac{\partial p_j}{\partial x}.$$

- Можно совершать forward pass вместе с подсчётом $\frac{\partial u}{\partial x}$ и нужно будет хранить только один слой графа.
- В forward propagation нужно хранить $\frac{\partial u_i}{\partial x}$, а в backward propagation $\frac{\partial f}{\partial u_i}$. Если размерность входа x намного больше, чем размерность выхода $f(x)$, то на каждом шаге forward pass нужно будет хранить настолько же больше данных.

Обсуждение Forward Propagation

$$u = u(p_1, \dots, p_k)$$
$$\frac{\partial u}{\partial x} = \sum_{j=1}^{k} \frac{\partial u}{\partial p_j} \cdot \frac{\partial p_j}{\partial x}.$$

- Можно совершать forward pass вместе с подсчётом $\frac{\partial u}{\partial x}$ и нужно будет хранить только один слой графа.
- В forward propagation нужно хранить $\frac{\partial u_i}{\partial x}$, а в backward propagation $\frac{\partial f}{\partial u_i}$. Если размерность входа x намного больше, чем размерность выхода $f(x)$, то на каждом шаге forward pass нужно будет хранить настолько же больше данных.

Вывод

Для $f : \mathbb{R}^n \to \mathbb{R}^m$ выгоднее использовать

- Back Propagation при $n \gg m$,
- Forward Propagation при $n \leq m$.