Онлайн-курс

ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

3. Сложные функции и графы вычислений

Автор: Шевляков Артём Николаевич

Сложные функции одного аргумента

Сложная функция

- **это такой математический термин, не несущий в себе эмоциональной окраски.**

Пусть $f(x)$, $g(x)$ – две функции аргумента x. Тогда функция $h(x)=f(g(x))$ называется **сложной функцией** (или **суперпозицией** функций f,g).

Функция $h(x)$ получается из $f(x)$ заменой каждого вхождения буквы x на выражение $g(x)$. Такой вот копипаст.

Например, если $f(x)=x\sin(x)$, $g(x)=x^2$, то $h(x)=f(g(x))=x^2\sin(x^2)$.

Производная сложной функции

Со школы известно правило: если $h(x)=f(g(x))$, то

$$h'(x)=f'(g(x))g'(x)$$

Например, есть функция $h(x) = e^{x^2}$. Чтобы найти ее производную надо представить ее как сложную функцию: $h(x)=f(g(x))$, где $f(x)=e^x$, $g(x)=x^2$. Тогда получим: $f'(x)=e^x$, $g'(x)=2x$, и поэтому $h'(x) = e^{x^2} \cdot 2x$

Аналогично для $h(x) = \frac{1}{\sin x}$ получаем $h(x)=f(g(x))$, где $f(x)=1/x$, $g(x)=\sin x$. Тогда $f'(x)=-1/x^2$, $g'(x)=\cos x$, и поэтому $h'(x) = \frac{-\cos x}{\sin^2 x}$

Рекурсивное вычисление производной

Если в выражении $h(x)=f(g(x))$ функция $g(x)=g_1(g_2(x))$ снова является сложной, то процесс рекурсивно повторяется

$$h'(x)=f'(g(x))g'(x)=f'(g_1(g_2(x)))g'_1(g_2(x))g'_2(x)$$

Таким образом, вычислять производную сложной функции можно с помощью вычисления производных всё более и более глубоких выражений.

Пример

Пусть $h(x) = \sin(\cos x^2)$, обозначим $f(x) = \sin(x),\ g(x) = (\cos x^2)$, тогда

$$h'(x) = \cos(g(x))g'(x)$$

Распишем *g(x)*:

$$g(x) = g_1(g_2(x)),\ g_1(x) = \cos x,\ g_2(x) = x^2$$

тогда $g'(x) = g_1'(g_2(x))g_2'(x) = -\sin(x^2)2x$

и окончательно получаем

$$h'(x) = \cos(\cos x^2) \cdot (-2x \sin(x^2))$$

Идея, что вычислении производной нужно спуститься до внутренних фрагментов выражения, лежит в основе алгоритма backpropagation.

Онлайн-курс

ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

3. Сложные функции и графы вычислений

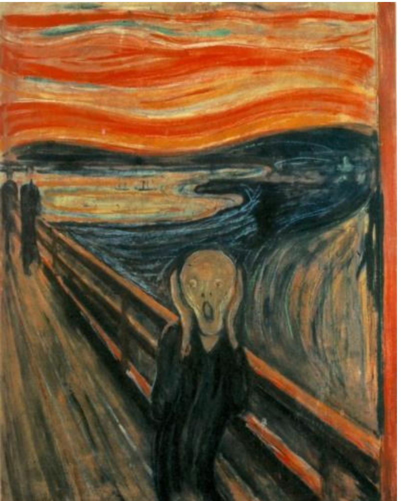
Сложные функции многих переменных

Начнём с двух аргументов

Пусть $f(x_1,x_2)$, $g_1(x_1,x_2)$, $g_2(x_1,x_2)$ − функции аргументов x_1,x_2. Тогда функция $f(g_1(x_1,x_2),g_2(x_1,x_2))$ называется **сложной функцией** (или **суперпозицией** функций f,g_1,g_2.

Формула для ЧП сложной функции $h(x_1,x_2)$:

Это не дробь!
Не вздумайте её сокращать



$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_1} + \frac{\partial f}{\partial g_2}\frac{\partial g_2}{\partial x_1}$$

$$\frac{\partial f}{\partial x_2} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_2} + \frac{\partial f}{\partial g_2}\frac{\partial g_2}{\partial x_2}$$

Пример 1

Пусть $f(x)=\sin x$, $g_1(x_1,x_2)=x_1x_2$ **и тогда для сложной функции** $f(g_1(x_1,x_2))=\sin g_1$ **имеем:**

$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_1} = \cos(x_1x_2)\cdot x_2$$

$$\frac{\partial f}{\partial x_2} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_2} = \cos(x_1x_2)\cdot x_1$$

Пример 2

Пусть $f(x_1,x_2)=x_1/x_2$, $g_1(x_1,x_2)=x_1x_2$, $g_2(x_1,x_2)=\sqrt{3x_1+5x_2}$

и тогда для сложной функции $f(g_1(x_1,x_2),g_2(x_1,x_2))=g_1(x_1,x_2)/g_2(x_1,x_2)$ **имеем:**

$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_1} + \frac{\partial f}{\partial g_2}\frac{\partial g_2}{\partial x_1} = \frac{1}{g_2}x_2 - \frac{g_1}{g_2^2}\frac{3}{2\sqrt{3x_1+5x_2}}$$

$$\frac{\partial f}{\partial x_2} = \frac{\partial f}{\partial x_2} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_2} + \frac{\partial f}{\partial g_2}\frac{\partial g_2}{\partial x_2} = \frac{1}{g_2}x_1 - \frac{g_1}{g_2^2}\frac{5}{2\sqrt{3x_1+5x_2}}$$

Пример 3

Пусть $f(x_1, x_2) = x_1^{x_2}$ $g_1(x_1,x_2)=x_1+x_2$, $g_2(x_1,x_2)=x_1x_2$ **и тогда для сложной функции**

$$f(g_1(x_1,x_2), g_2(x_1,x_2)) = (x_1 + x_2)^{x_1 x_2} \quad f = g_1^{g2}$$

имеем

$$\frac{\partial f}{\partial g_1} = g_2 g_1^{g_2 - 1}, \quad \frac{\partial f}{\partial g_2} = g_1^{g_2} \ln g_1$$

$$\frac{\partial f}{\partial x_1} = g_2 g_1^{g_2 - 1} \cdot 1 + g_1^{g_2} \ln g_1 \cdot x_2$$

$$\frac{\partial f}{\partial x_2} = g_2 g_1^{g_2 - 1} \cdot 1 + g_1^{g_2} \ln g_1 \cdot x_1$$

Общая формула

Если $f(x_1, \ldots, x_m),\ g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n)$ **, то**

$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_1} + \ldots + \frac{\partial f}{\partial g_m}\frac{\partial g_m}{\partial x_1}$$

$$\ldots$$

$$\frac{\partial f}{\partial x_n} = \frac{\partial f}{\partial g_1}\frac{\partial g_1}{\partial x_2} + \ldots + \frac{\partial f}{\partial g_m}\frac{\partial g_m}{\partial x_n}$$

Вот и живите теперь с этим!

Онлайн-курс

ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

3. Сложные функции и графы вычислений

Графы вычислений

Представление функции в виде графа

Как вы видели, ФМП являются сложными выражениями, одни фрагменты выражения вкладываются в другие.

Как визуально представлять процесс вычисления значения функции?

Да в виде **графа**!

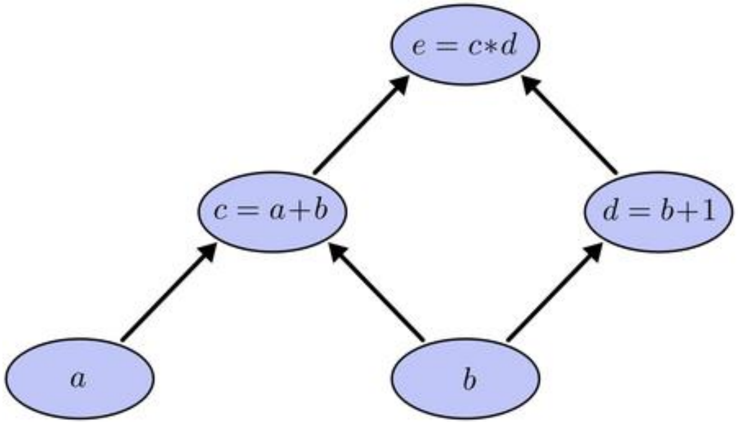
Он будет называться **граф вычислений**.

Граф – это...

у него есть узлы и стрелки между ними. В каждом узле стоит операция, применяемая к выражениям, входящим в узел.

Узлы, в которые не входят стрелки, называются листьями, и в них стоят переменные.

Еще есть корень – единственная вершина, из которой ничего не выходит.



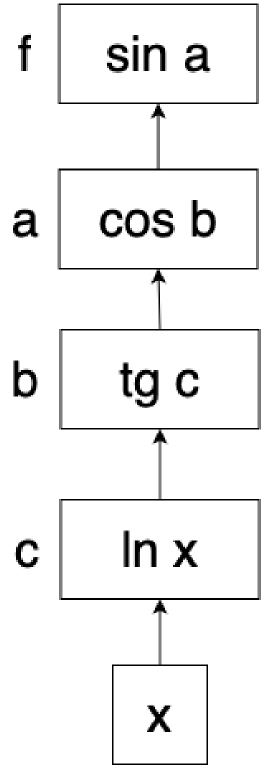
Пример графа вычислений

$f = \sin(\cos(tg(\ln x)))$

Аргумент x – в листе, корень соответствует всей функции.

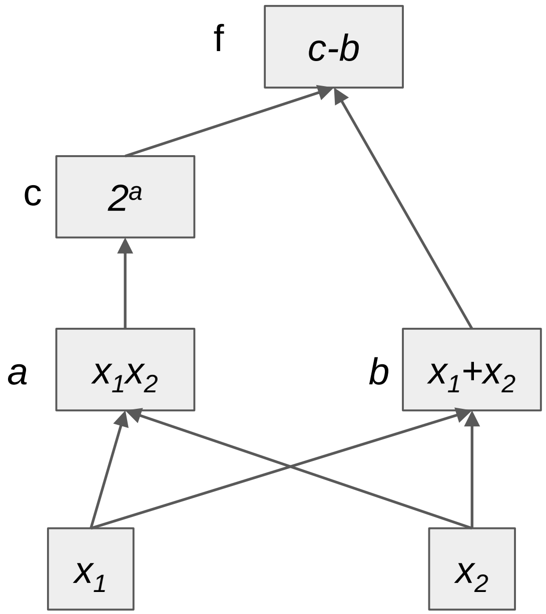
Граф определяет выражение

$f(x)=\sin(a)=\sin(\cos(b))=\sin(\cos(tg(c)))=$
$=\sin(\cos(tg(\ln(x))))$



Вычисление значений функций по графу

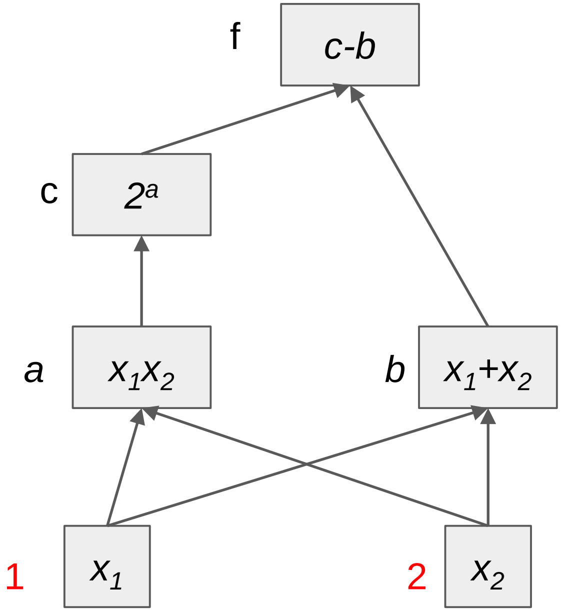
Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



Найдем $f(1,2)$ с помощью графа.
Для этого вычисляем значение каждого узла, начиная с листьев.

Вычисление значений функций по графу

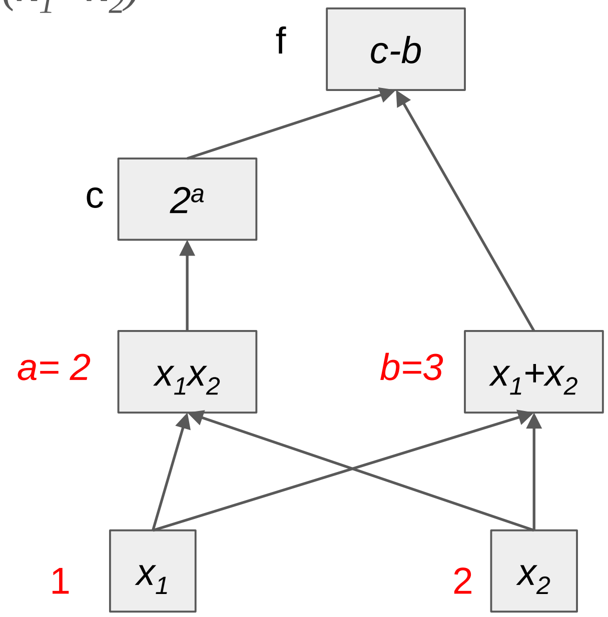
Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c-b=2^a-b=2^{x_1x_2}-(x_1+x_2)$



Найдем $f(1,2)$ с помощью графа.
Для этого вычисляем значение каждого узла, начиная с листьев.

Вычисление значений функций по графу

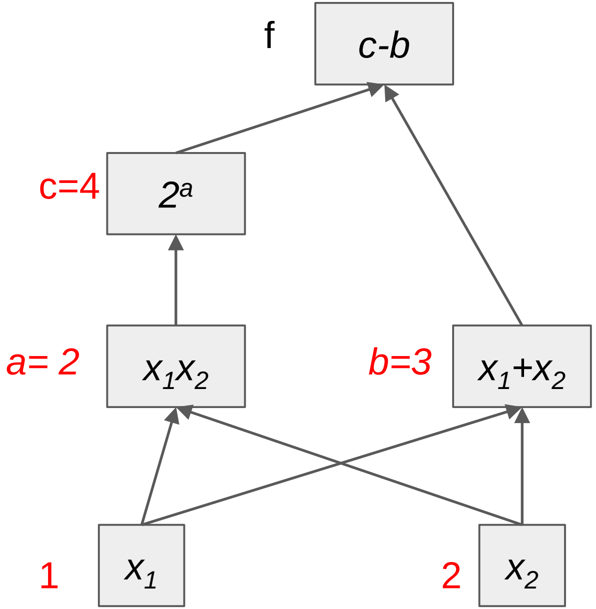
Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



Найдем $f(1,2)$ с помощью графа.
Для этого вычисляем значение каждого узла, начиная с листьев.

Вычисление значений функций по графу

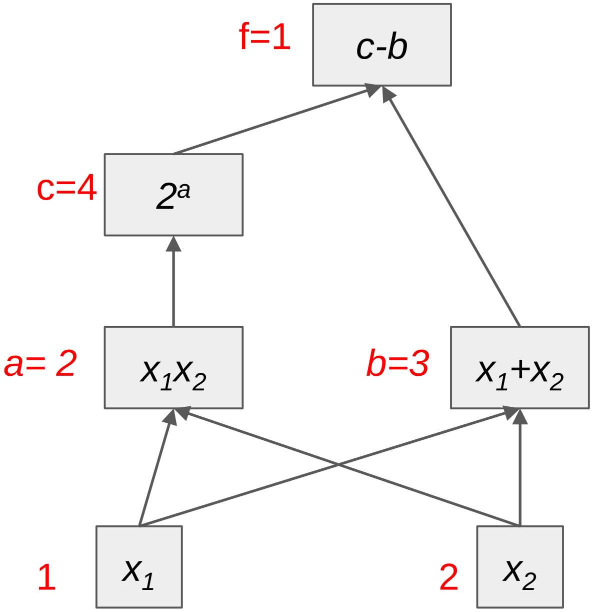
Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c-b=2^a-b=2^{x_1x_2}-(x_1+x_2)$



Найдем $f(1,2)$ с помощью графа.
Для этого вычисляем значение каждого узла, начиная с листьев.

Вычисление значений функций по графу

Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



Найдем $f(1,2)$ с помощью графа. Для этого вычисляем значение каждого узла, начиная с листьев.

Онлайн-курс

ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

3. Сложные функции и графы вычислений

Алгоритм обратного распространения

Вычисление ЧП по графу вычислений

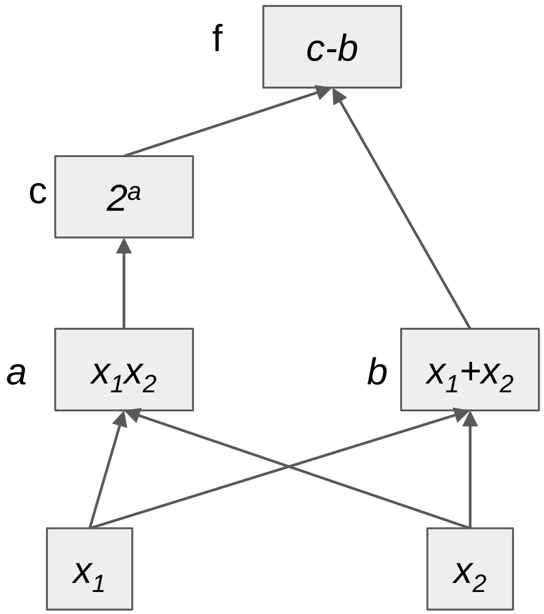
По графу вычислений можно вычислить не только значение функции в точке, а найти выражения для всех её ЧП.

Для этого в графе нужно **идти от корня к листьям**. Это моделирует вычисления производной по формуле дифференцирования сложной функции.

Этот алгоритм называется алгоритмом обратного распространения (back propagation, backprop).

Вычисление ЧП по графу

Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



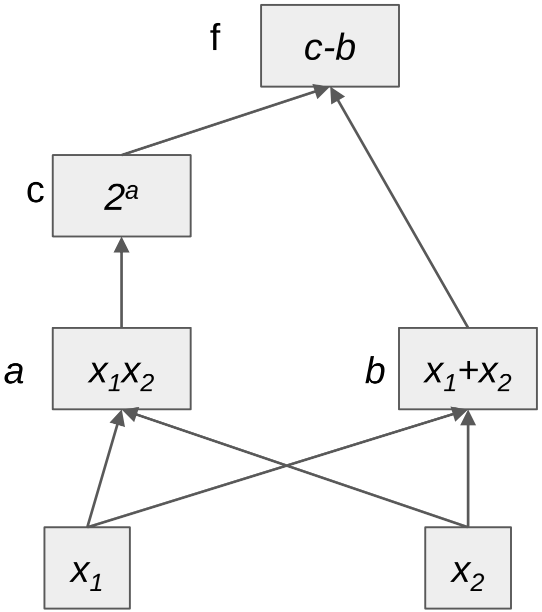
Найдем ее ЧП с помощью обхода графа от корня к листьям.

Будем последовательно вычислять ЧП $\partial f / \partial v$

где v - очередной узел графа, начиная с верхних.

Вычисление ЧП по графу

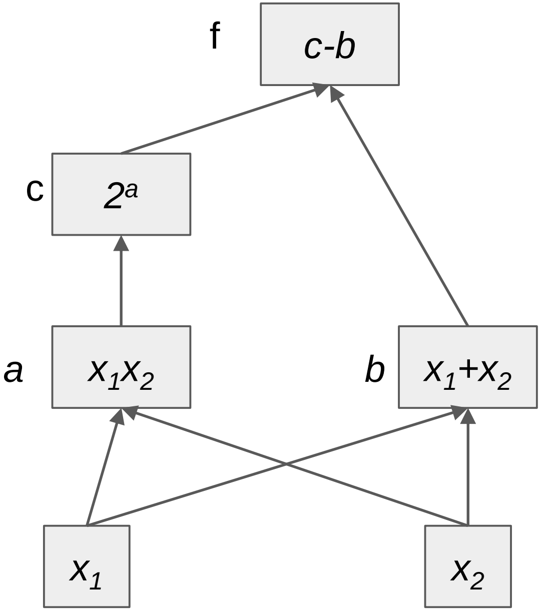
Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



$$\partial f/\partial c = \partial(c-b)/\partial c = 1,$$
$$\partial f/\partial b = \partial(c-b)/\partial b = -1$$

Вычисление ЧП по графу

Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$

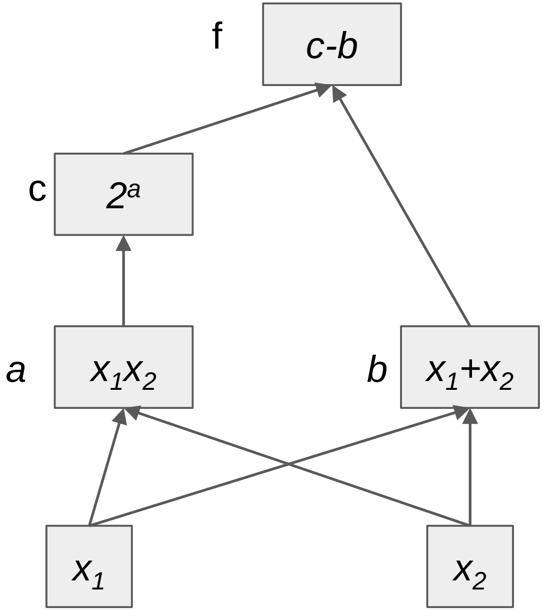


$$\partial f/\partial c = \partial(c-b)/\partial c = 1,$$
$$\partial f/\partial b = \partial(c-b)/\partial b = -1$$

$$\partial f/\partial a = \partial f/\partial c{\cdot}\partial c/\partial a = 1{\cdot}2^a \ln a = 2^a \ln a,$$

Вычисление ЧП по графу

Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



$$\partial f/\partial c = \partial(c-b)/\partial c = 1,$$
$$\partial f/\partial b = \partial(c-b)/\partial b = -1$$

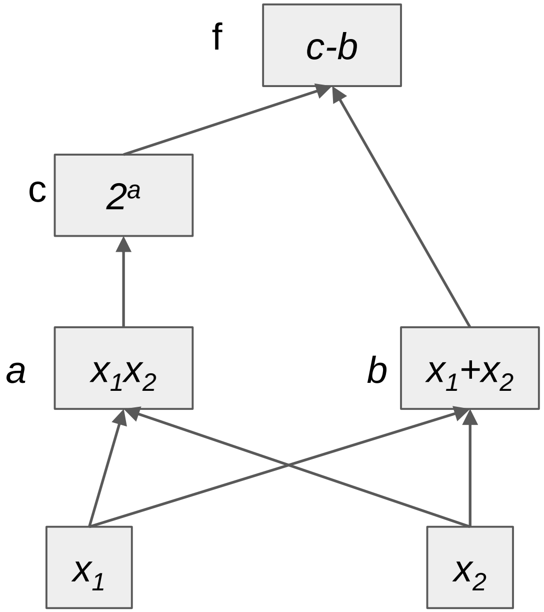
$$\partial f/\partial a = \partial f/\partial c \cdot \partial c/\partial a = 1 \cdot 2^a \ln a = 2^a \ln a,$$

$$\partial f/\partial x_1 = \partial f/\partial a \cdot \partial a/\partial x_1 + \partial f/\partial b \cdot \partial b/\partial x_1 =$$

$$= 2^a \ln a \cdot x_2 - 1 \cdot 1 = 2^a x_2 \ln a - 1$$

Вычисление ЧП по графу

Этот граф определяет функцию двух аргументов $f(x_1,x_2)=c\text{-}b=2^a\text{-}b=2^{x_1x_2}\text{-}(x_1+x_2)$



$$\partial f/\partial c = \partial(c-b)/\partial c = 1,$$
$$\partial f/\partial b = \partial(c-b)/\partial b = -1$$

$$\partial f/\partial a = \partial f/\partial c \cdot \partial c/\partial a = 1 \cdot 2^a \ln a = 2^a \ln a,$$
$$\partial f/\partial x_1 = \partial f/\partial a \cdot \partial a/\partial x_1 + \partial f/\partial b \cdot \partial b/\partial x_1 =$$

$$= 2^a \ln a \cdot x_2 - 1 \cdot 1 = 2^a x_2 \ln a - 1$$
$$\partial f/\partial x_2 = \partial f/\partial a \cdot \partial a/\partial x_2 + \partial f/\partial b \cdot \partial b/\partial x_2 =$$

$$= 2^a \ln a \cdot x_1 - 1 \cdot 1 = 2^a x_1 \ln a - 1$$

Основные правила алгоритма backprop

- Начать с корня, посчитать ЧП по всем буквам, входящим в корень.
- Взять узел графа *v* такой, что для всех его соседних узлов сверху $v_1,\dots,v_k$ ЧП $\partial f/\partial v_i$ уже подсчитаны.
- Тогда ЧП $\partial f/\partial v$ считается по формуле (18+):

$$\frac{\partial f}{\partial v} = \frac{\partial f}{\partial v_1}\frac{\partial v_1}{\partial v} + \dots + \frac{\partial f}{\partial v_k}\frac{\partial v_k}{\partial v}$$

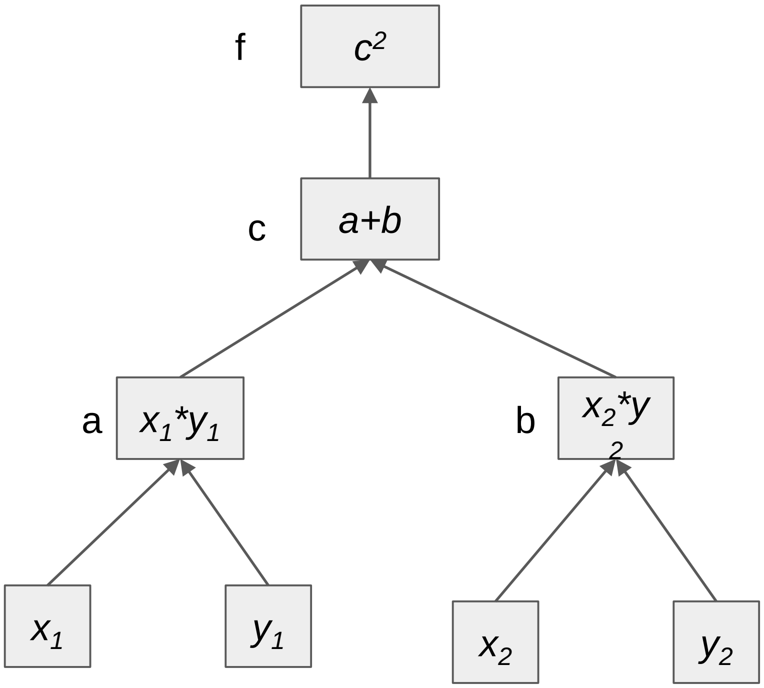
Ну и чтобы окончательно вас добить (18+):

$$\frac{\partial f}{\partial v} = \sum_{i=1}^{k} \frac{\partial f}{\partial v_i}\frac{\partial v_i}{\partial v}$$

Еще пример

Дан граф вычисления функции f.

Он моделирует вычисления внутри искусственных нейронов.



Еще пример

Ищем ЧП сверху вниз:

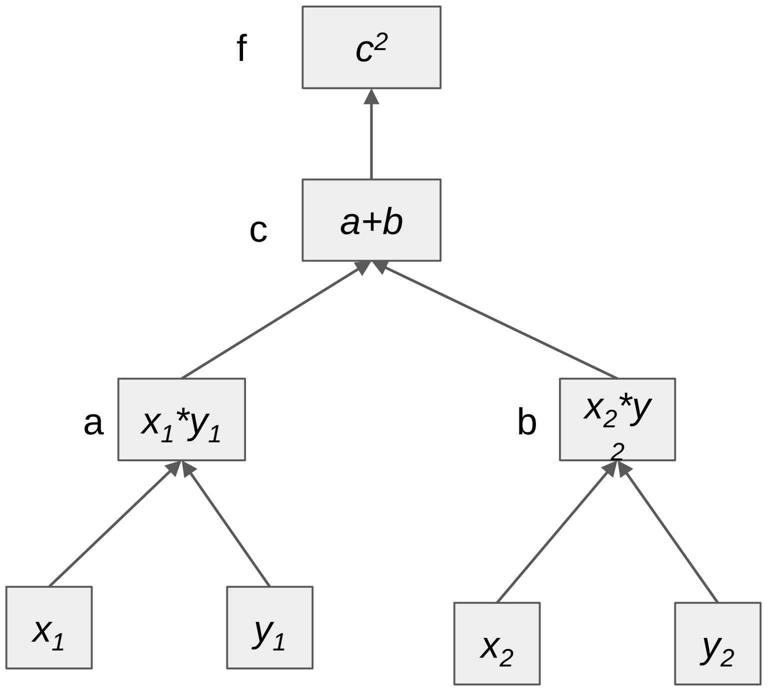
$$\frac{\partial f}{\partial c} = 2c,$$

$$\frac{\partial f}{\partial a} = \frac{\partial f}{\partial c}\frac{\partial c}{\partial a} = 2c \cdot 1 = 2c,$$

$$\frac{\partial f}{\partial b} = \frac{\partial f}{\partial c}\frac{\partial c}{\partial b} = 2c \cdot 1 = 2c,$$

$$\frac{\partial f}{\partial x_1} = \frac{\partial f}{\partial a}\frac{\partial a}{\partial x_1} = 2cy_1,$$

$$\frac{\partial f}{\partial y_1} = \frac{\partial f}{\partial a}\frac{\partial a}{\partial y_1} = 2cx_1,$$

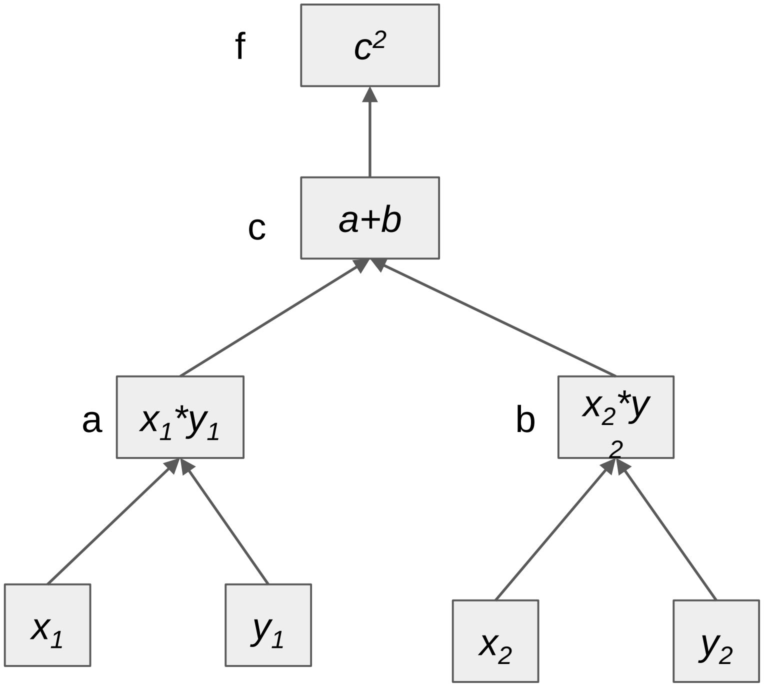


Еще пример

$$\frac{\partial f}{\partial x_2} = \frac{\partial f}{\partial b}\frac{\partial b}{\partial x_2} = 2cy_2,$$

$$\frac{\partial f}{\partial y_2} = \frac{\partial f}{\partial a}\frac{\partial a}{\partial y_2} = 2cx_2$$

Чтобы получить окончательные выражения, нужно буквы a,b,c выразить через переменные x_1,x_2,y_1,y_2.



Еще пример

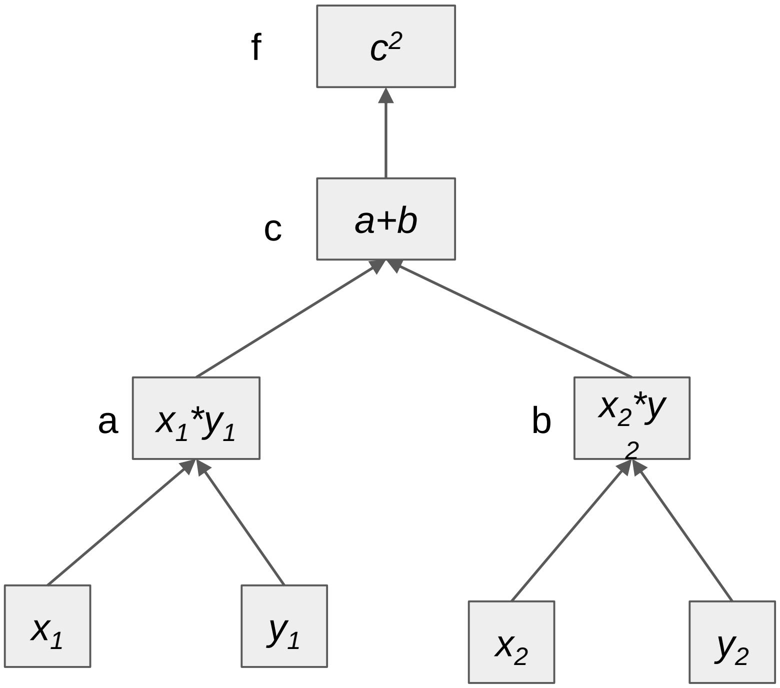
Чтобы получить окончательные выражения, нужно буквы a,b,c выразить через переменные x_1, x_2, y_1, y_2.

$$\frac{\partial f}{\partial x_1} = 2cy_1 = 2y_1(x_1y_1 + x_2y_2),$$

$$\frac{\partial f}{\partial y_1} = 2cx_1 = 2x_1(x_1y_1 + x_2y_2),$$

$$\frac{\partial f}{\partial x_2} = 2cy_2 = 2y_2(x_1y_1 + x_2y_2),$$

$$\frac{\partial f}{\partial y_2} = 2cx_2 = 2x_2(x_1y_1 + x_2y_2),$$



Онлайн-курс

ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

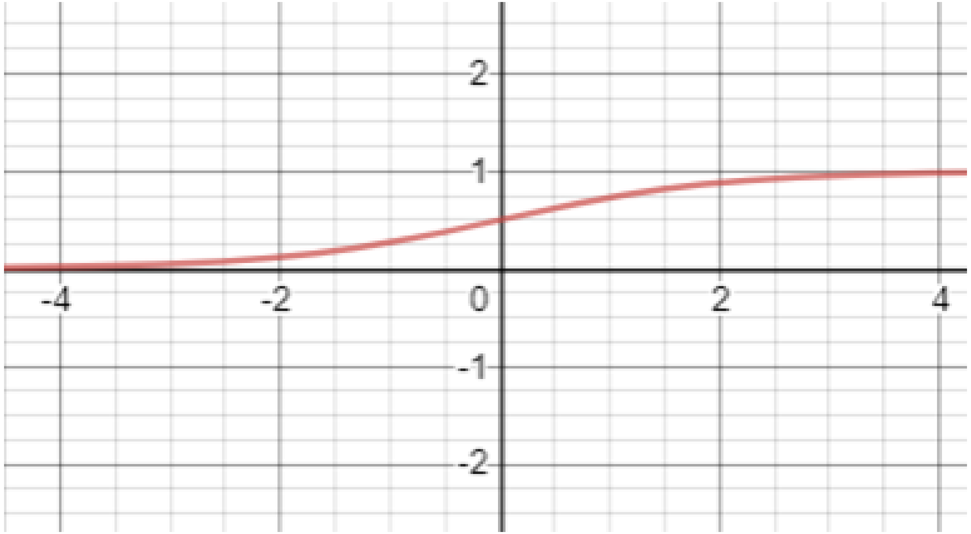
3. Сложные функции и графы вычислений

Сигмоида

Сигмоидальная функция (сигмоида)

В теории нейронных сетей целая эпоха связана с этой функцией

$$\sigma(x) = \frac{e^x}{1 + e^x}$$



Производная сигмоиды

С помощью школьных правил дифференцирования получим

$$\sigma'(x) = \left(\frac{e^x}{1+e^x}\right)' = \frac{e^x(1+e^x) - e^x e^x}{(1+e^x)^2} =$$
$$= \frac{e^x}{1+e^x} \cdot \frac{1}{1+e^x} = \frac{e^x}{1+e^x} \cdot \left(1 - \frac{e^x}{1+e^x}\right) =$$
$$= \sigma(x)(1-\sigma(x))$$

Производные суперпозиции сигмоид

Дифференцируя как сложную функцию, получим:

$$(\sigma(\sigma(x)))' = \sigma(\sigma(x))(1 - \sigma(\sigma(x))) \cdot$$
$$\cdot\, \sigma(x)(1 - \sigma(x)),$$

$$(\sigma(\sigma(\sigma(x))))' = \sigma(\sigma(\sigma(x)))(1 - \sigma(\sigma(\sigma(x)))) \cdot$$
$$\cdot\, \sigma(\sigma(x))(1 - \sigma(\sigma(x))) \cdot$$
$$\cdot\, \sigma(x)(1 - \sigma(x))$$

Чем больше суперпозиция, тем больше множителей в производной.

Производные суперпозиции сигмоид

Но все эти множители положительны и строго меньше 1. А произведение таких чисел стремится к 0 с возрастанием числа множителей.

$$(\sigma(\sigma(x)))' = \sigma(\sigma(x))(1 - \sigma(\sigma(x))) \cdot$$
$$\cdot\, \sigma(x)(1 - \sigma(x)),$$

$$(\sigma(\sigma(\sigma(x))))' = \sigma(\sigma(\sigma(x)))(1 - \sigma(\sigma(\sigma(x)))) \cdot$$
$$\cdot\, \sigma(\sigma(x))(1 - \sigma(\sigma(x))) \cdot$$
$$\cdot\, \sigma(x)(1 - \sigma(x))$$

Следовательно, для большой суперпозиции $f(x) = (\sigma(\ldots(\sigma(x))\ldots))$ **для почти всех значений *a* будет выполнено** $f'(a) \approx 0$

Ну и что?

Если для почти всех значений *a* выполнено $f'(a) \approx 0$, то это означает трудности с нахождением минимума функции с помощью ГС.

Иными словами, функция *f(x)* почти всюду имеет относительно ровное плато. В этом случае ГС не сможет существенно изменять положение точки, так как

$$a_{n+1} := a_n - hf'(a_n) \approx a_n - h \cdot 0 \approx a_n$$

Например, уже значения первой производной сигмоиды почти нулевые:

$$\sigma'(5) = 0.0067, \ \sigma'(10) = 0.00005$$

Ну и что?

Из последующих глав будет понятно, почему:

- сигмоиду сейчас не используют в построении искусственных нейронных сетей;
- тем не менее её когда-то начинали использовать в нейронных сетях, и в те времена считалось, что сигмоида – это круто!

Онлайн-курс

ВВЕДЕНИЕ В ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

3. Сложные функции и графы вычислений

Выводы

Выводы:

- Мы научились представлять функции в виде графов.
- Мы показали, что по графу можно находить частные производные функции.
- Данный алгоритм будет использоваться при тренировке нейронных сетей (алгоритм обратного распространения ошибки, backpropagation).
- Мы познакомились с интересной функцией под названием сигмоида. Ждите ее появления в лекции по искусственным нейронам.