Глубокое обучение и вообще

Ульянкин Филипп

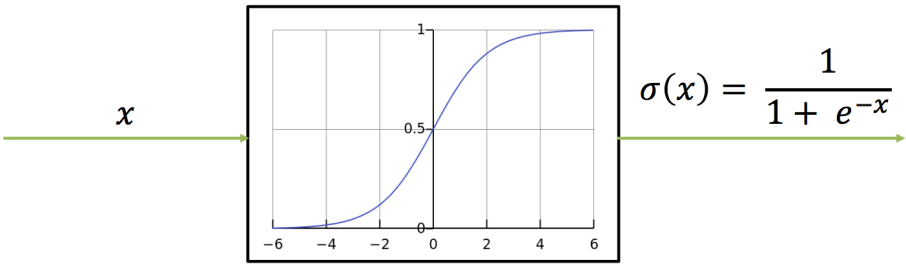
Посиделка 6: Нейросети — конструктор LEGO (часть 1)

Agenda

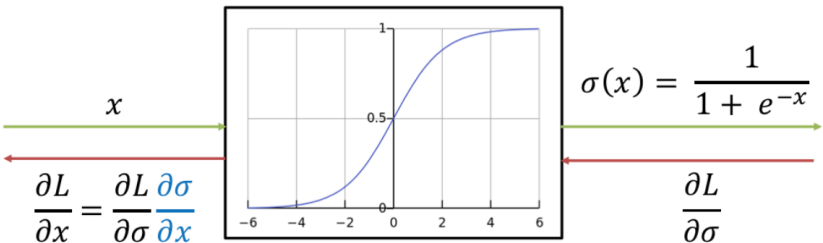
- Какими бывают функции активации
- Паралич нейронной сети и взрыв градиентов
- Инициализация весов
- Переобучение нейронных сетей
- Регуляризация: l_1, l_2, дропаут, ранняя остановка, их взаимосвязь
- Первая порция советов по обучению нейросетей

Какими бывают функции активации

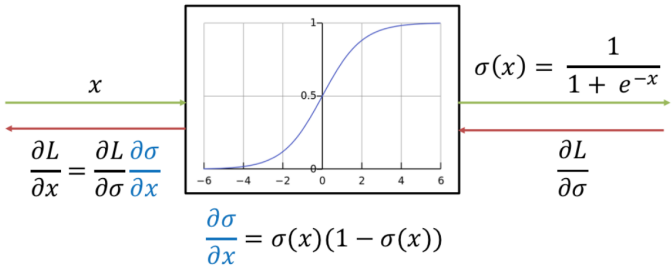
Сигмоида (sigmoid activation)



Сигмоида (sigmoid activation)



Сигмоида (sigmoid activation)



- Сигмоида была популярна как классическая функция активации, но у неё есть ряд проблем
- Насыщенные нейроны зануляют градиенты, в глубоких сетях возможен паралич

Затухание градиента (vanishing gradient problem)

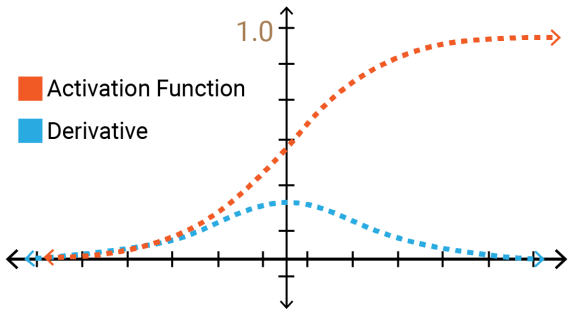
- Изменение параметров в ходе обучения происходит на величину, которая не влияет на выход сети

- Проблема связана с очень маленькими градиентами при обновлении весов:

$$w_t = w_{t-1} - \gamma \cdot \nabla L(w_{t-1})$$

- При насыщении сети сигмоида убивает градиенты

$$f(t) = \frac{1}{1 + e^{-t}}$$



Затухание градиента (vanishing gradient problem)

- Изменение параметров в ходе обучения происходит на величину, которая не влияет на выход сети

- Проблема связана с очень маленькими градиентами при обновлении весов:

$$w_t = w_{t-1} - \gamma \cdot \nabla L(w_{t-1})$$

- При насыщении сети сигмоида убивает градиенты

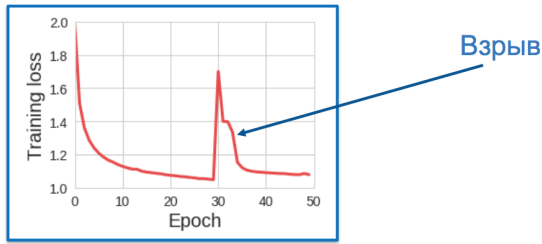
Толи обучение сошлось, толи веса просто больше не обновляются …

Паралич сети

- В случае сигмоиды $\sigma'(x) = \sigma(x) \cdot (1 - \sigma(x))$
- Сигмоида принимает значения на отрезке $[0; 1]$, значит максимальное значение её производной это $^1/_4$
- Если сеть очень глубокая, происходит затухание градиента
- Градиент затухает экспоненциально $\Rightarrow$ сходимость замедляется, более ранние веса обновляются дольше, более глубокие веса быстрее $\Rightarrow$ значение градиента становится ещё меньше $\Rightarrow$ наступает паралич сети
- В сетях с небольшим числом слоёв этот эффект незаметен

Взрыв градиента (exploding gradient problem)

- Изменение параметров в ходе обучения происходит на очень большую величину и обучение деградирует
- Проблема связана с очень большими градиентами при обновлении весов:

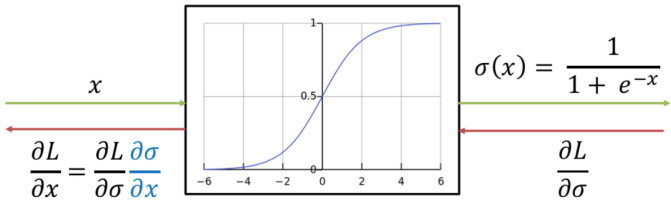


$$w_t = w_{t-1} - \gamma \cdot \nabla L(w_{t-1})$$

Взрыв и затухание градиента

- Затухание градиента связано не только с сигмоидой
- Взрыв градиента также встречается довольно часто
- $\Rightarrow$ грамотные инициализация и оптимизация
- $\Rightarrow$ разработка новых архитектур и специальных слоёв

Сигмоида (sigmoid activation)



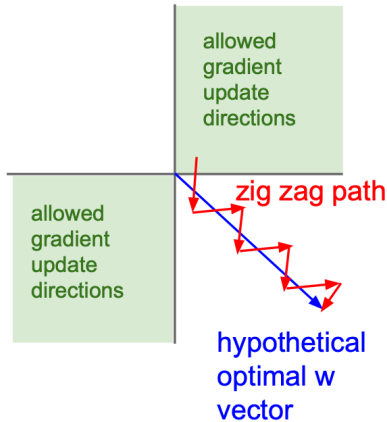
- Насыщенные нейроны зануляют градиенты, в глубоких сетях возможен паралич
- Не центрирована относительно нуля
- Что мы можем сказать о градиентах нейрона с сигмоидой?

Центрирование относительно нуля

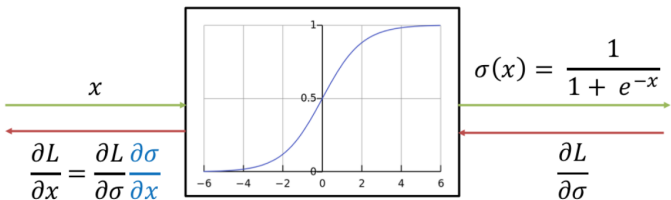
- Выход слоя мы обычно находим как

$$o_i = \sigma(h_i)$$

- он всегда положительный, значит градиент по весам, идущим на вход в текущий нейрон тоже положительные
- $\Rightarrow$ все веса обновляются в одинаковом направлении
- $\Rightarrow$ сходимость идёт медленнее, причём зиг-загами

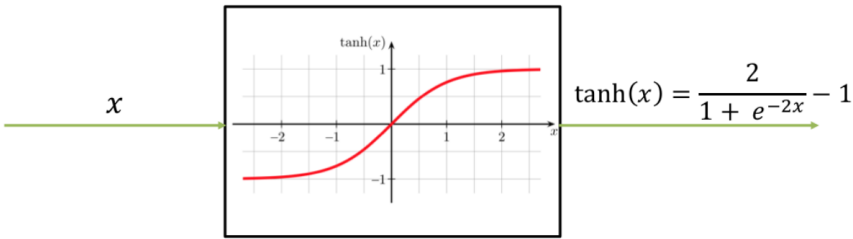


Сигмоида (sigmoid activation)



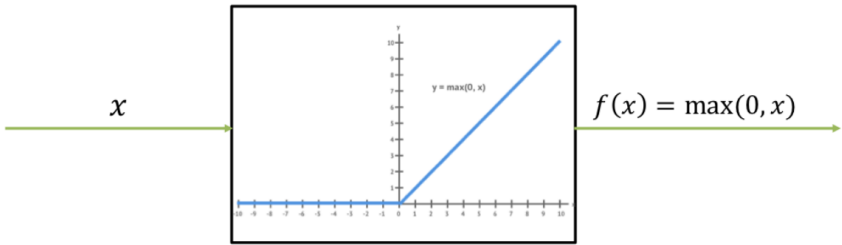
- Насыщенные нейроны зануляют градиенты, в глубоких сетях возможен паралич
- Не центрирована относительно нуля
- Вычислять e^x дорого

Гиперболический тангенс (tanh activation)



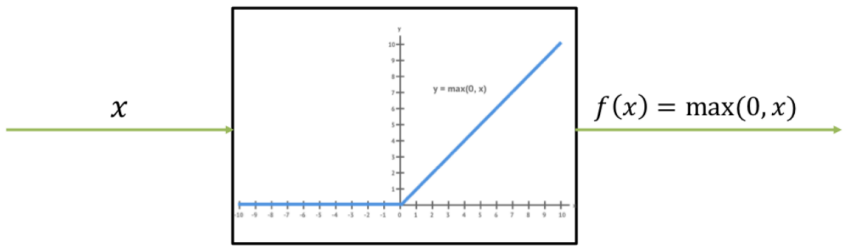
- Центрирован относительно нуля
- Всё ещё похож на сигмоиду
- $f'(x) = 1 - f(x)^2 \Rightarrow$ затухание градиента

Rectifier Linear Unit activation, ReLU (2012)



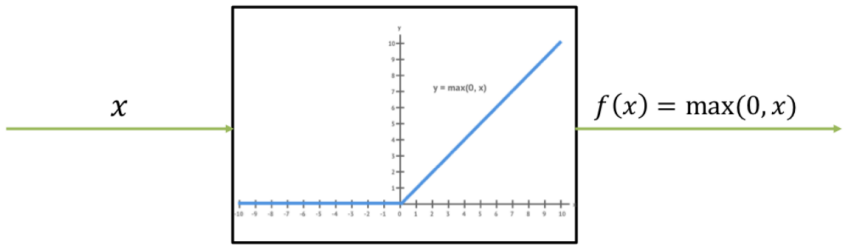
- Быстро вычисляется
- Градиенты не угасают при $x > 0$
- На практике ускоряет сходимость

ReLU (2012)



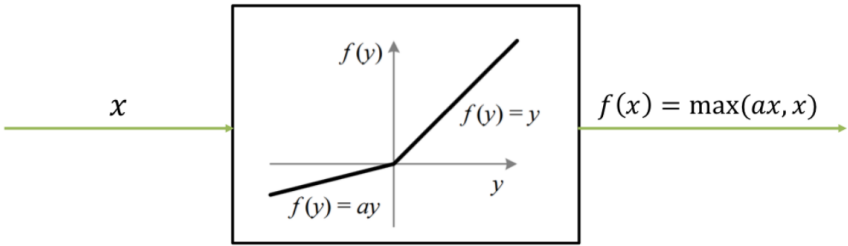
- Сетка может умереть, если активация занулится на всех нейронах
- Не центрирован относительно нуля

Зануление ReLU



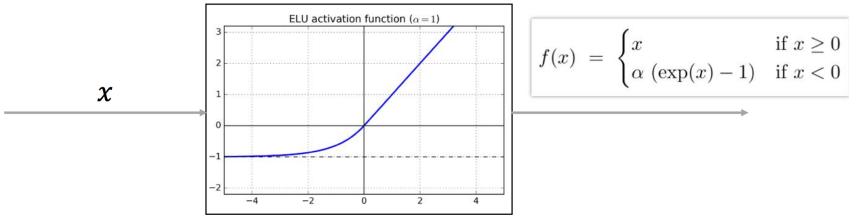
- $f(x) = \max(0, w_0 + w_1 \cdot h_1 + ... + w_k \cdot h_k)$
- Если w_0 инициализировано большим отрицательным числом, нейрон сразу умирает $\Rightarrow$ надо аккуратно инициализировать веса

Leaky ReLU activation (2013)

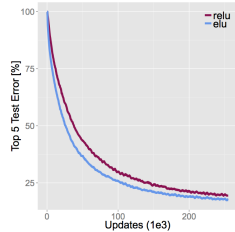


- Как ReLU, но не умирает, всё ещё легко считается
- Производная может быть любого знака
- Важно, чтобы $a \neq 1$, иначе линейность
- Не центрирован относительно нуля

Exponential Linear Units activation, ELU (2015)



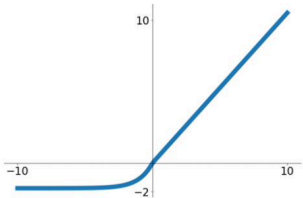
- Примерно центрирован в нуле (в контексте математического ожидания)
- Сходимость быстрее ReLU
- На практике ускоряет сходимость



(b) Top-5 test error

https://arxiv.org/pdf/1511.07289.pdf

Scaled Exponential Linear Units activation, SELU (2017)



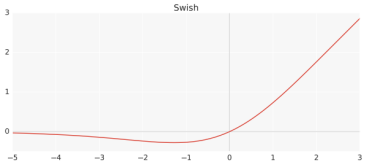
$$f(x) = \begin{cases} \lambda x & \text{if } x > 0 \\ \lambda\alpha(e^x - 1) & \text{otherwise} \end{cases}$$

α = 1.6733, λ = 1.0507

- Нормализованная версия ELU, лучше работает для глубоких сетей
- Обладает свойством самонормализации
- Можно учить сетки без нормализации по батчам (будем обсуждать позже)

https://arxiv.org/abs/1706.02515

Swish (2017)



$$f(x) = x \cdot \sigma(\beta x)$$

параметр β обучается

- В 2017 году Google Brain хитрым автоматическим поиском на основе RNN нашла функцию активации Swish, работающую лучше ReLU

Function	RN	WRN	DN
ReLU $[\max(x, 0)]$	93.8	95.3	94.8
$x \cdot \sigma(\beta x)$	94.5	95.5	94.9
$\max(x, \sigma(x))$	94.3	95.3	94.8
$\cos(x) - x$	94.1	94.8	94.6
$\min(x, \sin(x))$	94.0	95.1	94.4
$(\tan^{-1}(x))^2 - x$	93.9	94.7	94.9
$\max(x, \tanh(x))$	93.9	94.2	94.5
$\text{sinc}(x) + x$	91.5	92.1	92.0
$x \cdot (\sinh^{-1}(x))^2$	85.1	92.1	91.1

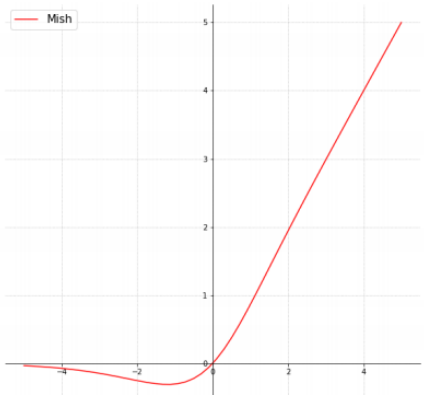
Table 1: CIFAR-10 accuracy.

Function	RN	WRN	DN
ReLU $[\max(x, 0)]$	74.2	77.8	83.7
$x \cdot \sigma(\beta x)$	75.1	78.0	83.9
$\max(x, \sigma(x))$	74.8	78.6	84.2
$\cos(x) - x$	75.2	76.6	81.8
$\min(x, \sin(x))$	73.4	77.1	74.3
$(\tan^{-1}(x))^2 - x$	75.2	76.7	83.1
$\max(x, \tanh(x))$	74.8	76.0	78.6
$\text{sinc}(x) + x$	66.1	68.3	67.9
$x \cdot (\sinh^{-1}(x))^2$	52.8	70.6	68.1

Table 2: CIFAR-100 accuracy.

https://arxiv.org/abs/1710.05941
https://krutikabapat.github.io/Swish-Vs-Mish-Latest-Activation-Functions/

Mish (2019)



Похожим образом получили функции активации Mish

$$f(x) = x \cdot tanh(\ln(1 + e^x))$$

https://arxiv.org/pdf/1908.08681.pdf
https://krutikabapat.github.io/Swish-Vs-Mish-Latest-Activation-Functions/

Activate or Not, ACON (2020)

- Swish -- гладкий вариант ReLU с гейтом
- Функция сама решает, нужна слою активация или нет
- Можно обобщить этот подход и придумать более интересные функции потерь, которые дадут улучшение

https://arxiv.org/pdf/2009.04759.pdf

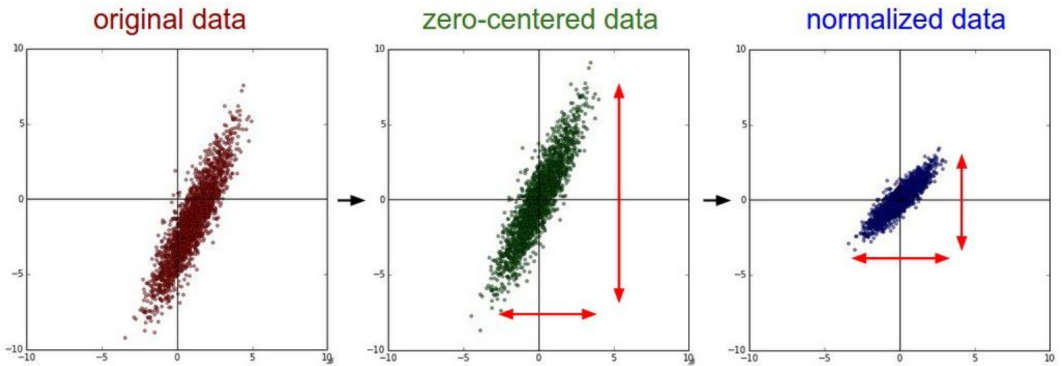
TLDR: что на практике

- Начните с ReLU, аккуратно инициализируйте веса и настраивайте скорость обучения
- Попробуйте Leaky ReLU / ELU / SELU / Swish / Mish, если есть время на эксперименты, чтобы выжать максимум
- Не используйте сигмоиду и tanh

Краткий обзор функций активаций: https://arxiv.org/pdf/1804.02763.pdf

Предобработка данных

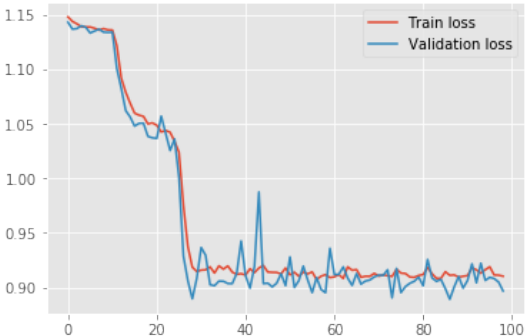
Предобработка данных



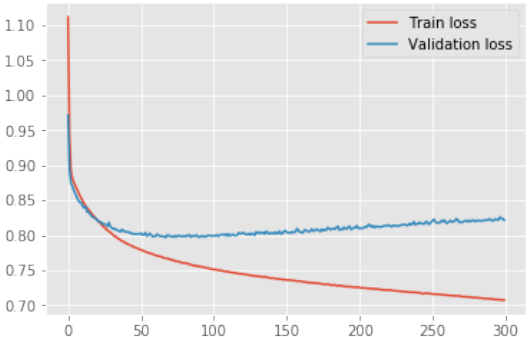
http://cs231n.stanford.edu/slides/2021/

Предобработка табличных данных

Без нормализации



С нормализацией



Зачем нормализация картинкам?

- Что происходит, когда все входы в нейрон положительные?
- Все градиенты либо положительные либо отрицательные
- Картинки центрируют для того, чтобы были отрицательные входы
- Картинки нормируют, чтобы инициализированным в окрестности нуля весам легче было сходиться