Основы практического использования нейронных сетей.

Лекция 3. Предобработка данных. Функция ошибки.

Дмитрий Буряк.
к.ф.-м.н.
dyb04@yandex.ru

Эффективность обучения

Значение функции потерь на обучающей выборке

- ❑ Предобработка данных
- ❑ Функция потерь – функция активации
- ❑ Затухающий градиент

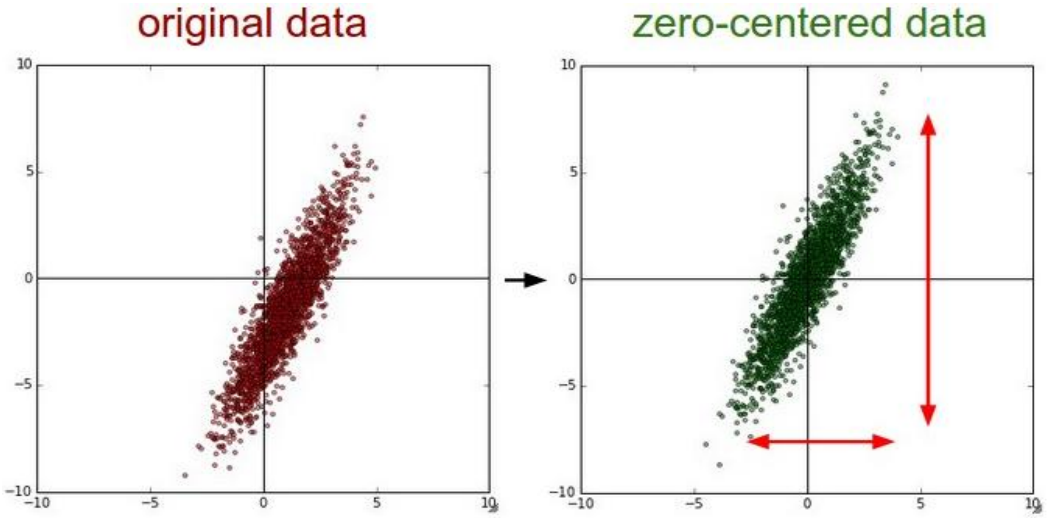
Основные этапы предобработки

- X – матрица (NxD) – вектора данных
  - N – число векторов,
  - D – размерность пространства (количество признаков).

- Вычитание среднего.
- Нормализация.
- Декорреляция.

Вычитание среднего

$$y_{ij} = x_{ij} - \underset{i}{mean}(x_{ij})$$



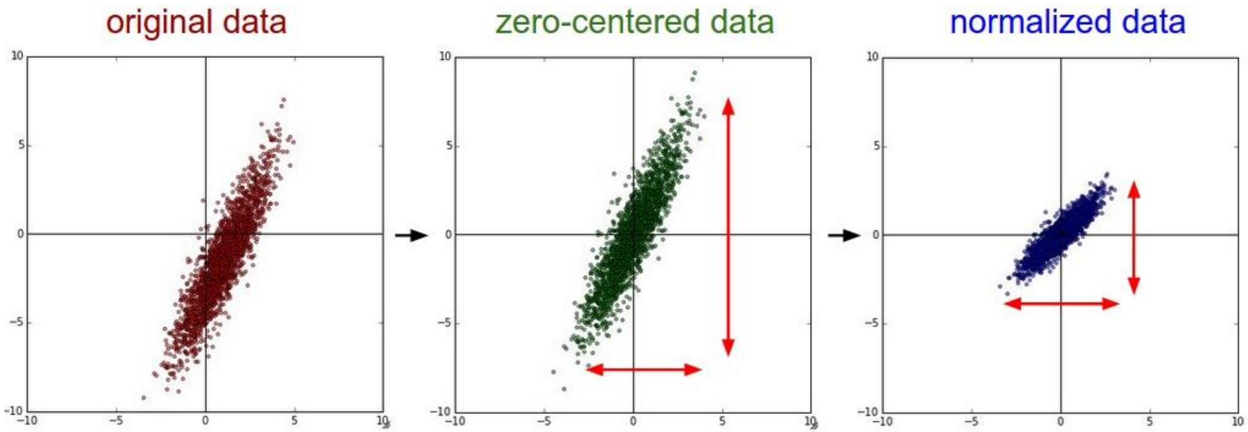
Нормализация

- Выравнивание диапазона изменения каждого признака

$$z_{ij} = \frac{y_{ij}}{\underset{i}{std}(y_{ij})}$$

или

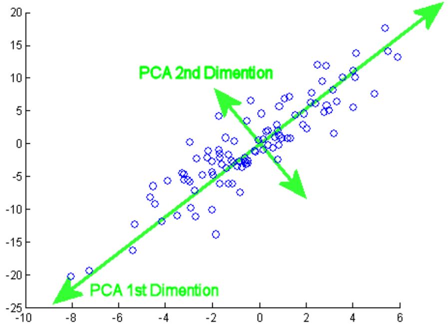
$$z_{ij} = \frac{y_{ij} - \min\limits_i(y_{ij})}{\max\limits_i(y_{ij}) - \min\limits_i(y_{ij})}$$

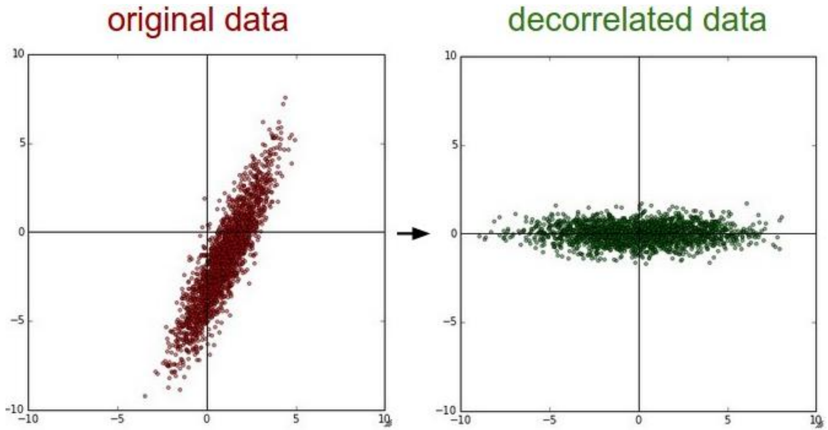


Декорреляция

- Удаление корреляции между признаками (PCA)

1. $y_{ij} = x_{ij} - \underset{i}{mean}(x_{ij})$
2. $C = Y * Y^T$ - матрица ковариации
3. Поиск собственных значений (S) и собственных векторов (U) матрицы C
4. $Z = Y * U^T$

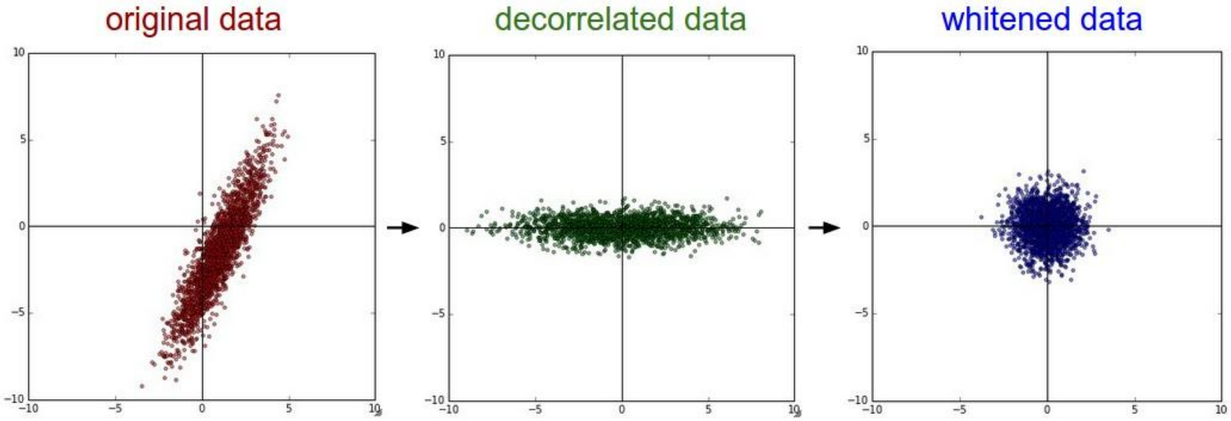




Выбеливание

- Удаление корреляции между признаками + выравнивание диапазона изменения каждого признака.

$$q_{ij} = \frac{z_{ij}}{\underset{i}{std}(z_{ij})}$$



Пример выбеливания

- Классификация изображений из базы CIFAR-10.

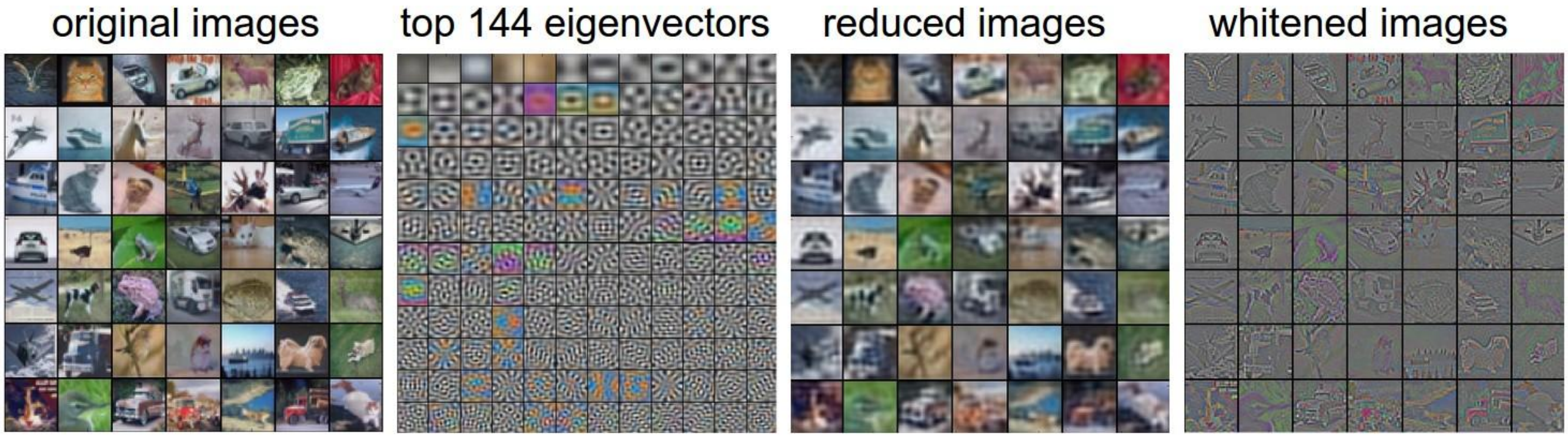


Схема применения предобработки

- X – обучающее множество, V – тестовые данные.

1. Вычисление параметров предобработки P по данным из X.
2. Выполнение предобработки: Y=F(X,P).
3. Обучение НС (NN) на данных из Y.
4. Выполнение предобработки тестовых данных: U=F(V,P)
5. Применение НС: NN(U)

- Вычитание среднего и нормализация – наиболее часто применяемые методы предобработки

Функция ошибки

$$E(X,Y,w) = C(X,Y,w) + \lambda R(w)$$

- C – функция потерь
  - X, Y – обучающее множество и желаемые выходы
  - w – настраиваемые параметры (веса, смещения)
  - C≥0
  - Вид функции потерь + функция активации в выходном слое → скорость сходимости алгоритма обучения.

- R – функция регуляризации

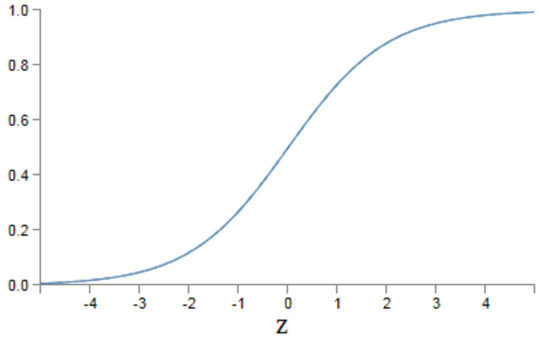
Среднеквадратичная функция (MSE)

$$C = \frac{1}{2n}\sum_i \|y_i - a_i\|^2$$

- 1 нейрон, 1 вход
- Функция активации - сигмоида

$$C = \frac{(y-a)^2}{2}$$

$$\frac{\partial C}{\partial w} = (a-y)\sigma'(z)x$$



- Зависимость от производной функции активации
- Большие ошибки (a≈0, y=1; a≈1, y=0) → область насыщения сигмоиды → замедление обучения
- MSE + сигмоида → низкая скорость сходимости алгоритма обучения

MSE + линейный нейрон

- ❑ Выходной слой – линейные нейроны

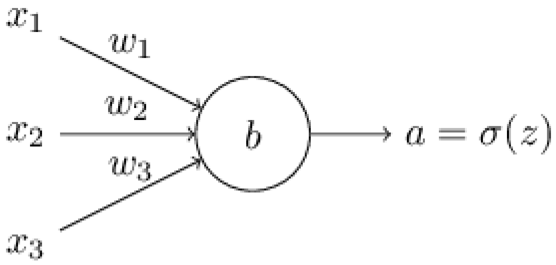
$$\frac{\partial C}{\partial w_{jk}^{L}} = \frac{1}{n} \sum_{x} a_{k}^{L-1} (a_{j}^{L} - y_{j})$$

- ❑ Зависимость от ошибки на выходе
- ❑ MSE + линейная функция активации → высокая скорость сходимости алгоритма обучения

Cross Entropy

$$C = -\frac{1}{n}\sum_i \sum_j \left[y_{ij} \ln a_{ij} - (1 - y_{ij}) \ln(1 - a_{ij}) \right]$$

❑ Функция активации - сигмоида



$$C = -\frac{1}{n}\sum_x \left[y \ln a + (1 - y) \ln(1 - a) \right]$$

$$\frac{\partial C}{\partial w_j} = \frac{1}{n}\sum_x \frac{\sigma'(z) x_j}{\sigma(z)(1 - \sigma(z))}(\sigma(z) - y)$$

$$\sigma'(z) = \sigma(z)(1 - \sigma(z))$$

$$\frac{\partial C}{\partial w_j} = \frac{1}{n}\sum_x x_j(\sigma(z) - y)$$

❑ Зависимость от ошибки на выходе

❑ Cross entropy + сигмоида → высокая скорость сходимости алгоритма обучения

Функция правдоподобия

$$C \equiv -\ln a_y^L$$

- N выходов, функция активации – Softmax $\quad a_j^L = \frac{e^{z_j^L}}{\sum_k e^{z_k^L}}$
- Входной вектор x относится к классу 1 → $a_1^L \to 1$ → $C \to \min$
  или → $a_1^L \to 0$ → $C \to \max$

$$\frac{\partial C}{\partial w_{jk}^L} = a_k^{L-1}(a_j^L - y_j)$$

- Зависимость от ошибки на выходе
- Функция правдоподобия + softmax → высокая скорость сходимости алгоритма обучения

Другие функции потерь

- Среднеквадратичная логарифмическая $$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\left(\log(y^{(i)}+1) - \log(\hat{y}^{(i)}+1)\right)^2$$
- Средняя по модулю (MAE) $$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\left|y^{(i)} - \hat{y}^{(i)}\right|$$
- Средняя по модулю в процентах (MAPE) $$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y^{(i)} - \hat{y}^{(i)}}{y^{(i)}}\right| . 100$$
- Расстоние KL (Kullback Leibler) $$\mathcal{L} = \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left(y^{(i)} . \log(y^{(i)})\right)}_{entropy} \underbrace{- \frac{1}{n}\sum_{i=1}^{n}\left(y^{(i)} . \log(\hat{y}^{(i)})\right)}_{cross-entropy}$$
- Hinge loss $$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y^{(i)} . \hat{y}^{(i)})$$

Выбор типа функции потерь

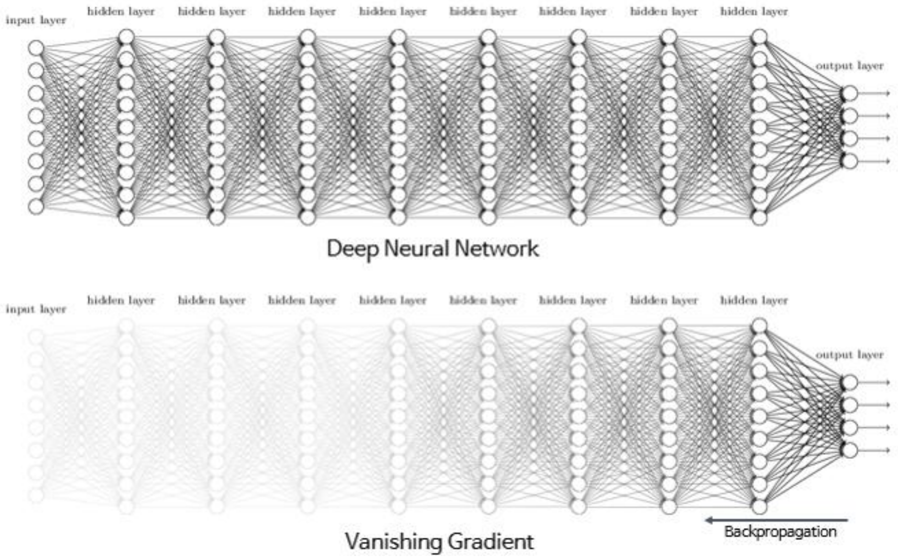
- Задача регрессии
  - функция активации в выходном слое: линейная
  - функция потерь: MSE или Cross Entropy

- Задача классификации
  - функция активации в выходном слое: сигмоида или softmax
  - функция потерь: Cross Entropy или Правдоподобие

Затухающий градиент

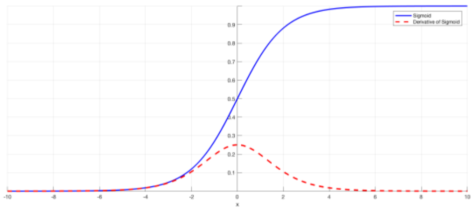
Значения градиента функции потерь близки к 0.

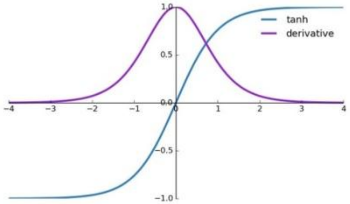
- Малые значения производных функции активации.
- Глубина НС.

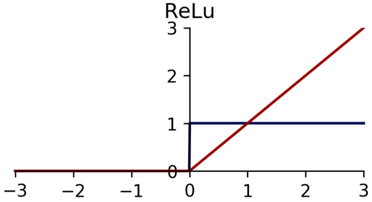


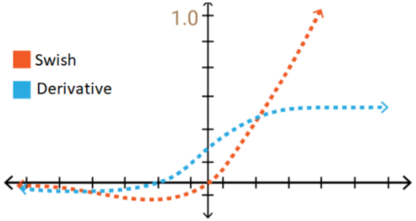
Функция активации

Activation	Function	Derivative	Min	Max
Logistic	$\sigma(z) = \frac{1}{1+e^{-z}}$	$\sigma(z)(1-\sigma(z))$	y $\to 0$	0.25
Hyperbolic Tangent	$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$	$1-(\tanh(z))^2$	y $\to 0$	1
ReLU	$\max(0, z)$	$\begin{cases} 0, & z < 0 \\ 1, & z \geq 0 \end{cases}$	0	1
Leaky ReLU	$\max(0.2z, z)$	$\begin{cases} 0.2, & z < 0 \\ 1, & z \geq 0 \end{cases}$	0	1
Swish	$f(z) = z \cdot \sigma(z)$	$f(z) + \sigma(z)(1 - f(z))$	≈ -0.099	≈ 1.0998







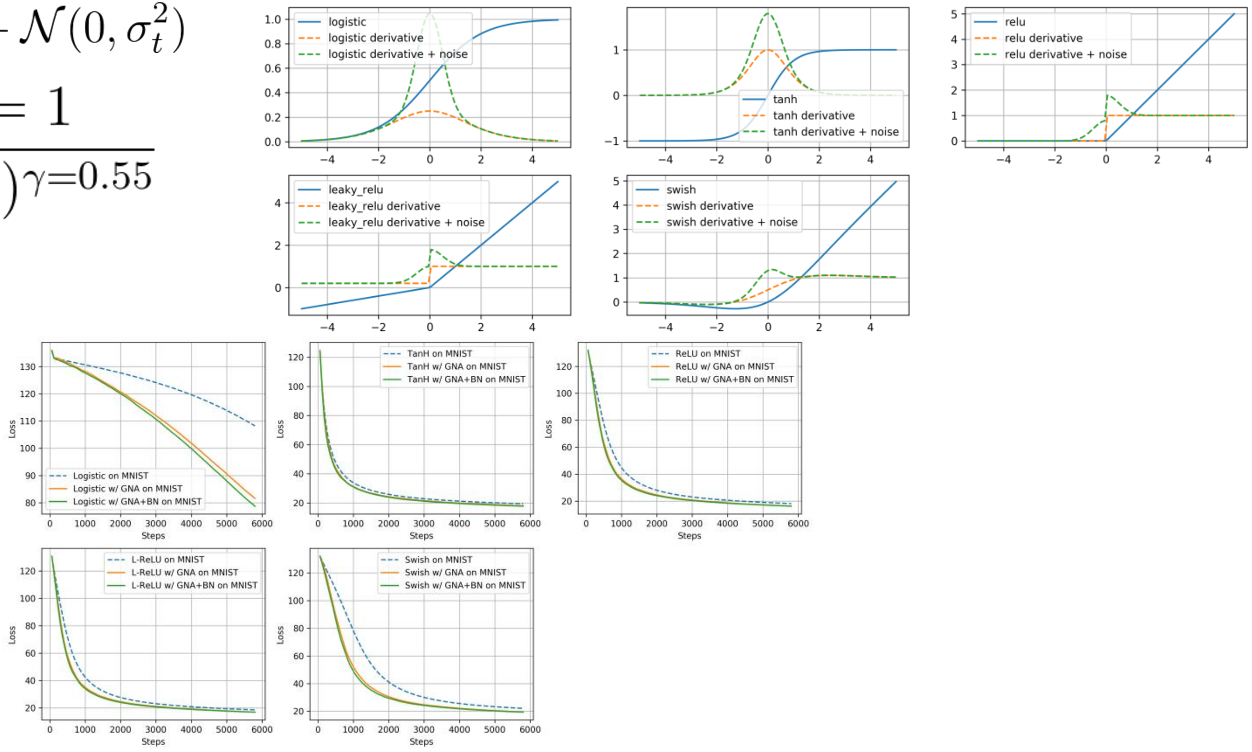


❑ Применение активационных функций с большим диапазоном значений производной (ReLU, Swish, варианты ReLU) способствует решению проблемы затухающего градиента

Добавление «шума» в значение градиента

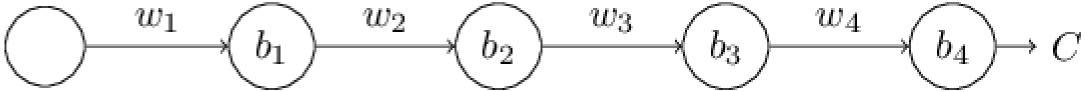
$$\nabla_{\theta_t} J := \nabla_{\theta_t} J + \mathcal{N}(0, \sigma_t^2)$$

$$\sigma_t^2 := \frac{\eta = 1}{(1+t)^{\gamma = 0.55}}$$



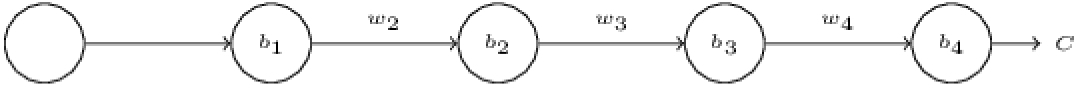
Глубина НС

- Глубокая НС → малый градиент на начальных слоях.
- 4 слоя с одним нейроном в каждом слое



- Градиент на первом слое

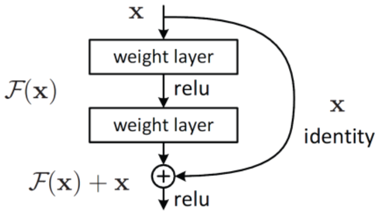
$$\frac{\partial C}{\partial b_1} = \sigma'(z_1) \times w_2 \times \sigma'(z_2) \times w_3 \times \sigma'(z_3) \times w_4 \times \sigma'(z_4) \times \frac{\partial C}{\partial a_4}$$



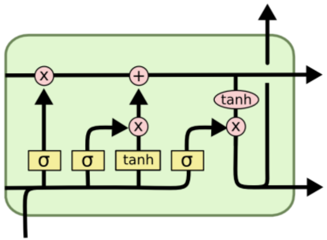
- При инициализации весов нормальным распределением с небольшим std → большинство $|w_j|<1$, $\max(\sigma'(z))=0.25$ → $|w_j \times \sigma'(z))|<0.25$ → $\partial C/\partial b_1 << \partial C/\partial b_3$

«Восстановление» затухающего градиента

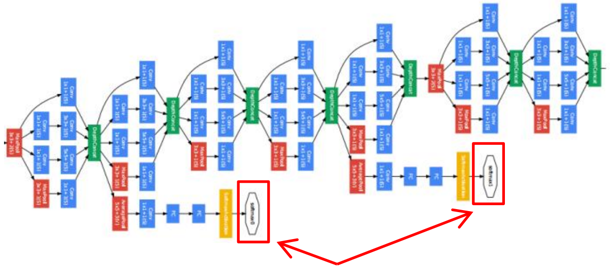
- Изменения в архитектуре НС:
  - Введение параллельных связей: обратное распространение градиента без умножения на производную функции активации: ResNet (остаточные связи), GRU и LSTM (передачи вектора состояния).
  - Использование дополнительных выходов сети на этапе обучения: GoogLeNet



Блок ResNet с остаточной связью



LSTM Нейрон



Фрагмент GoogLeNet