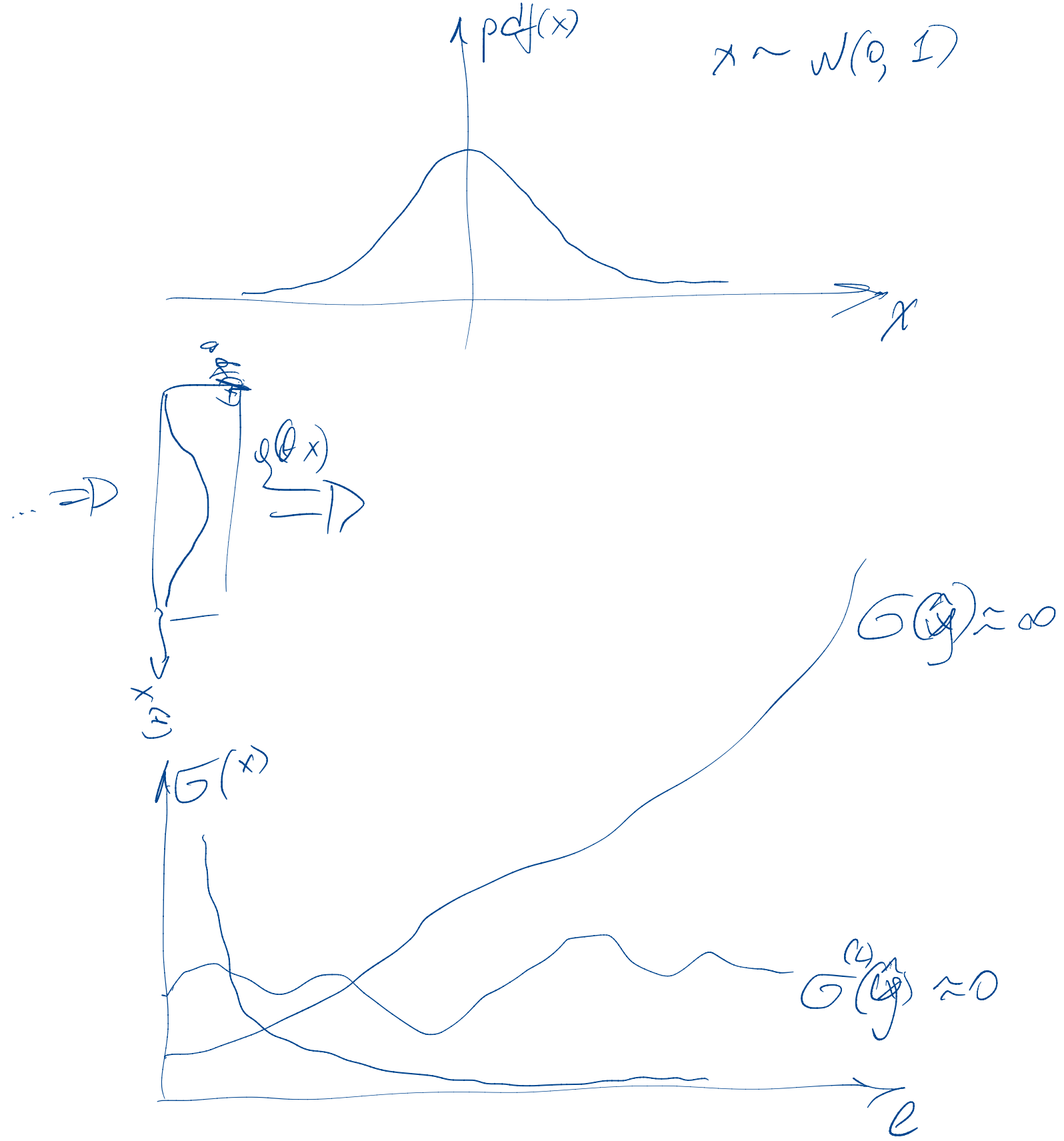

pdf(x)
x ~ N(0, I)
x
...
g(θ, x)
x (z)
σ(x)
σ(x) ≈ ∞
σ(x) ≈ 0
e

Batch Normalization

Weights Initialization

## Инициализация для полносвязных ИНС

$Var\,\theta = \frac{2}{m+k}$

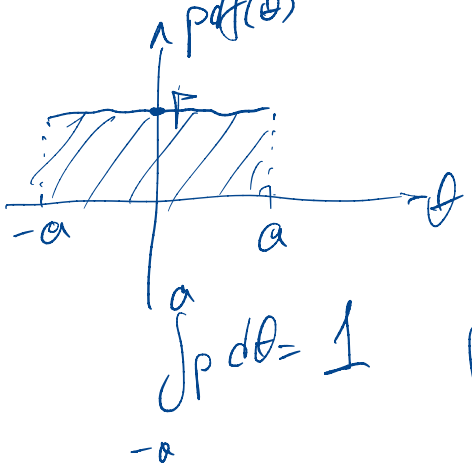
$m$ – кол-во входов слоя

$k$ – кол-во выходов слоя

$\theta \sim U[-a; a]$ $\quad a - ?$

$\theta \sim N(0, \sigma^2)$ $\quad \sigma^2 - ? = \frac{2}{m+k}$

$$\sigma^2(\theta) = \int_{-a}^{a} (\theta - \bar{\theta})^2 \cdot pdf(\theta)\, d\theta = f(a) = \sigma^2(\theta) = \frac{2}{m+k}$$

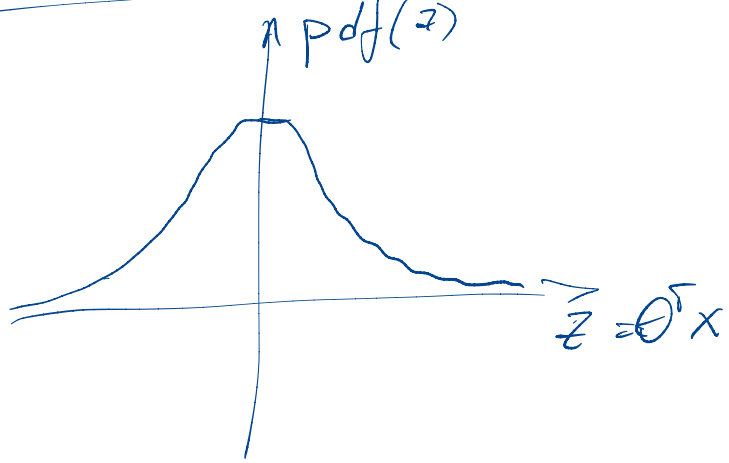


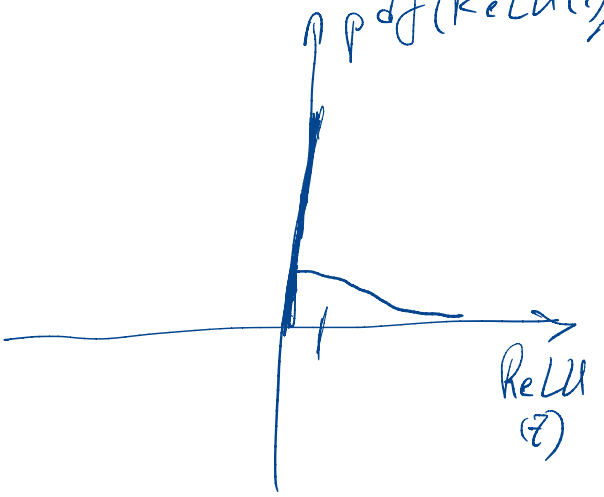

$$a = \frac{\sqrt{6}}{\sqrt{m+k}}$$

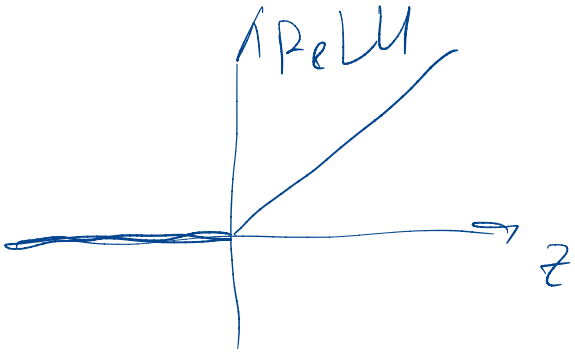
$$\int_{-a}^{a} p\, d\theta = 1 \qquad p \int_{-a}^{a} d\theta = 1 \qquad 2a \cdot p = 1 \qquad p = \frac{1}{2a}$$

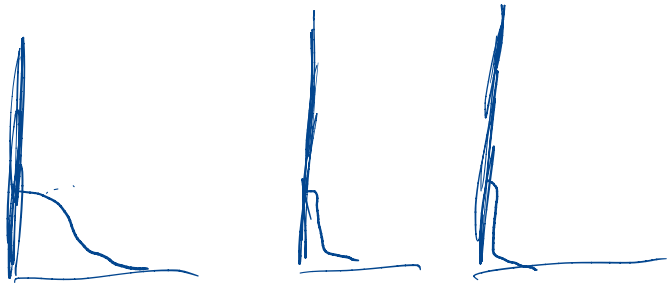
Xavier Glorot

# Batch Normalization



$$z^{(\ell)} = \theta^T h^{(\ell-1)}$$

$$z^* = \frac{z - \mu_B(z)}{\sigma_B(z)}$$

$$\mu_t^* = \mu_{t-1}^* \cdot \beta + \mathbb{E}_{B_t}(z)$$

$$\sigma_t^* = \sigma_{t-1}^* \cdot \alpha + \sigma_{B_t}(z)$$

$$z^* = \frac{z - \mu_t^*(z)}{\sigma_t^*(z) + \varepsilon} \cdot \gamma + \xi \approx 0$$

($\gamma \approx 1$)

# Batch Norm

## Pros

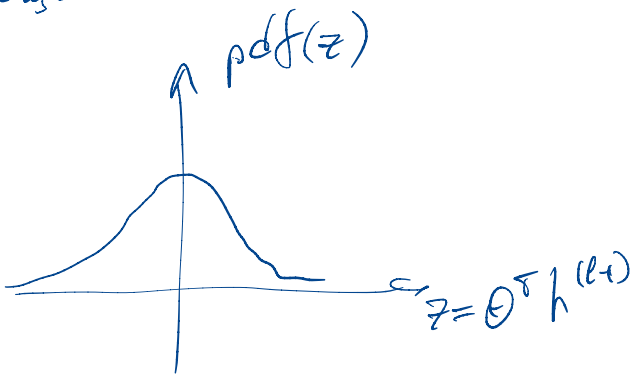
1. Ускоряет сходимость оптимизации
2. Регуляризация не так сильно зависит от $\theta_0$
3. Сглаживает эффективный ландшафт функции потерь
4. Работает как регуляризация.

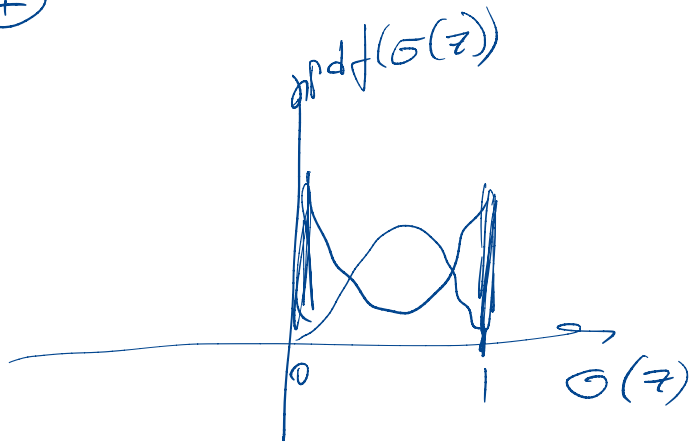
## Cons

1. Плохо работает при небольших batch_size
2. Повышает потребление памяти

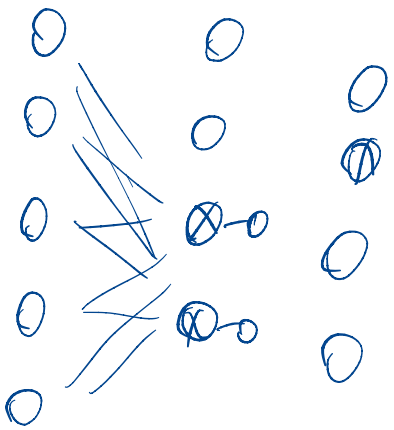
# Функции активации

| | ReLU | $\sigma$ | tanh |
|---|---|---|---|
| смещает $\mu$ | + | + | − |
| смещает $\sigma$ | + | + | + |




## ReLU:

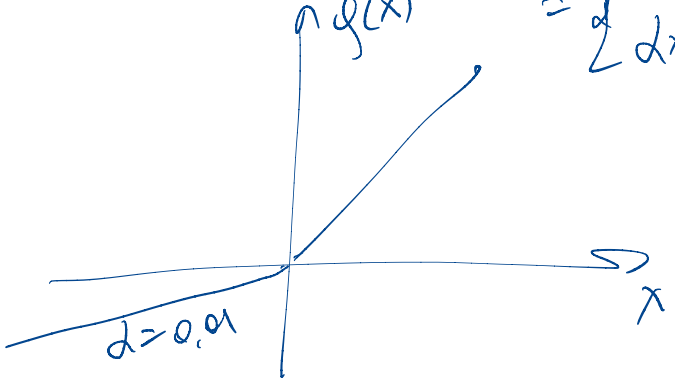


dead neurons

## Leaky ReLU

$g(x)$

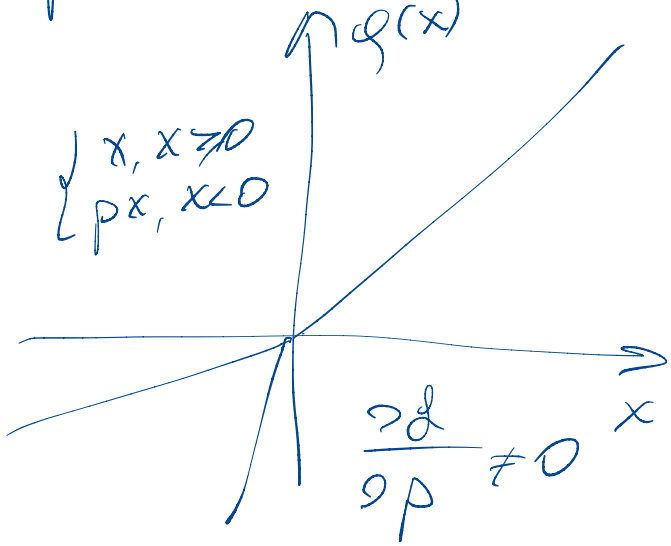
$= \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}$




## parametric ReLU

$g(x)$

$\begin{cases} x, & x \geq 0 \\ px, & x < 0 \end{cases}$



$$ELU(x) = \begin{cases} x, & x \geq 0 \\ \alpha(e^x - 1), & x < 0 \end{cases}$$

Если $x \sim \mathcal{N}(0, 1)$

$\theta_0 \sim \mathcal{N}\left(0, \frac{\sqrt{6}}{\sqrt{m+h}}\right)$

то $\mu(ELU(x)) \approx \mu(x)$

При применении $ELU$ $\mu(ELU(x))$ смещается по ходу обучения не так быстро, как у $ReLU$ или $Leaky\ ReLU$