DEEP LEARNING

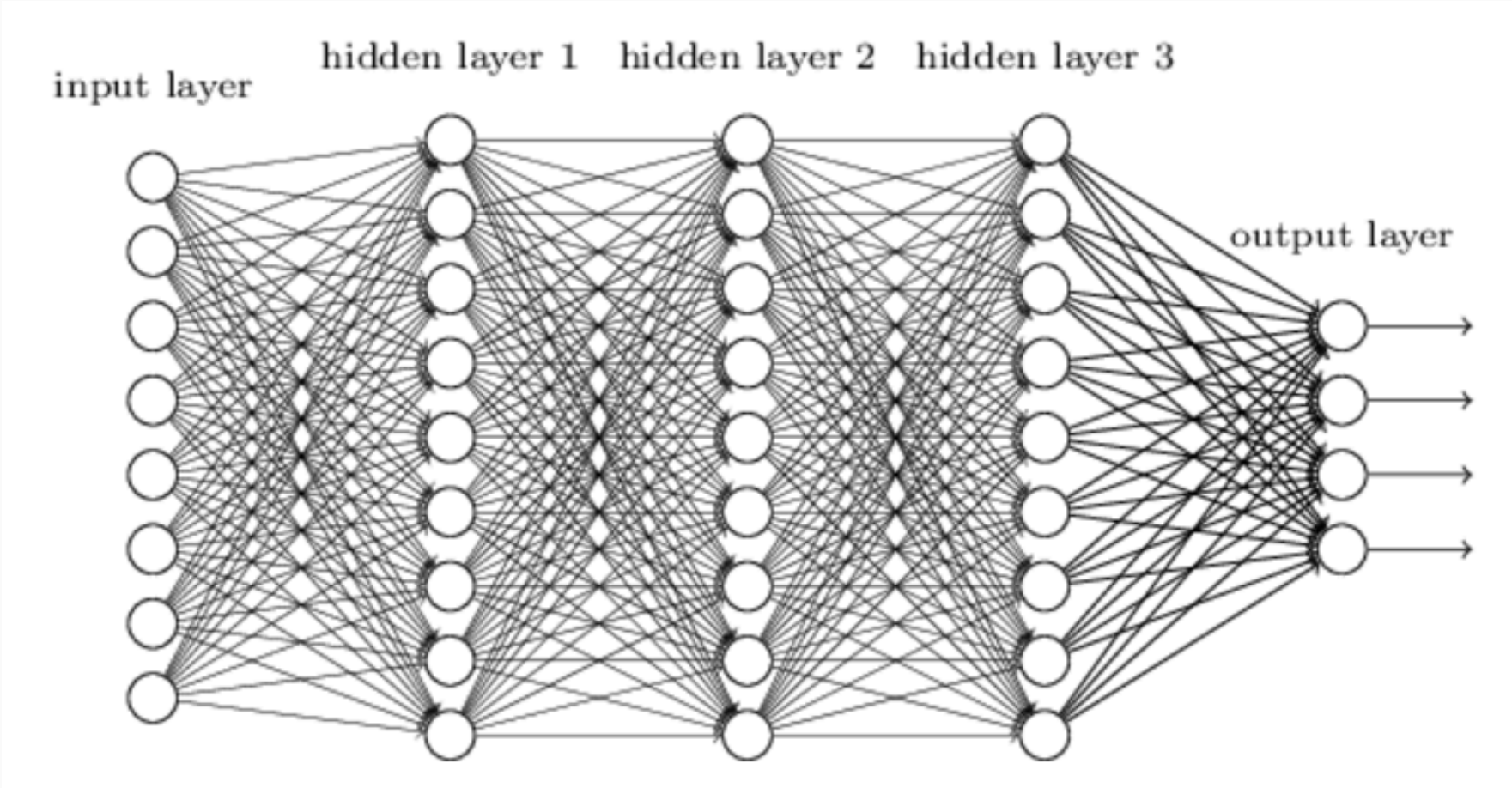
Свёрточные сети

Борис Коваленко, Святослав Елизаров, Артём Грачёв

25 ноября 2017

Высшая школа экономики

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ



Как инициализировать параметры сети?

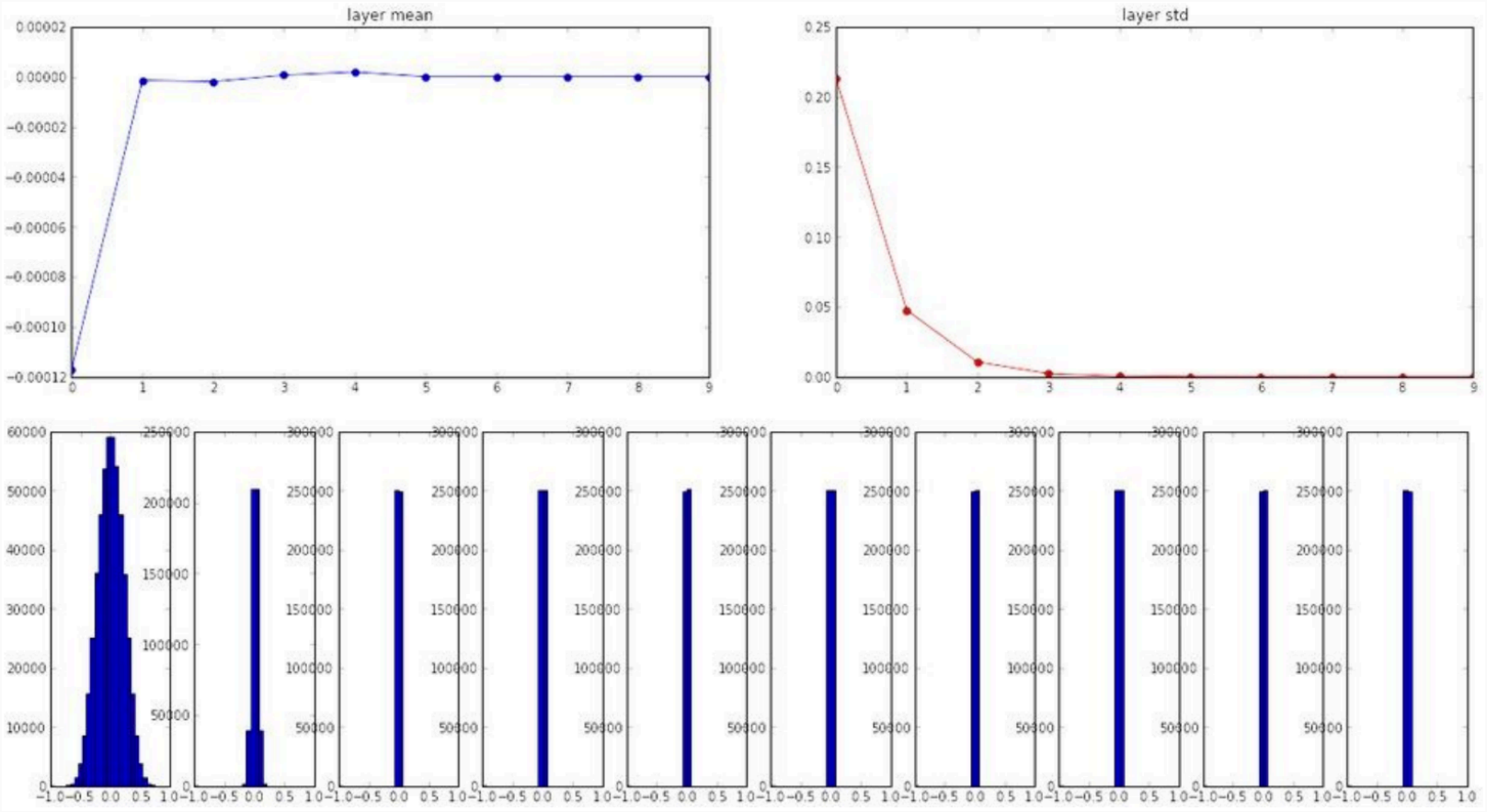
ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

- Инициализация всех весов 0. Сработает?

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

- Инициализация всех весов 0. Сработает?
- Инициализация маленькими случайными числами $W_i \sim N(0, 0.01)$

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

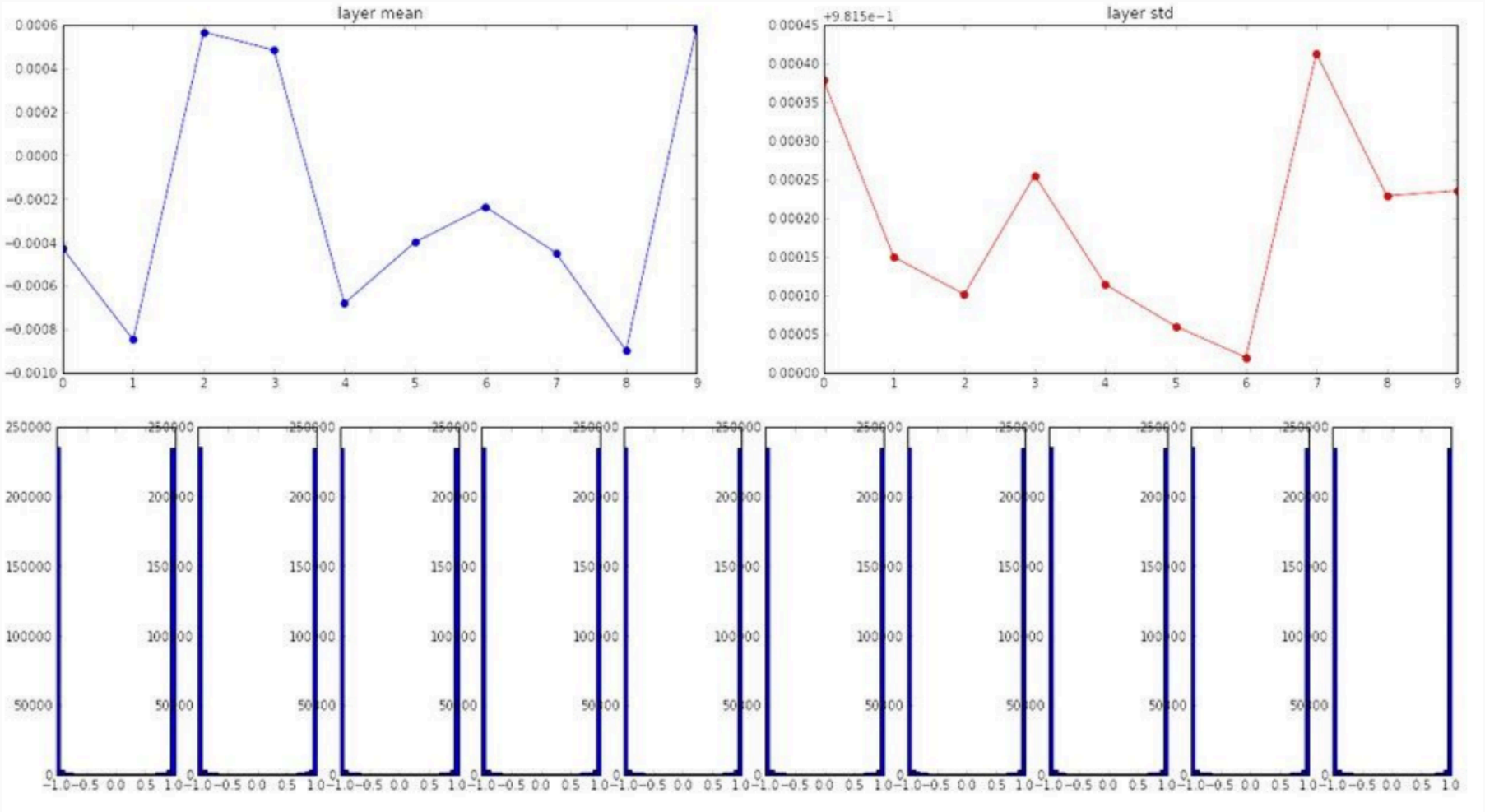


Сеть с 10 слоями, функция активации - tanh, 500 нейронов в каждом слое

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

- Инициализация всех весов 0. Сработает?
- Инициализация ~~маленькими~~ случайными числами $W_i \sim N(0, 1)$

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ



ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

Дисперсия признаков после полносвязного слоя растет с количеством входных признаков. Нужна нормировка, чтобы дисперсия не росла и распределение не ”размазывалось”.

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

Дисперсия признаков после полносвязного слоя растет с количеством входных признаков. Нужна нормировка, чтобы дисперсия не росла и распределение не ”размазывалось”.

Рассмотрим дисперсию для 1 выходного признака:

$$\mathrm{Var}(s) = \mathrm{Var}(\sum_{i}^{n} w_i x_i) = \sum_{i}^{n} \mathrm{Var}(w_i x_i) =$$

$$= \sum_{i}^{n} [E(w_i)]^2 \mathrm{Var}(x_i) + E[(x_i)]^2 \mathrm{Var}(w_i) + \mathrm{Var}(x_i)\mathrm{Var}(w_i) =$$

$$= \sum_{i}^{n} \mathrm{Var}(x_i)\mathrm{Var}(w_i) =$$

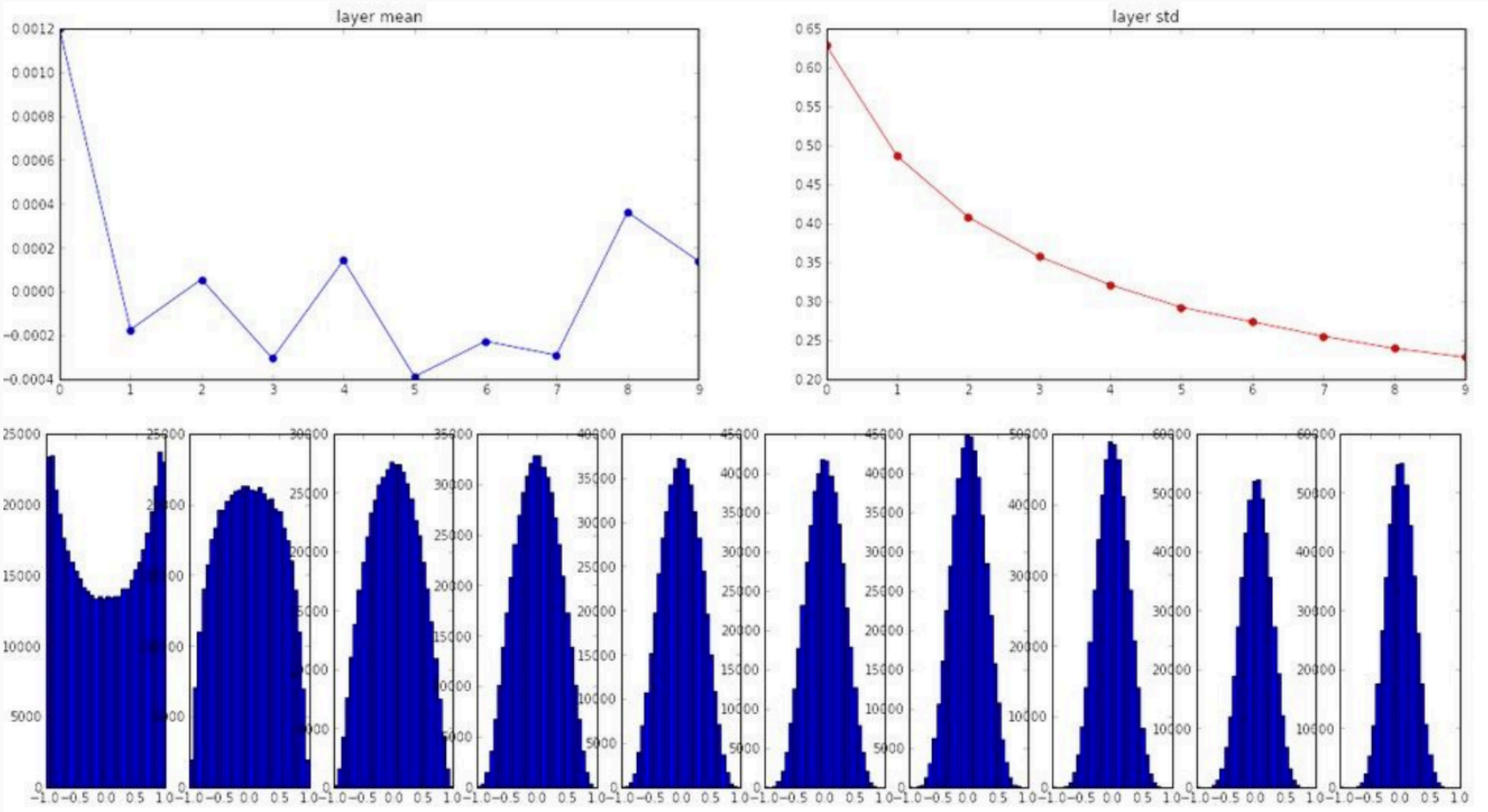
$$= (n\mathrm{Var}(w))\,\mathrm{Var}(x)$$

Чему должна быть равна $\mathrm{Var}(w)$?

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

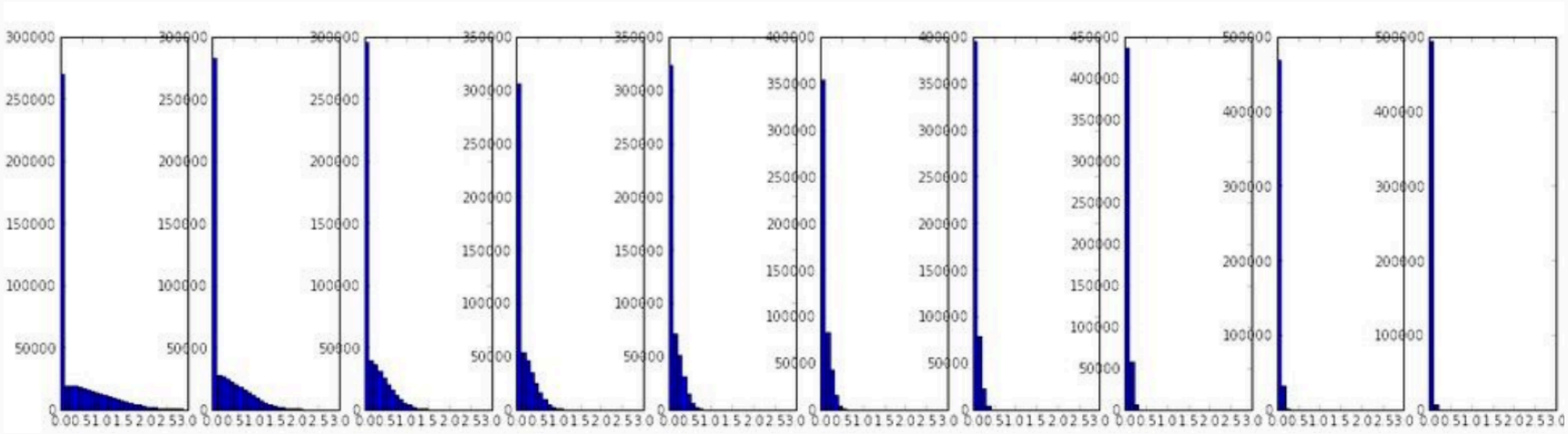
- Инициализация всех весов 0. Сработает?
- Инициализация ~~маленькими~~ случайными числами $W_i \sim N(0, 0.01)$
- Инициализация случайными числами с нормализацией $W_i \sim N(0, \frac{1}{n_{i-1}})$

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ



Xavier initialization

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ



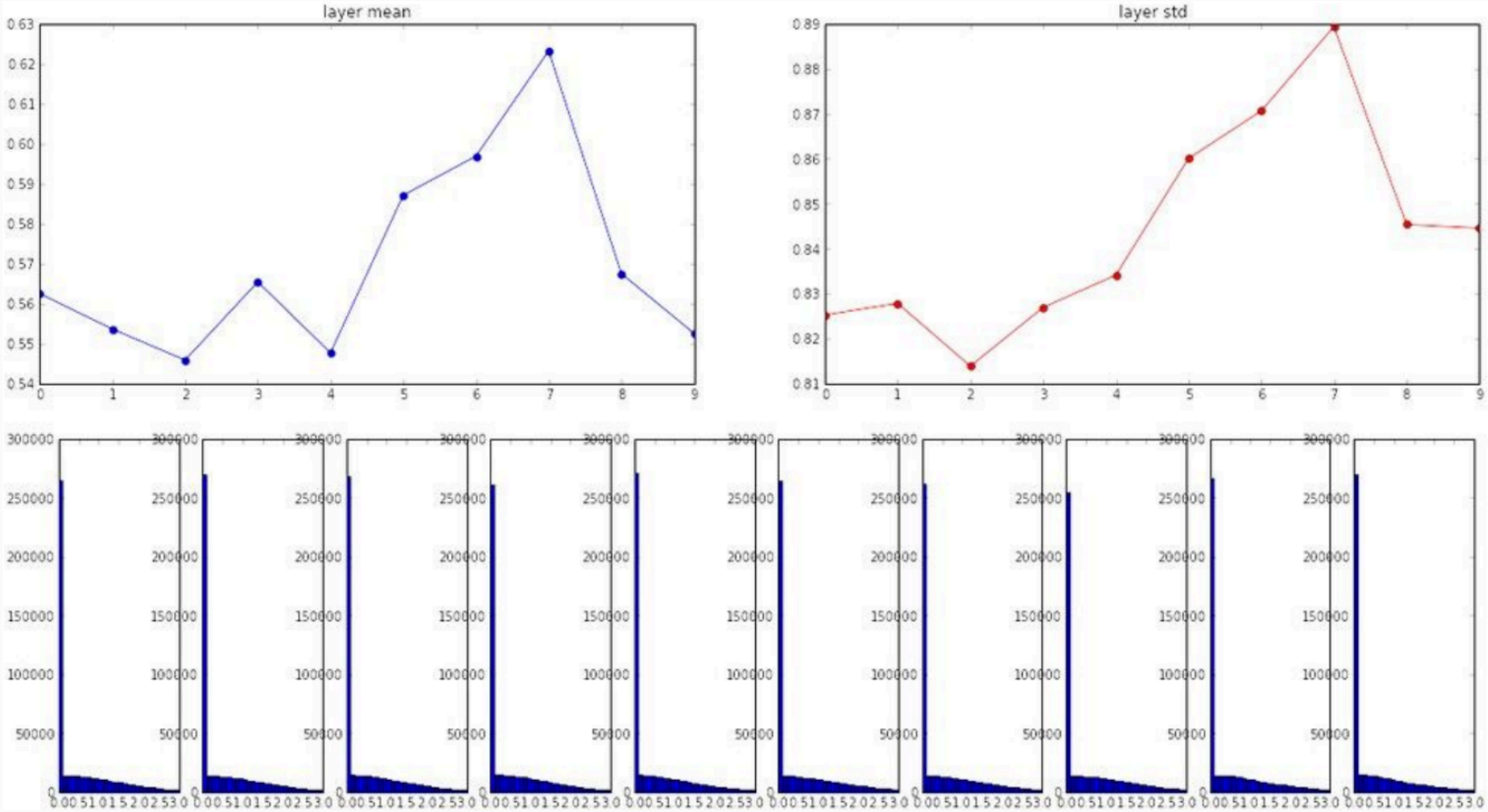
Xavier initialization + ReLU

Glorot & Bengio, AISTATS 2010 - `http://jmlr.org/proceedings/papers/v9/glorot10a/glorot10a.pdf`

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ

- Инициализация всех весов 0. Сработает?
- Инициализация ~~маленькими~~ случайными числами $W_i \sim N(0, 0.01)$
- Инициализация случайными числами с нормализацией $W_i \sim N(0, \frac{1}{n_{i-1}})$
- Инициализация случайными числами с нормализацией для ReLU $W_i \sim N(0, \frac{2}{n_{i-1}})$

ИНИЦИАЛИЗАЦИЯ ПАРАМЕТРОВ



Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification `https://arxiv.org/abs/1502.01852`

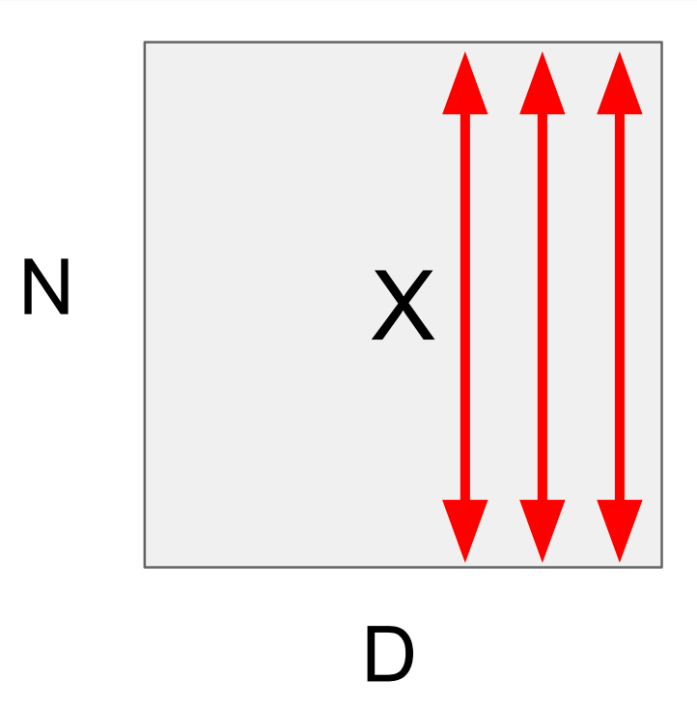
BATCH NORM

Хотелось бы, чтобы гистограмма активаций была ”красивой” - похожа на нормальное распределение. Как этого добиться?

BATCH NORM

Хотелось бы, чтобы гистограмма активаций была ”красивой” - похожа на нормальное распределение. Как этого добиться?

$$\hat{x}^i = \frac{x^i - E[x^i]}{\sqrt{Var(x^i)}}$$



BATCH NORM

Хотим ли мы иметь гистограммы активации с $\mu = 0, \sigma = 1$?

BATCH NORM

Хотим ли мы иметь гистограммы активации с $\mu = 0, \sigma = 1$?

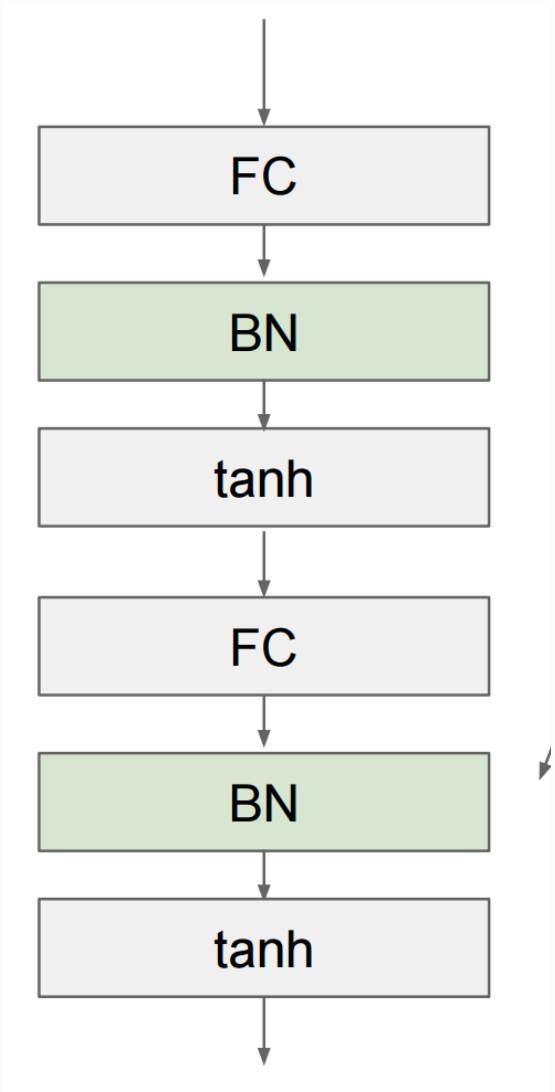
Введем дополнительные параметры:

$$y^i = \zeta^i \hat{x}^i + \beta^i$$

В процессе обучения сети, вектора параметров ζ^i, β^i находятся, как и веса слоев

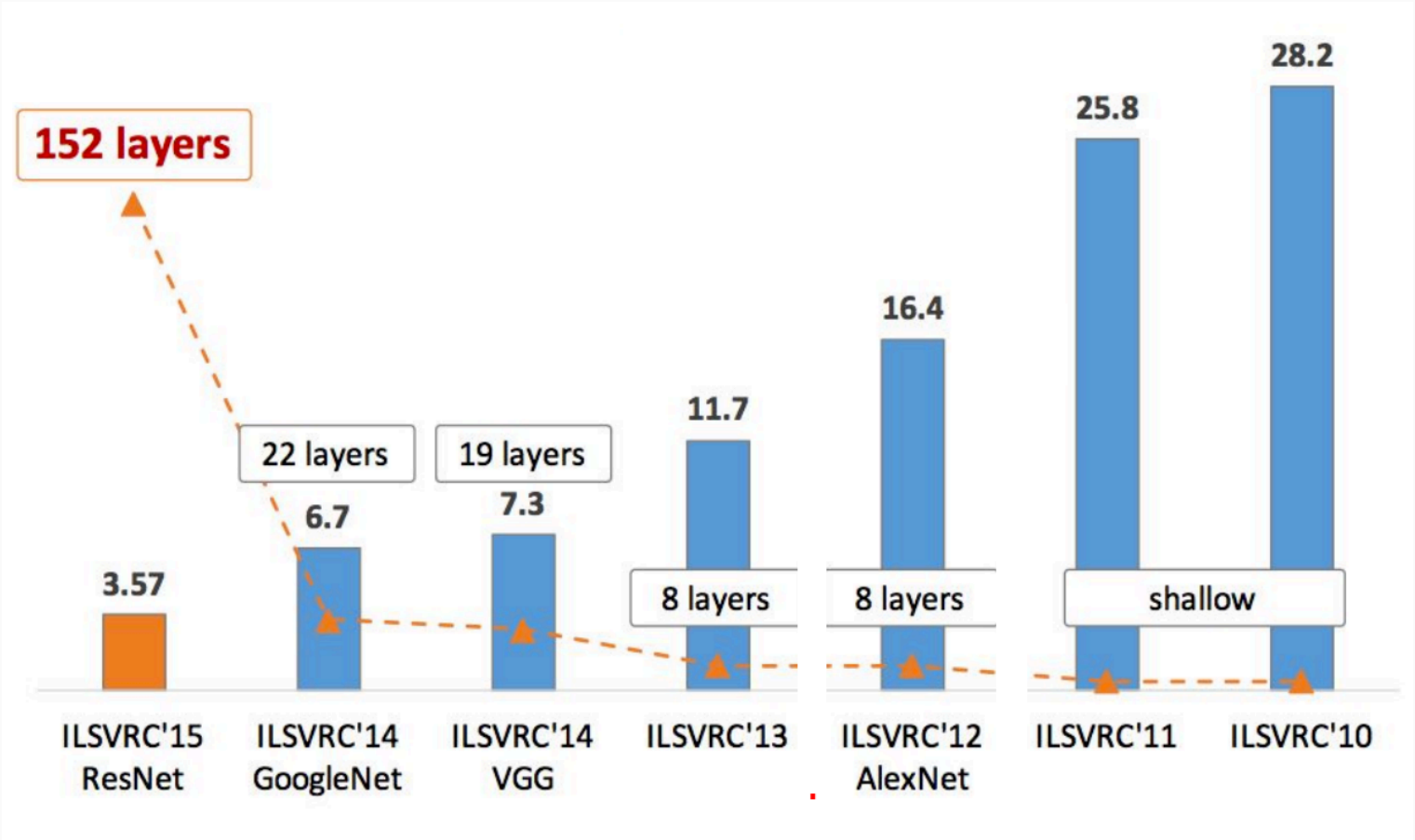
Sergey Ioffe, Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift
`https://arxiv.org/abs/1502.03167`

BATCH NORM



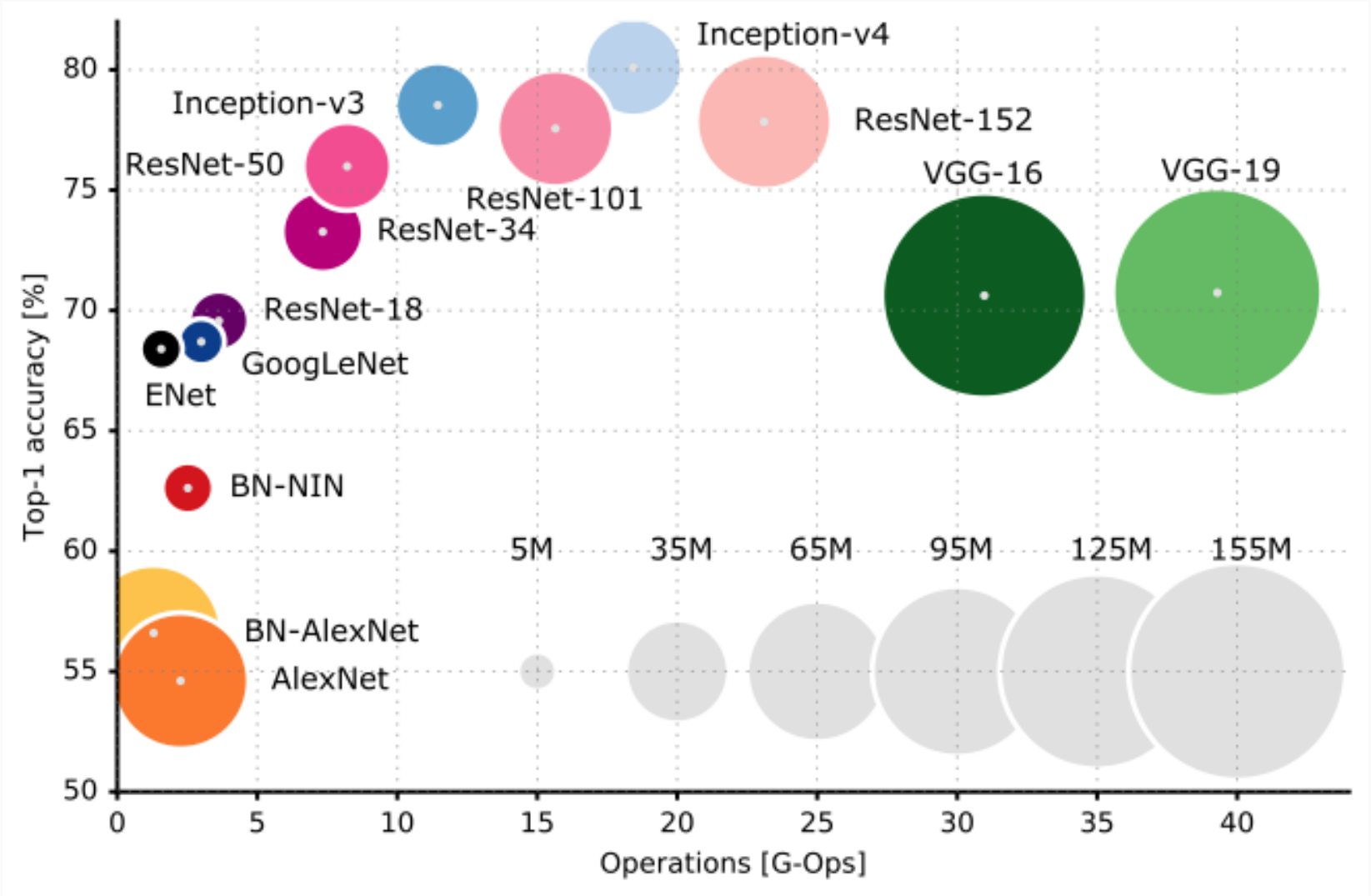
ИСТОРИЧЕСКИЙ ОБЗОР

ImageNet

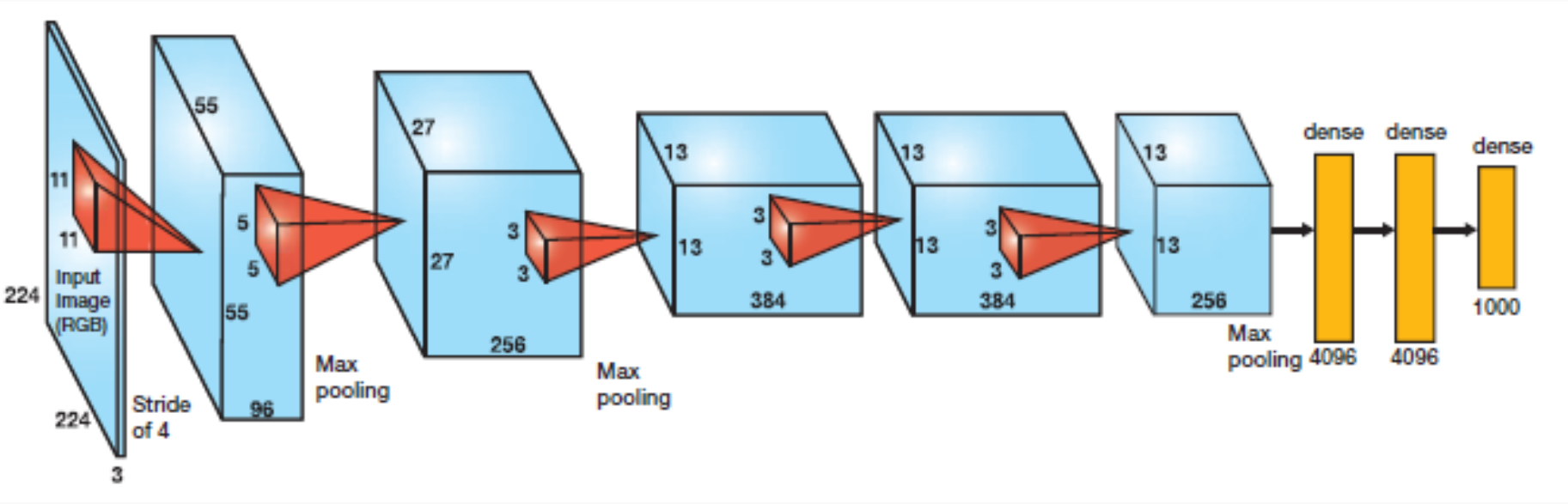


ИСТОРИЧЕСКИЙ ОБЗОР

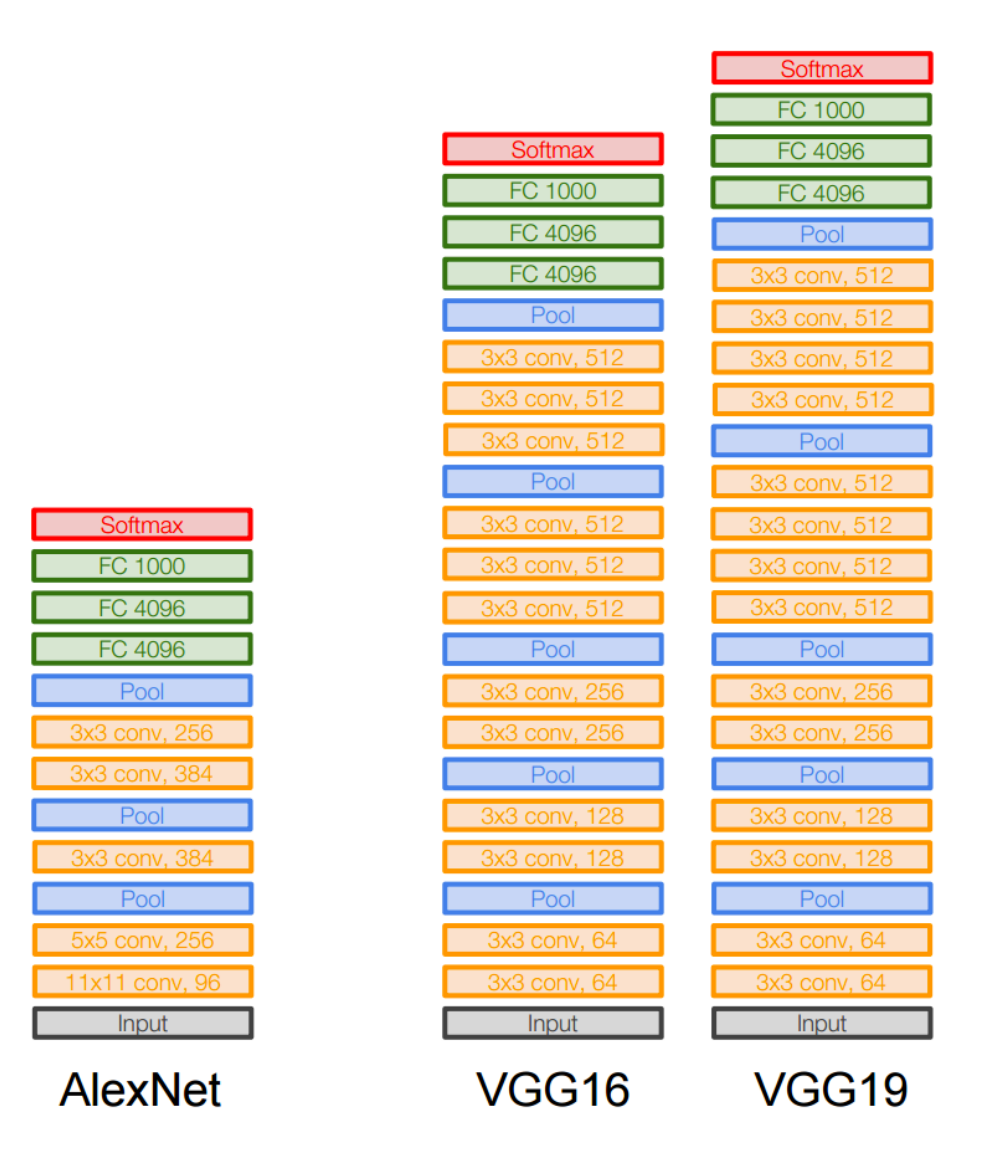
ImageNet



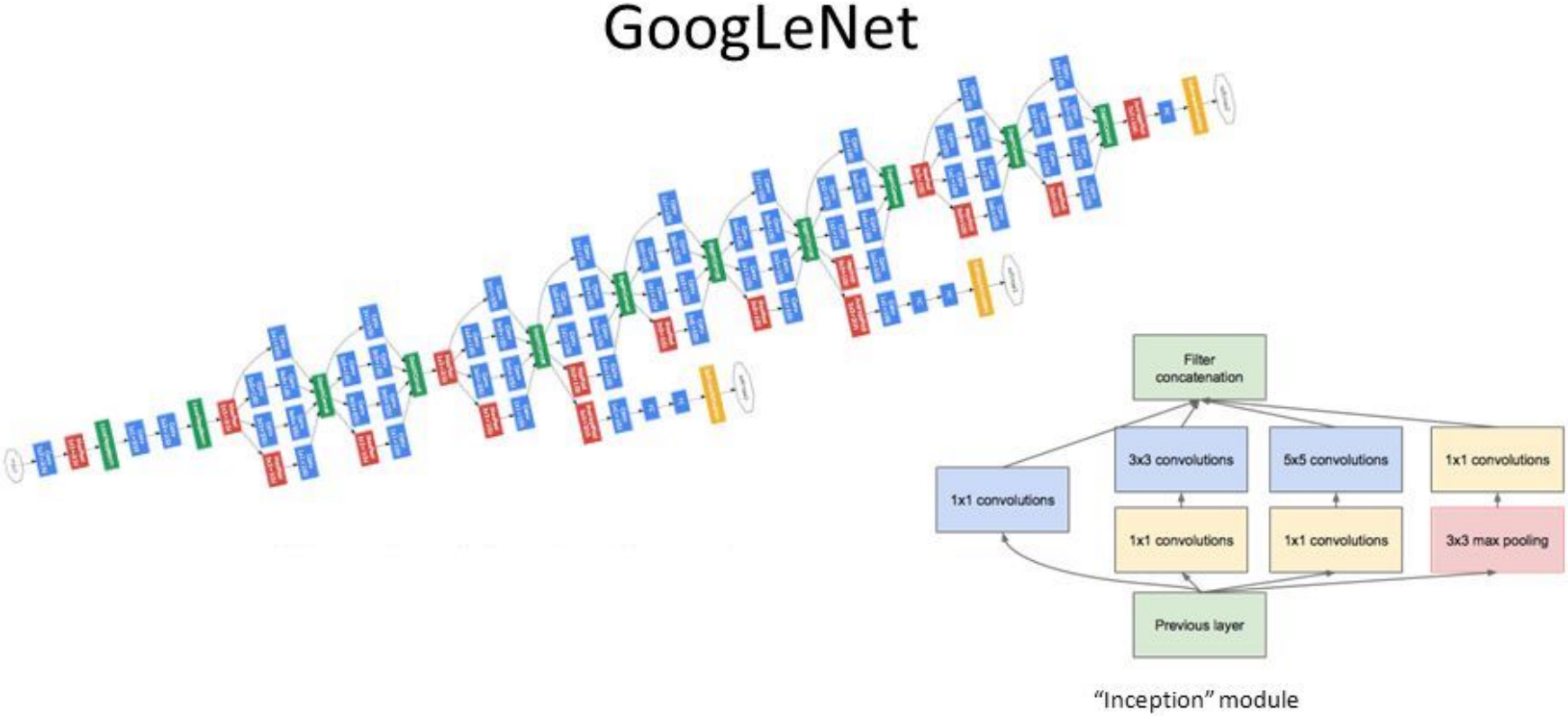
ALEXNET


224
Input
Image
(RGB)
224
3
11
11
Stride
of 4
55
55
96
5
5
Max
pooling
27
27
256
3
3
Max
pooling
13
13
384
3
3
13
13
384
3
3
13
13
256
Max
pooling
dense
4096
dense
4096
dense
1000

VGG


Softmax
FC 1000
FC 4096
FC 4096
Pool
3x3 conv, 256
3x3 conv, 384
Pool
3x3 conv, 384
Pool
5x5 conv, 256
11x11 conv, 96
Input
AlexNet
Softmax
FC 1000
FC 4096
FC 4096
Pool
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
Pool
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
Pool
3x3 conv, 256
3x3 conv, 256
Pool
3x3 conv, 128
3x3 conv, 128
Pool
3x3 conv, 64
3x3 conv, 64
Input
VGG16
Softmax
FC 1000
FC 4096
FC 4096
Pool
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
Pool
3x3 conv, 512
3x3 conv, 512
3x3 conv, 512
Pool
3x3 conv, 256
3x3 conv, 256
Pool
3x3 conv, 128
3x3 conv, 128
Pool
3x3 conv, 64
3x3 conv, 64
Input
VGG19

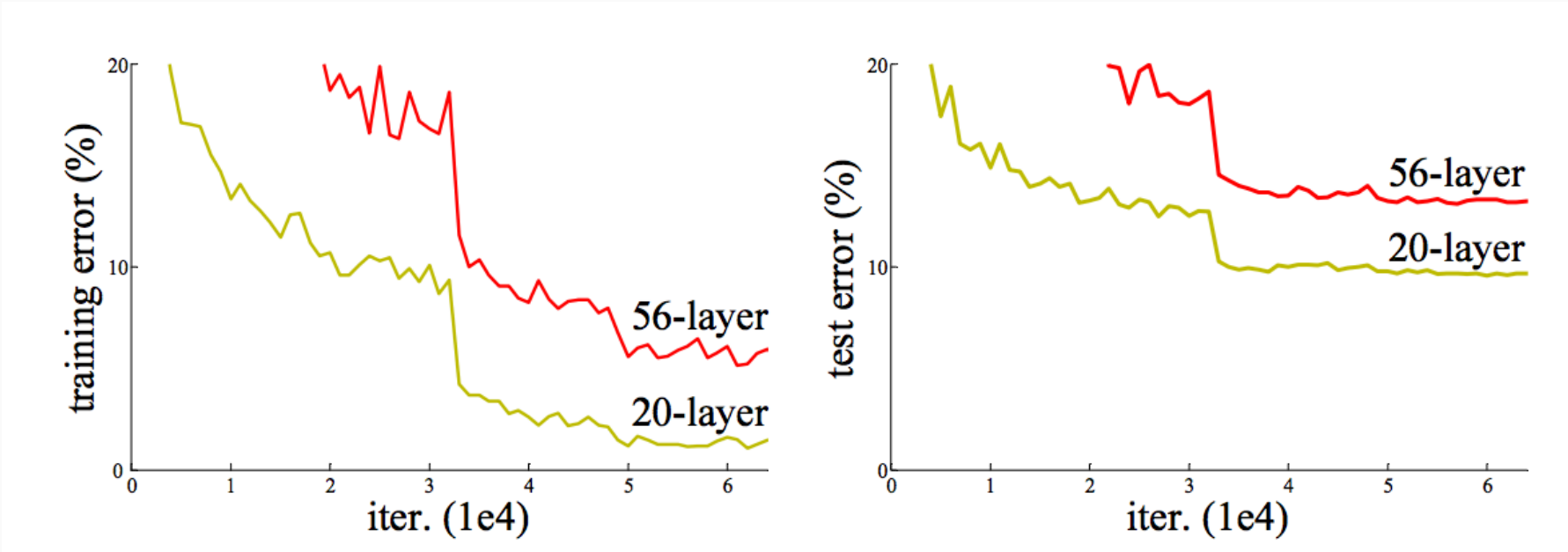
GOOGLENET


GoogLeNet
Filter concatenation
3x3 convolutions
5x5 convolutions
1x1 convolutions
1x1 convolutions
1x1 convolutions
1x1 convolutions
3x3 max pooling
Previous layer
"Inception" module

ГЛУБИНА

- Во многих задачах предпочтительнее использовать более глубокую сеть. Например, чтобы иметь более широкое поле обзора.
- Теоретически доказано, что глубина сети влияет на обобщающую способность. [Eldan, Shamir, 2016 The Power of Depth for Feedforward Neural Networks] и [Matus Telgarsky. 2016 Benefits of depth in neural networks]
- Однако “наивный” способ добавления слоёв может не сработать. **Почему?**

ГЛУБИНА


training error (%)
20
10
0
0
1
2
3
4
5
6
iter. (1e4)
56-layer
20-layer
test error (%)
20
10
0
0
1
2
3
4
5
6
iter. (1e4)
56-layer
20-layer

RESNET

- Если добавленные слои будут отображением вида $F(x) = x$, тогда ошибка не должна превышать ошибку базовой, маленькой сети?

RESNET

- Если добавленные слои будут отображением вида $F(x) = x$, тогда ошибка не должна превышать ошибку базовой, маленькой сети?
- Давайте “пробросим” связь от входа до выхода (shortcut)
- Таким образом слои сети будут учить разницу между входом и выходом.

RESNET

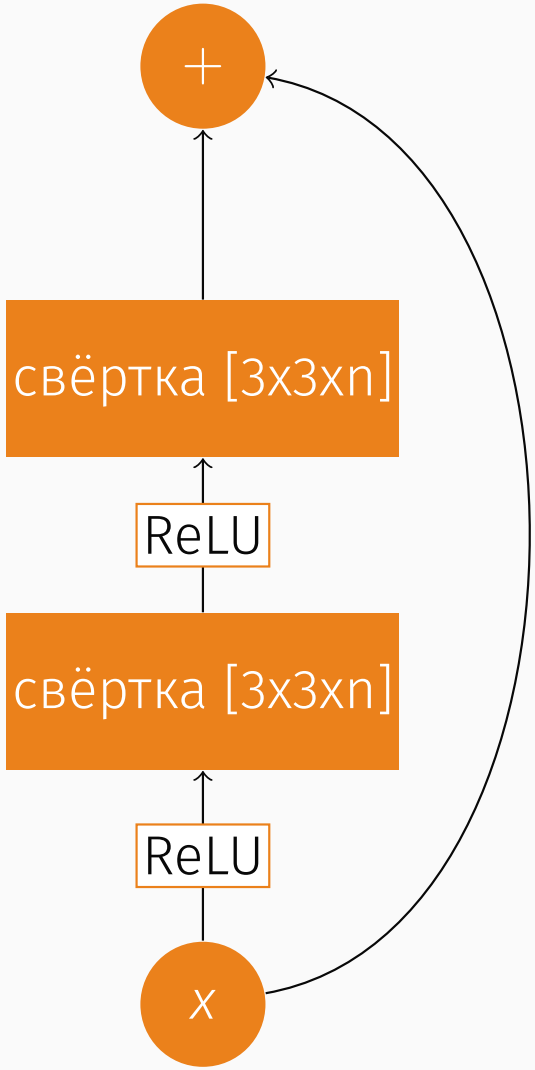
$$\psi(x, W) = \phi(x, W) + x$$

Где x – вход, W – тензор параметров, ϕ – слой или несколько слоёв нейронной сети (например свёрточный слой).

Такой тип архитектуры называется *разностным* (residual network) или просто resnet

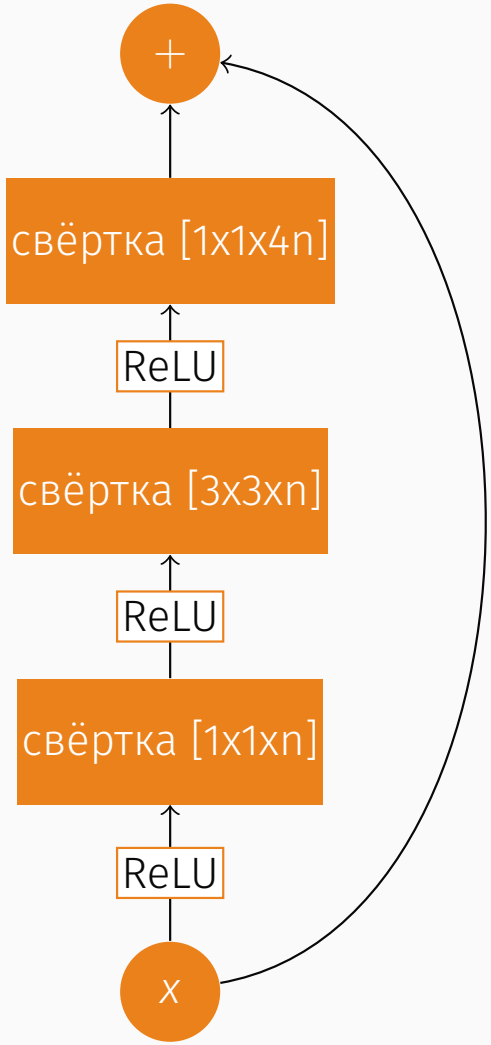
[He et al. 2015 Deep Residual Learning for Image Recognition]

RESNET



Составной базовый блок разностной сети

RESNET



Составной блок ”бутылочное горлышко” разностной сети

RESNET

- Из таких блоков могут быть построены сети с сотнями (и даже тысячами) слоёв
- Теоретически обосновано, что наиболее эффективным является shortcut длины 2. [Le et al. 2017 Demystifying ResNet]

RESNET

Для обучения очень глубоких resnet-сетей используется **метод стохастической глубины** (stochastic depth) описаный в работе [Huang et al. 2016 Deep Networks with Stochastic Depth]

Метод напоминает dropout, однако работает на уровне “блоков” сети. С заданной вероятностью произвольные resnet-блоки заменяются тождественной связью во время тренировки. Вывод, как и в случае с dropout, производится на полностью активированной сети.

HIGHWAY NETWORKS

- Resnet является частным случаем Highway Networks
- Вход не просто прибавляется к выходу, а смешивается с ним в пропорции, которая обучается автоматически.

$$\psi(x, W_{layer}, W_\sigma) = \sigma(x, W_\sigma)\phi(x, W) + (1 - \sigma(x, W_\sigma))x$$

Где σ – функция, значения которой лежат от 0 до 1. Эта функция называется *воротами*.

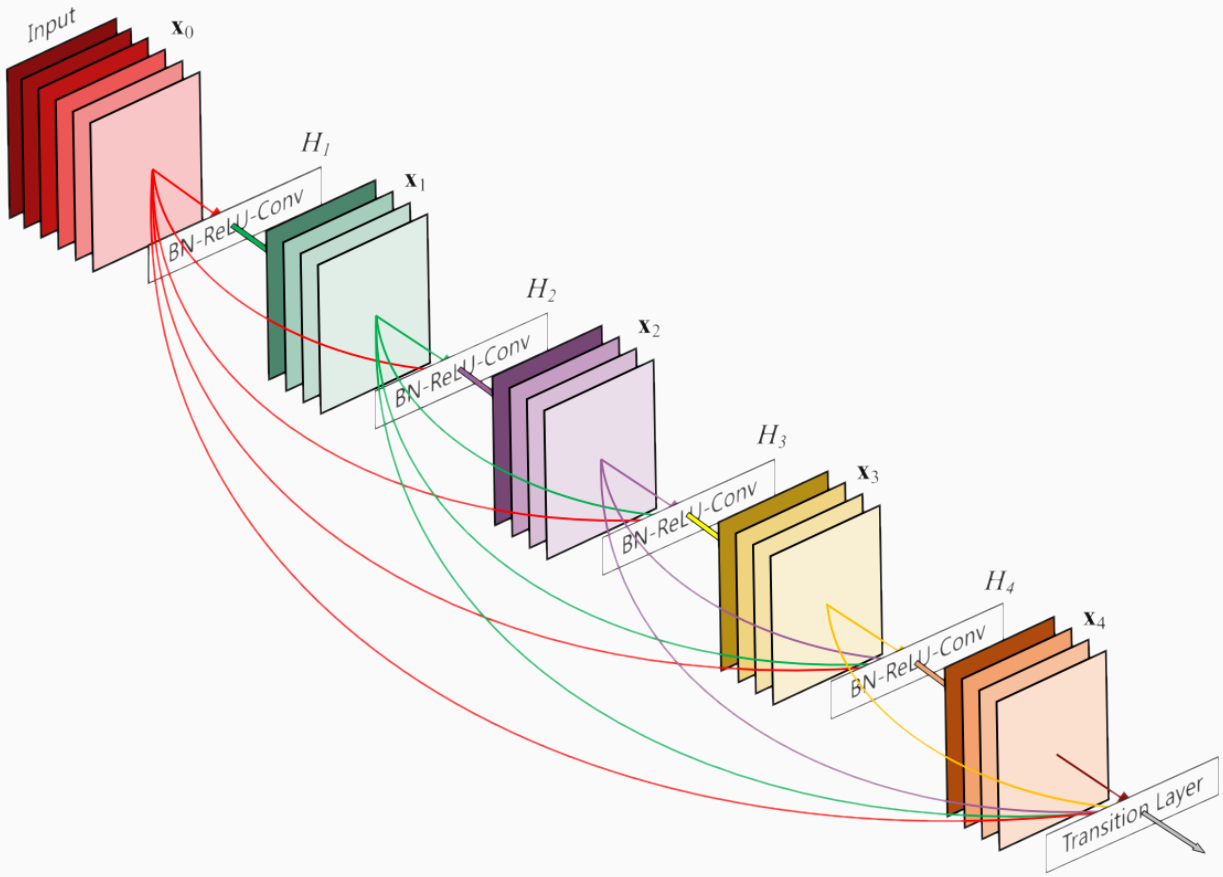
[Srivastava, Greff, Schmidhuber. 2015 Training Very Deep Networks]

DENSE NETWORKS

- Другой интересной архитектурой для тренировки глубоких сетей являются **плотные сети** (dense networks).
- Основная идея заключается в том, что вход каждого слоя связывается со всеми выходами промежуточных слоёв
- В отличие от resnet в связи используется не суммирование, а конкатенация

[Huang, Liu, Weinberger. 2016 Densely Connected Convolutional Networks]

DENSE NETWORKS

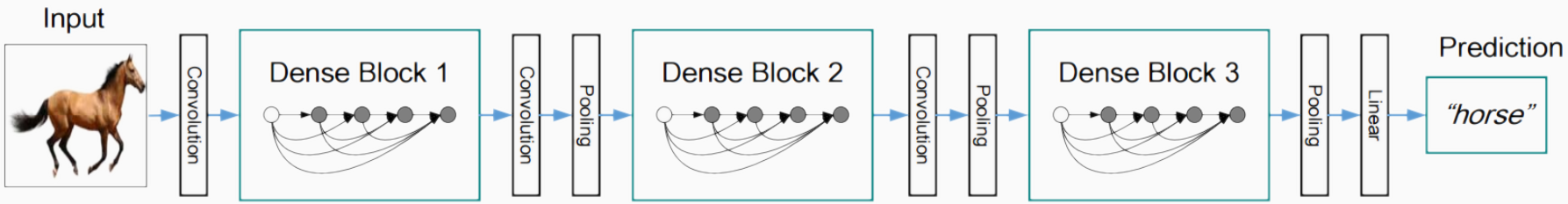


Составной блок плотной сети

DENSE NETWORKS

- Градиент легко передаётся на нижние слои
- Каждый слой может использовать признаки, полученные на предыдущих, что может быть полезно
- Из-за того, что выходы приклеиваются ко входам ширина слоёв растёт.

DENSE NETWORKS



Чтобы сеть не расширялась не используют блоки глубины больше 5