

# Нейросетевые технологии

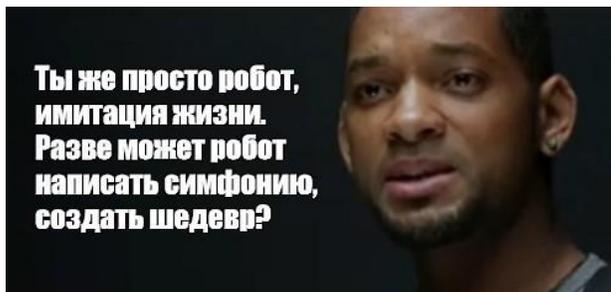
Лекция 1. Базовые понятия нейросетей

# План

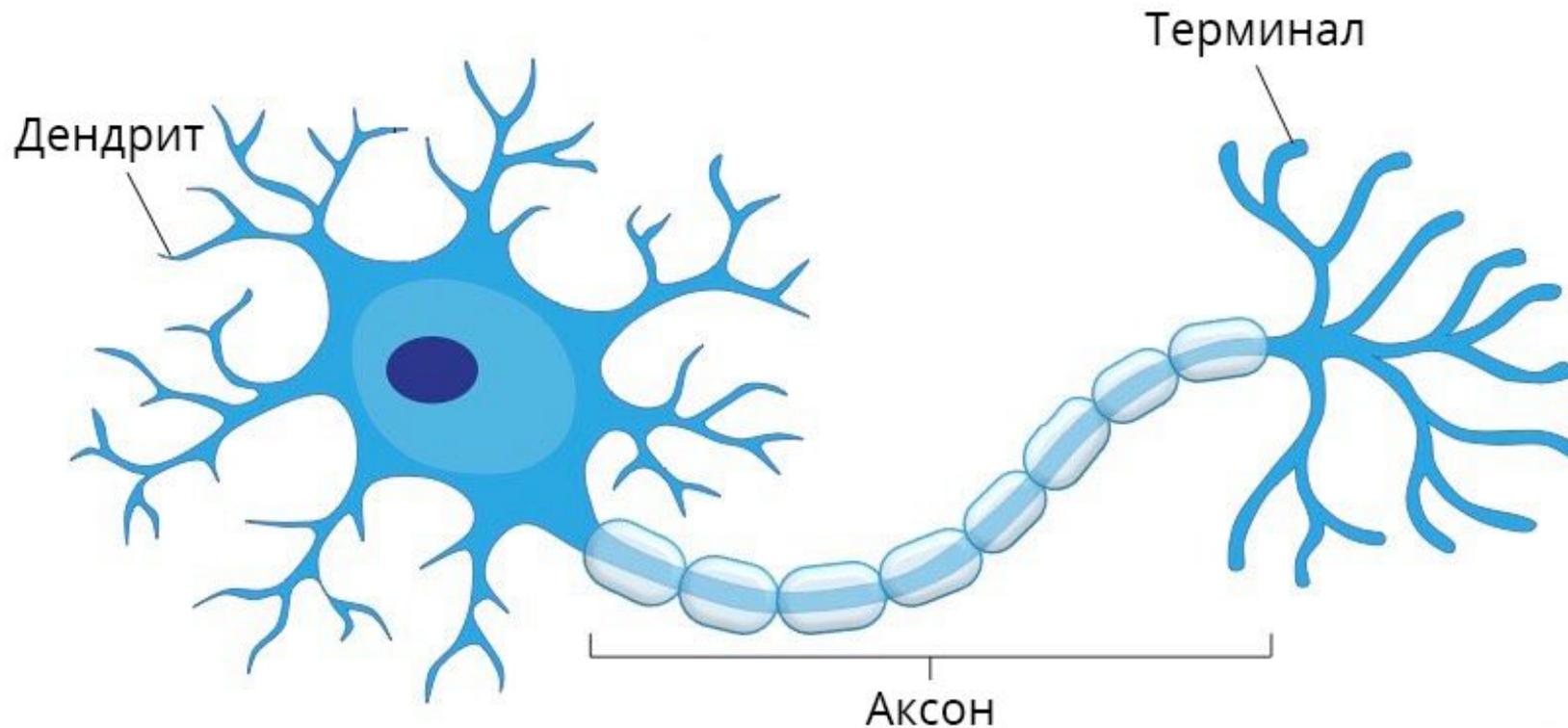
- ▶ Примеры применения нейросетей;
- ▶ Биологический и искусственный нейрон;
- ▶ Разновидности нейросетей;
- ▶ Разбор простейшей нейросети.

# Примеры применения

- ▶ Приложения по улучшению/изменению фотографии: FaceApp, Meitu и другие;
- ▶ Ретушь изображения (Neural Filters) - Adobe Photoshop начиная с версий 2021 года;
- ▶ Генерация изображений: объекты, лица, котики...
- ▶ Генерация видео - алгоритмы компании NVIDIA;
- ▶ Распознавание образов: Tesla и иные беспилотные авто; система опознавания лиц;
- ▶ Анализ «больших данных» BigData: финансовые, научные и иные данные
- ▶ Анализ звуковых данных: Алиса, Siri;
- ▶ Генерация текстов, историй: AI Dungeon, «Порфирьевич» - основанные на нейросети GPT-2 (generative pre-training);
- ▶ Генерация мелодий, голоса
- ▶ Все ссылки в конце презентации.

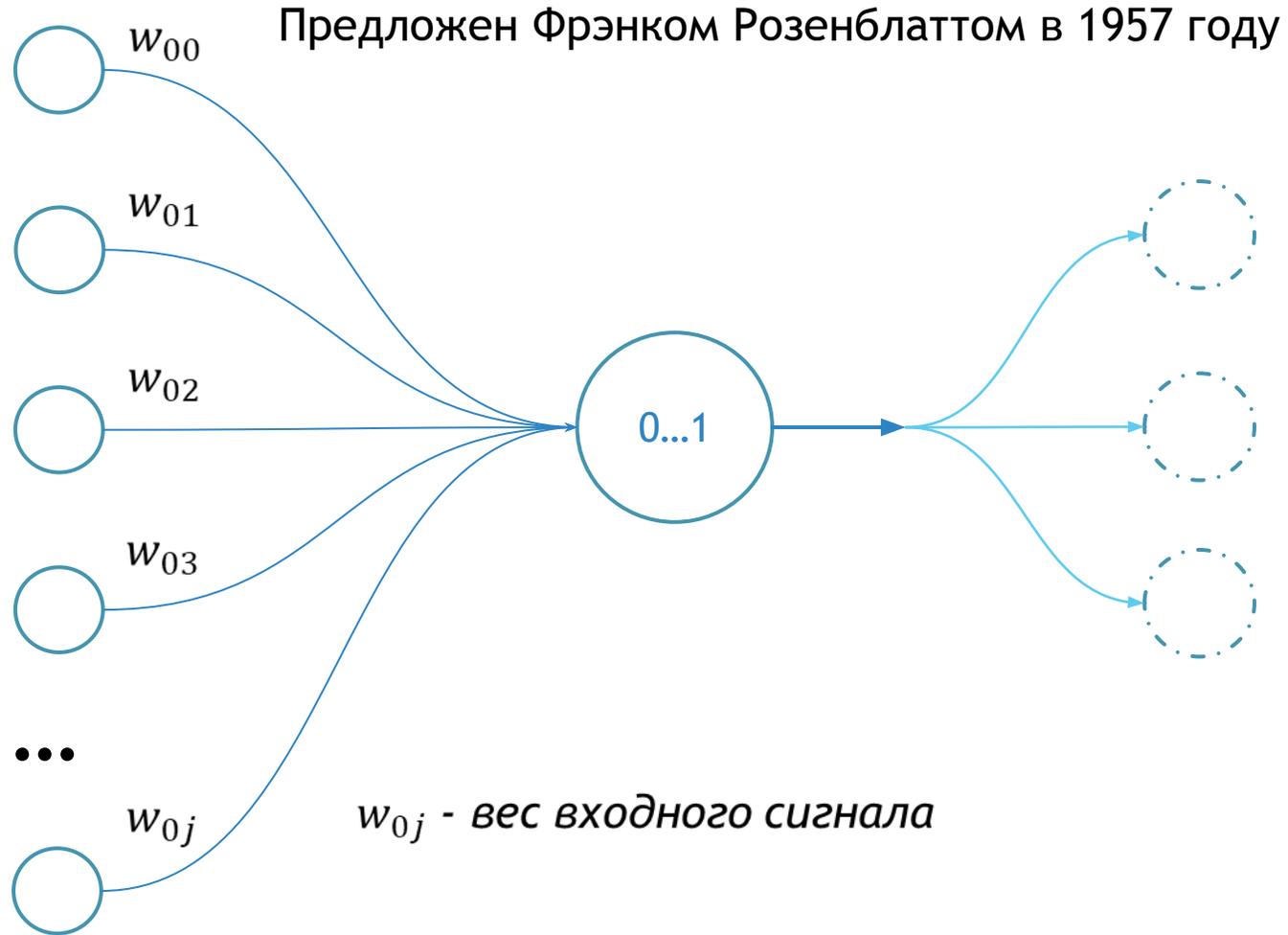


# Биологический нейрон



- Дендриты - отвечают за получение сигналов возбуждения и торможения к нейрону.
- Аксон - передаёт информацию от одного нейрона к другому
- Терминаль - синаптическое окончание, контактирующее с другой клеткой (мишенью). Образует синапс.

# Искусственный нейрон (персептрон)



# Искусственный нейрон (персептрон)

$a^{(0)}$

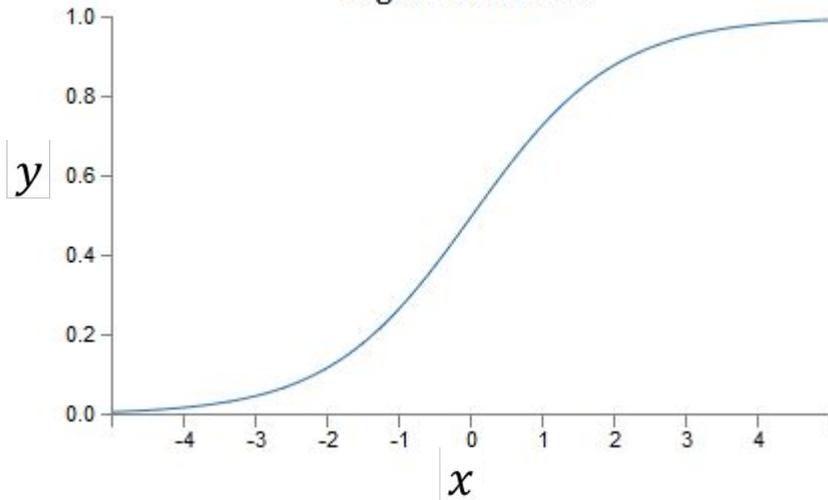
$$a_{\theta_0}^{(1)} = \left( \sum_{j=0}^n w_{0j}^{(0)} * a_j^{(0)} + b \right) \sigma(x)$$

$b$  – [bias] – коэффициент смещения

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

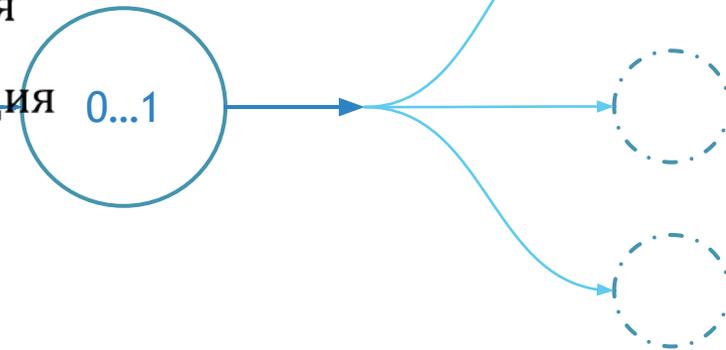
– сигмоидная функция

sigmoid function



входного сигнала

$a^{(1)}$  – первый слой  
 $a_0$  – индекс нейрона



# Искусственный нейрон (персептрон)

$$a_0^{(1)} = \sigma\left(\sum_{j=0}^n w_{0j} * a_j^{(0)} + b\right)$$

$$W^{(0)} = |w_{00} \quad w_{01} \quad w_{02} \quad \dots \quad w_{0j}|$$

$$a^{(0)} = |a_0^{(0)} \quad a_1^{(0)} \quad a_2^{(0)} \quad \dots \quad a_j^{(0)}|$$

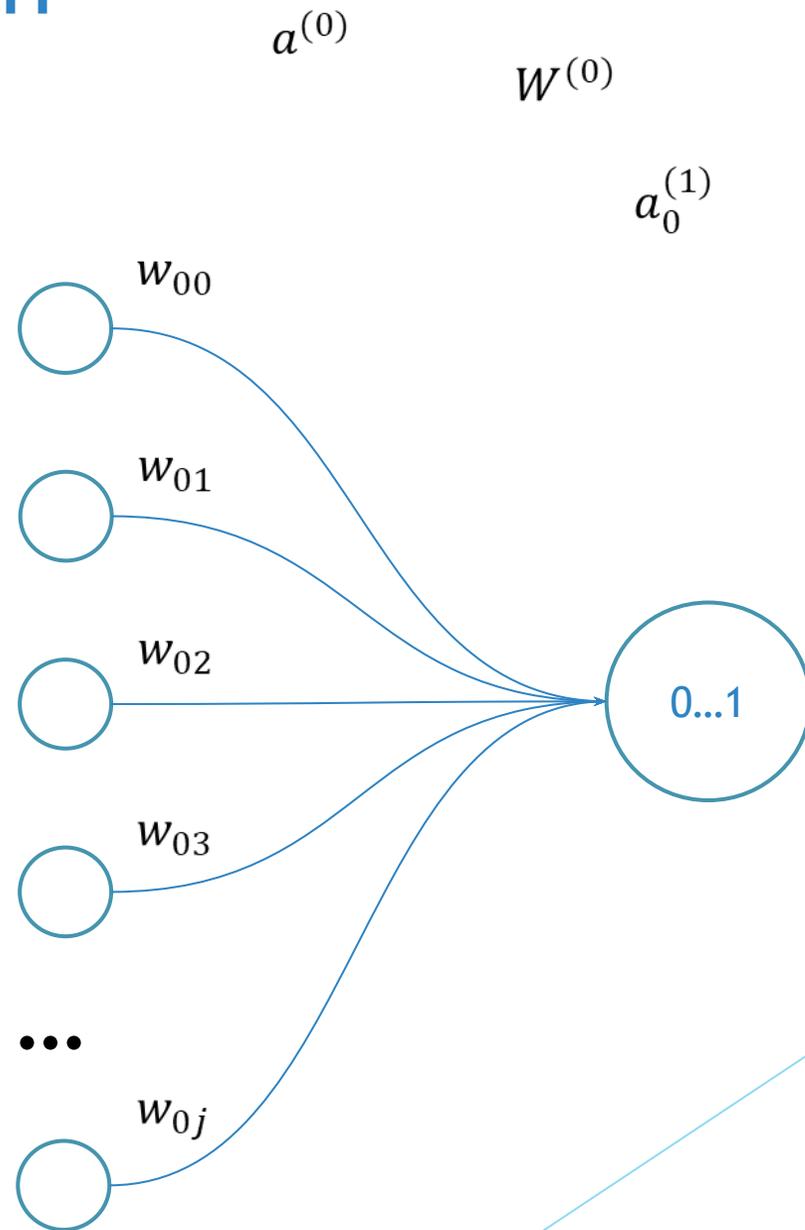
$$b = |b_0|$$

$$a_0^{(1)} = \sigma(W^{(0)} * a^{(0)} + b)$$

$$a_0^{(L)} = \sigma(W^{(L-1)} * a^{(L-1)} + b)$$

$L$  – индекс слоя

$$a^{(L)} = \sigma(W^{(L-1)} * a^{(L-1)} + b)$$



# Виды искусственных нейронных сетей (ИНС)

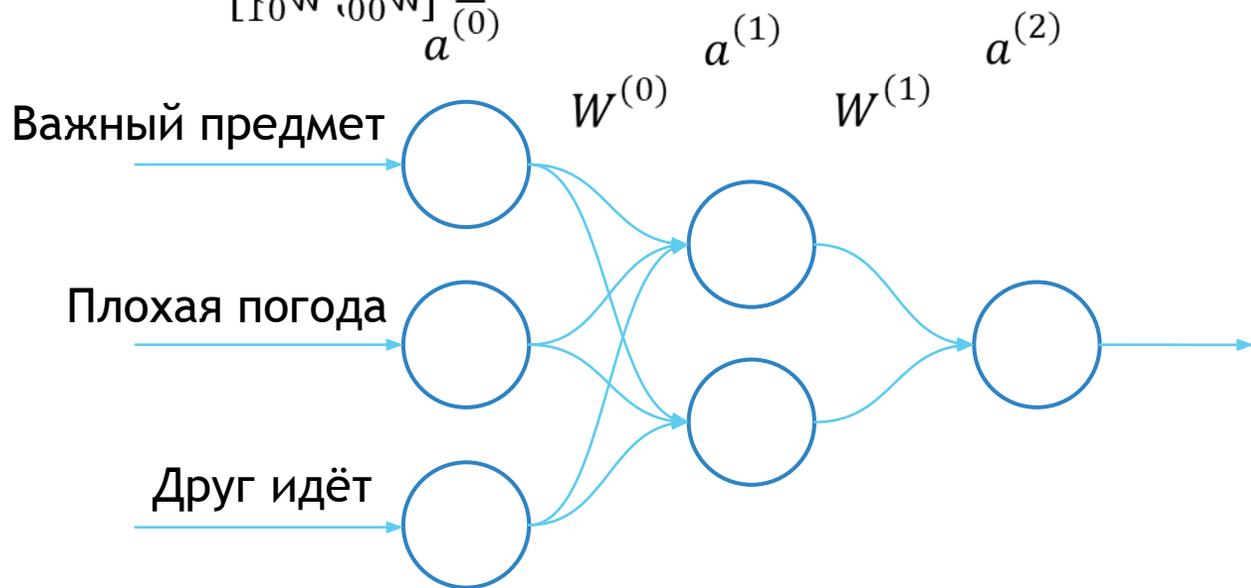
- ▶ Нейросети прямого распространения;
- ▶ Рекуррентные ИНС (RNN);
  - LSTM;
  - GRU.
- ▶ Сеть Хопфилда;
- ▶ Свёрточные ИНС (CNN и DCNN).
- ▶ Развёрточные ИНС (DN)



Пример простой ИНС - ИНС с бинарным выходом

$$[w_{11}, w_{12}, w_{21}, w_{22}, w_{31}, w_{32}] =$$

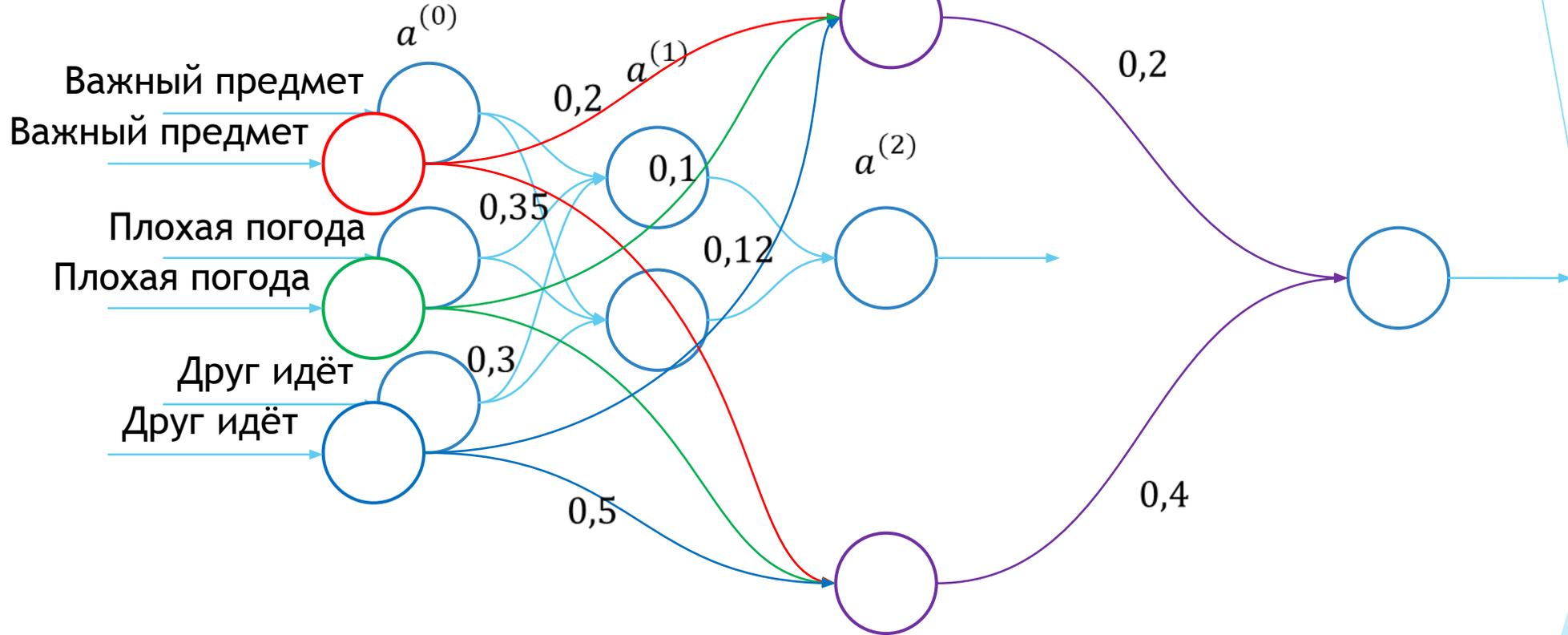
$$[w_{10}, w_{00}] a^{(0)}$$

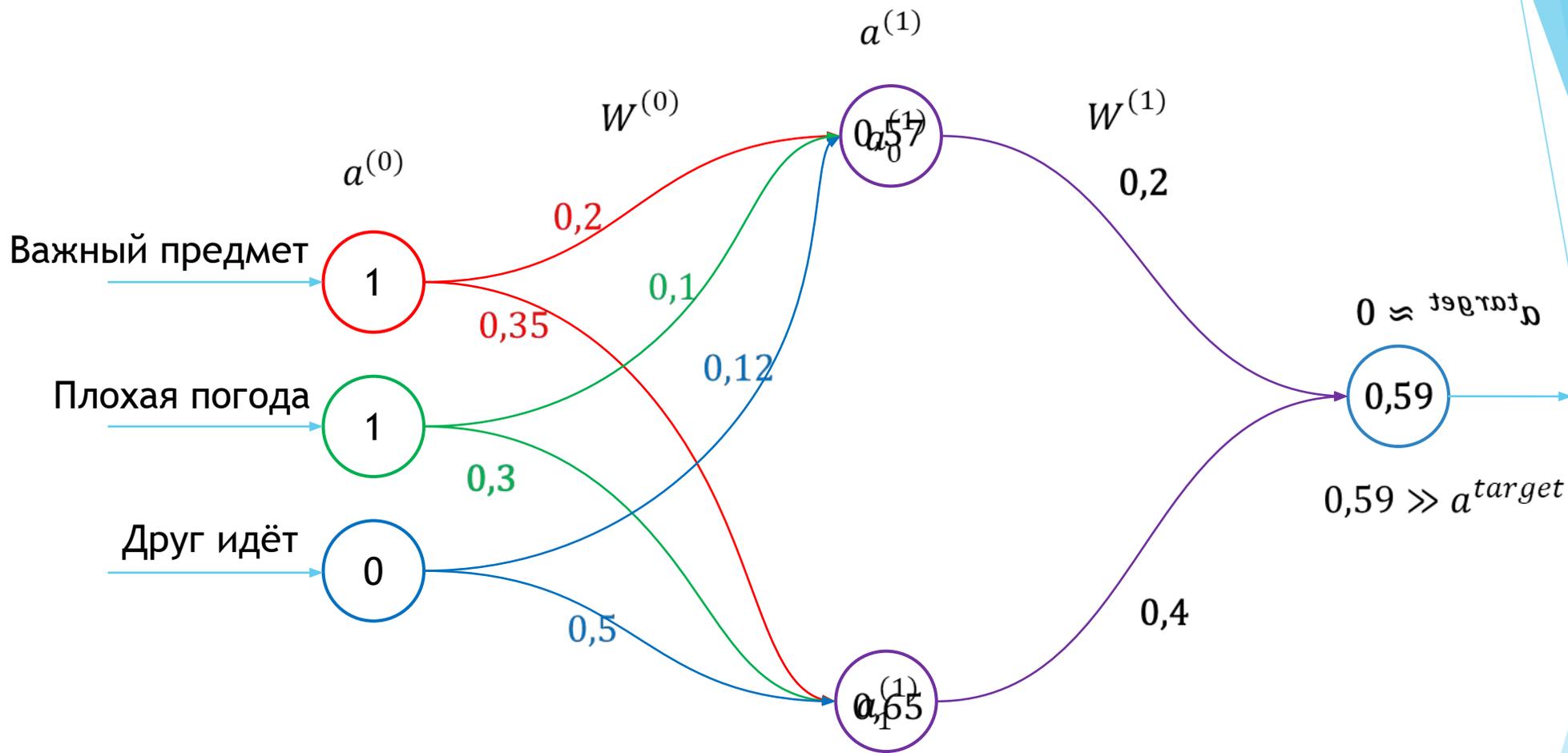




$$W^{(0)} [2,0 ; 3,0 ; 23,0 ; 11,0 ; 1,0 ; 2,0] =$$

$$W^{(1)} [4,0 ; 2,0] =$$





$$a^{(L)} = \sigma(W^{(L-1)} * a^{(L-1)} + b)$$

$$L = 1 \quad a_0^{(1)} = \sigma\left(\sum_j W_{0j}^{(0)} * a_j^{(0)} + b\right)$$

$$b = 0$$

$$a_0^{(1)} = 0,2 * 1 + 0,1 * 1 + 0,12 * 0 + 0 = 0,3$$

$$\sigma(0,3) = \frac{1}{1 + e^{-0,3}} \approx 0,57$$

$$a_0^{(1)} = 0,57$$

# Обучение нейросети.

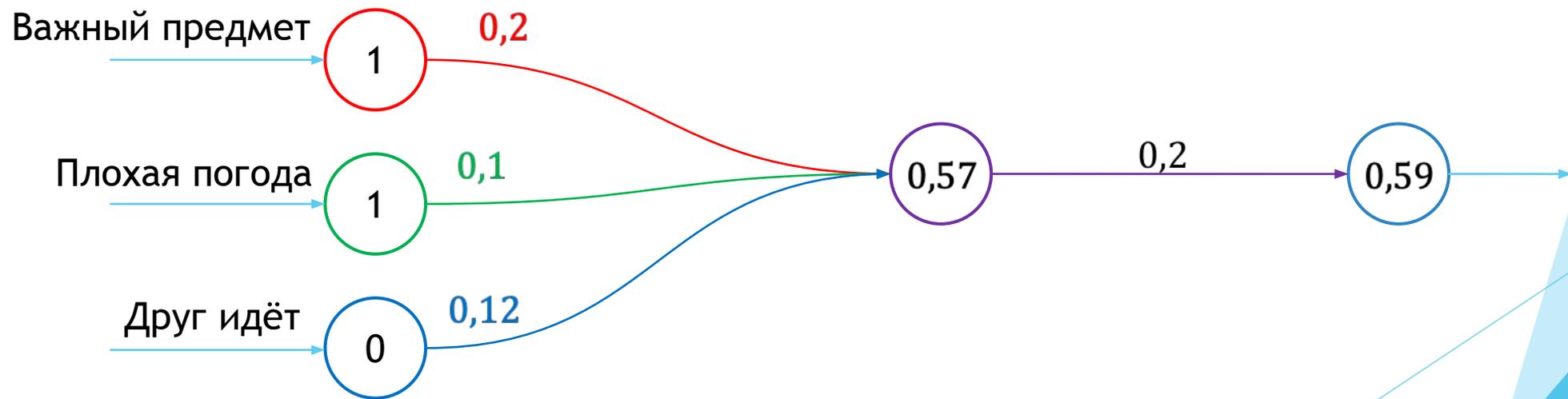
Метод обратного распространения ошибки

$$0 \approx \text{target}_0$$

$$e_{z,0} = \text{target}_0$$

$$C_0 = \frac{1}{2} \left( a^{(L)\text{target}} - a^{(L)\text{output}} \right)^2 - \text{функция стоимости ошибки}$$

$$\nabla C = \frac{\delta C}{\delta w_{kj}^{(L-1)}} = \frac{\delta C}{\delta a_i^{(L)\text{output}}} * \frac{\delta a_i^{(L)\text{output}}}{\delta z_i^{(L)}} * \frac{\delta z_i^{(L)}}{w_{kj}}$$



# Обучение нейросети.

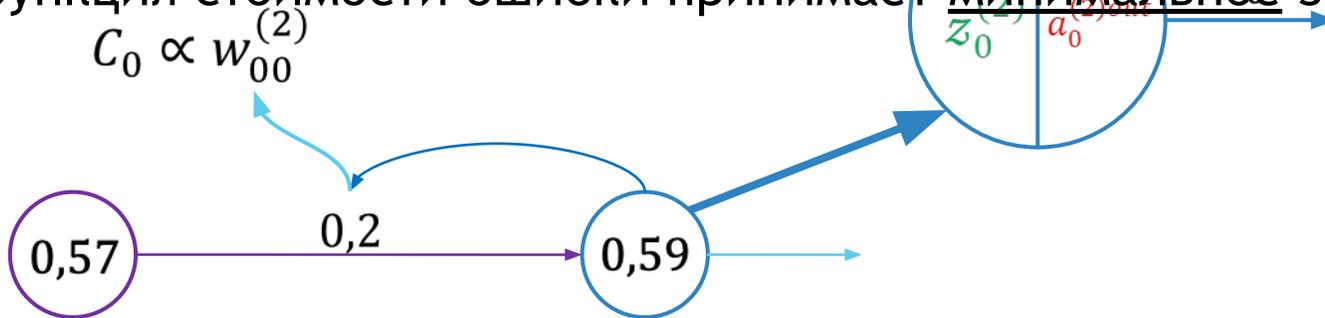
Метод обратного распространения ошибки.

$$0 \approx \text{target}_D \quad e_{z,0} = \text{target}_D$$

$$C_0 = \frac{1}{2} \left( a^{(L)target} - a^{(L)output} \right)^2 - \text{функция стоимости ошибки}$$

$$\nabla C = \frac{\delta C}{\delta w_{kj}^{(L-1)}} = \frac{\delta C}{\delta a_i^{(L)output}} * \frac{\delta a_i^{(L)output}}{\delta z_i^{(L)}} * \frac{\delta z_i^{(L)}}{w_{kj}}$$

Задача обучения нейросети - нахождение таких значений весов, при которых функция стоимости ошибки принимает минимальное значение.



# Обучение нейросети. Вычисление градиента

$$0 \approx \text{target}_0 \quad e_{2,0} = \text{target}_0 \quad C_0 = \frac{1}{2} \left( a^{(L)\text{target}} - a^{(L)\text{output}} \right)^2 - \text{функция стоимости ошибки}$$

$$C_0 = \frac{1}{2} (0 - 0,59)^2 = \frac{0,34}{2} = 0,17$$

$$\nabla C = \frac{\delta C}{\delta w_{kj}^{(L-1)}} = \frac{\delta C}{\delta a_i^{(L)\text{output}}} * \frac{\delta a_i^{(L)\text{output}}}{\delta z_i^{(L)}} * \frac{\delta z_i^{(L)}}{w_{kj}}$$

$$\frac{\delta C}{\delta a_i^{(L)\text{output}}} = C' \left( a_i^{(L)\text{output}} \right) = 2 * \frac{1}{2} \left( a_i^{(L)\text{target}} - a_i^{(L)\text{output}} \right) * -1 = - \left( a_i^{(L)\text{target}} - a_i^{(L)\text{output}} \right)$$

$$z_i^{(L)} = z_0^{(2)} = \sum_j w_{0j} * a_j^{(1)} + b = 0,374$$

$$\frac{\delta a_i^{(L)\text{output}}}{\delta z_i^{(L)}} = \sigma' \left( \delta z_i^{(L)} \right) = \sigma \left( \delta z_i^{(L)} \right) * \left( 1 - \sigma \left( \delta z_i^{(L)} \right) \right)$$

$$\frac{\delta z_i^{(L)}}{w_{kj}} = 1 * a_j^{(L-1)\text{output}} * w_{kj} = a_i^{(L-1)\text{output}}$$

# Обучение нейросети. Вычисление градиента

$$0 \approx \text{target}_0 \quad z_i^{(L)} = z_1^{(2)} = \sum_j w_{0j} * a_j^{(0)} + b = 0,374$$

$0,0 = \text{target}_0$

$$\nabla C = \frac{\delta C}{\delta w_{00}^{(1)}} = - \left( a_0^{(2)target} - a_0^{(2)output} \right) * \left( \sigma \left( \delta z_0^{(2)} \right) * \left( 1 - \sigma \left( \delta z_0^{(2)} \right) \right) \right) * a_i^{(1)output}$$

$$\nabla C = \frac{\delta C}{\delta w_{00}^{(1)}} = - (0 - 0,59) * \left( \sigma(0,374) * (1 - \sigma(0,374)) \right) * 0,57 =$$

$$= 0,59 * (0,59 * (1 - 0,59)) * 0,57 = 0,59 * 0,24 * 0,57 = 0,08$$

$$\nabla C = \frac{\delta C}{\delta w_{00}^{(1)}} = 0,08$$

$$w_{00}^{(1)new} = w_{00}^{(1)} - \eta * \nabla C \quad \eta - \text{ шаг обучения (Learning rate)}$$

# Обучение нейросети. Вычисление градиента

$$0 \approx \text{target}_0 \quad z_i^{(L)} = z_0^{(2)} = \sum_j w_{0j} * a_j^{(0)} + b = 0,374$$

$0,2 = \text{actual}_0$

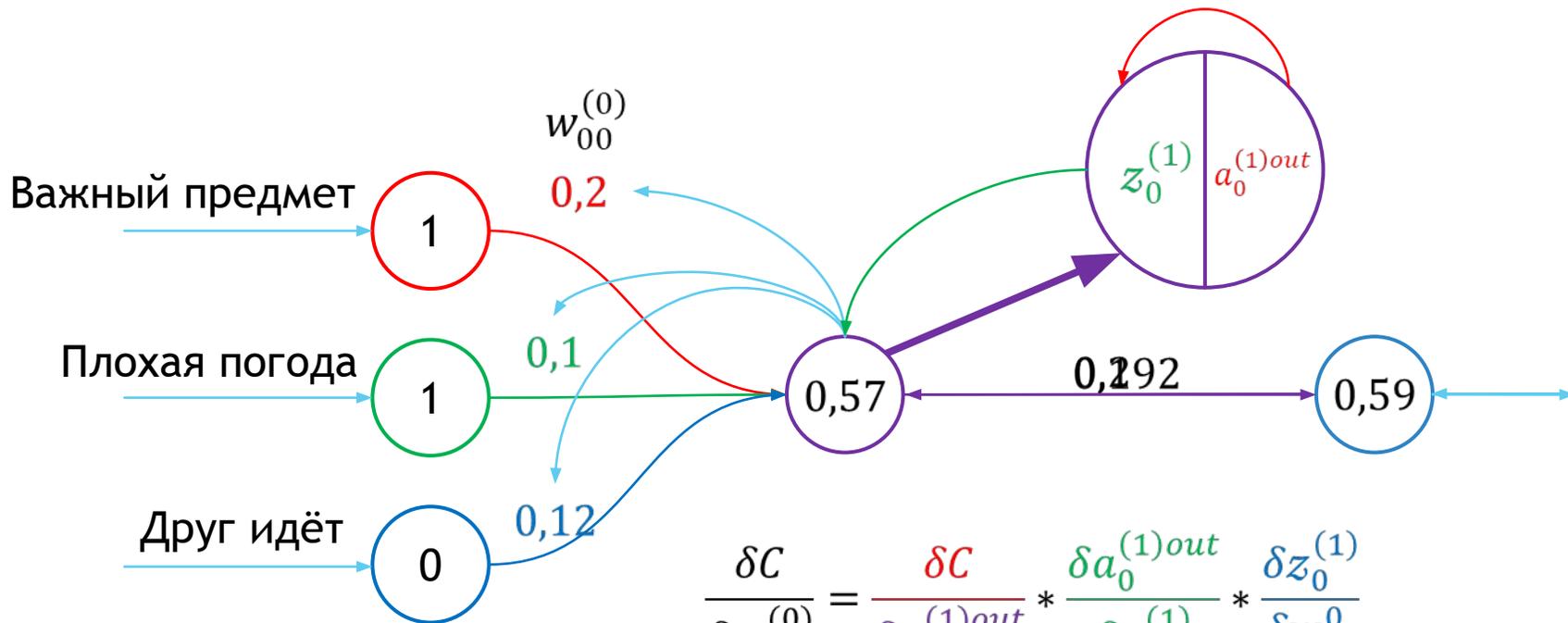
$$\nabla C = \frac{\delta C}{\delta w_{00}^{(1)}} = -\left(a_1^{(2)target} - a_1^{(2)output}\right) * \left(\sigma\left(\delta z_1^{(2)}\right) * \left(1 - \sigma\left(\delta z_1^{(2)}\right)\right)\right) * a_i^{(1)output}$$

$$\nabla C = \frac{\delta C}{\delta w_{00}^{(1)}} = -(0 - 0,59) * \left(\sigma(0,374) * (1 - \sigma(0,374))\right) * 0,57 =$$

$$\eta = 0,1$$
$$= 0,59 * (0,59 * (1 - 0,59)) * 0,57 = 0,59 * 0,24 * 0,57 = 0,08$$
$$w_{00}^{(1)new} = 0,2 - 0,1 * 0,08 = 0,192$$

$$\nabla C = \frac{\delta C}{\delta w_{00}^{(1)}} = 0,08$$

$$w_{00}^{(1)new} = w_{00}^{(1)} - \eta * \nabla C \quad \eta - \text{ шаг обучения (Learning rate)}$$

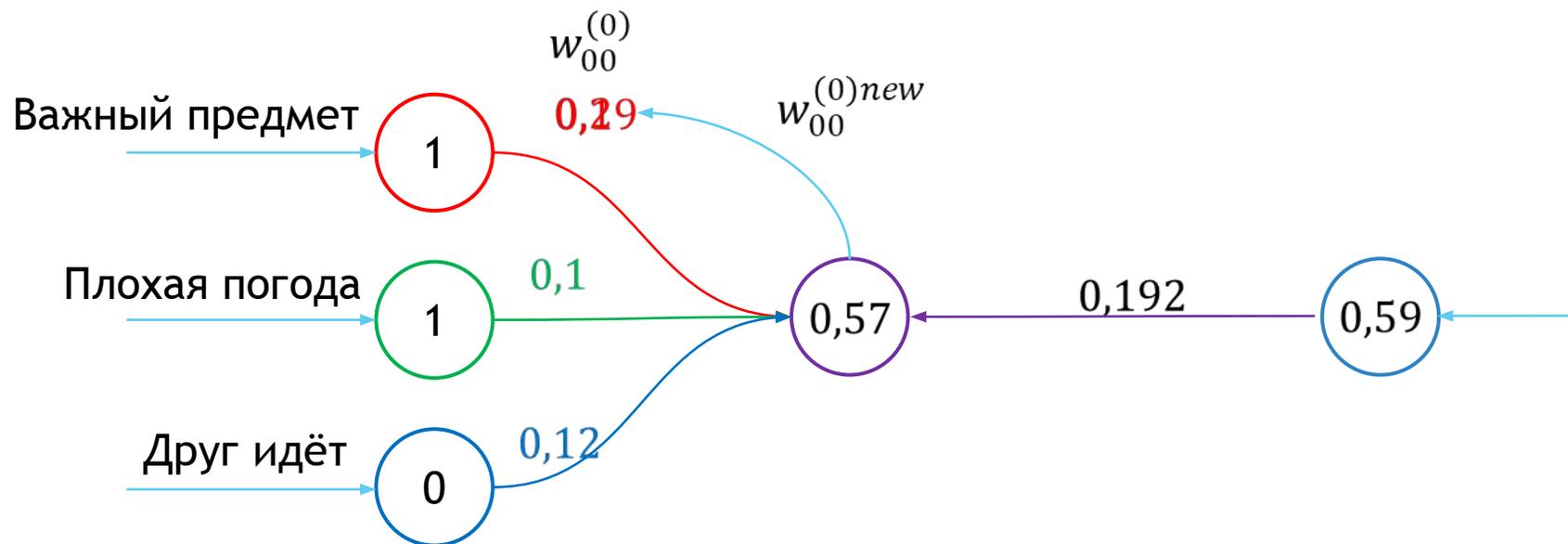


$$\frac{\delta C}{\delta w_{00}^{(0)}} = \frac{\delta C}{\delta a_0^{(1)out}} * \frac{\delta a_0^{(1)out}}{\delta z_0^{(1)}} * \frac{\delta z_0^{(1)}}{\delta w_{10}^0}$$

$$w_{00}^{(1)old} = 0,2$$

$$\frac{\delta C}{\delta a_0^{(1)out}} = 0,14 * 0,2 = 0,028$$

$$\frac{\delta C}{\delta z_0^{(2)}} = 0,59 * 0,24 = 0,14$$



$$\frac{\delta C}{\delta a_0^{(1)out}} = 0,028$$

$$\frac{\delta a_0^{(1)out}}{\delta z_0^{(1)}} = 0,573 * (1 - 0,57(0,3)) * 0,24$$

$$\frac{\delta z_0^{(1)}}{\delta w_{00}^0} = a_0^{(0)} = 1$$

$$\nabla C = \frac{\delta C}{\delta w_{00}^{(0)}} = 0,028 * 0,24 * 1 = 0,006$$

$$w_{00}^{(0) new} = w_{00}^{(0)} - \eta * \nabla C = 0,2 - 0,1 * 0,006 = 0,19$$

# Ещё раз повторим алгоритм

1. Размечаем произвольные значения веса на всех слоях, а так же значения на входных нейронах;
2. Вычисляем значения активации нейронов на скрытых слоях по формуле (1);
3. Вычисляем значение активации на выходном нейроне;
4. Сравниваем с ожидаемым значением;
  1. Если значение близко к ожидаемому, то меняем значения на входных нейронов и идём к шагу 2 (без изменения весов);
  2. Если нет, то идём на шаг 5.
5. Считаем функцию ошибки по формуле (2);
6. Используем метод обратного распространения ошибки (формулы 3 и 4) и записываем новые значения для весов;
7. Возвращаемся на шаг 2.

# Формулы

1.  $a^{(L)} = \sigma(W^{(L-1)} * a^{(L-1)} + b)$  – значение активации входного нейрона на  $L$  слое, где  
 $W^{(L-1)}$  – веса между  $L$  и  $L - 1$  слоем;  
 $a^{(L-1)}$  – значения активации нейронов на предыдущем слое;  
 $b$  – коэффициент смещения

1.1  $a^{(L)} = \sigma(\sum_j w_{kj} * a_j^{(L-1)} + b)$  – формула 1 в раскрытом виде

2.  $C = \frac{1}{2} (a^{(L)target} - a^{(L)output})^2$  – функция стоимости ошибки, где  
 $a^{(L)target}$  – ожидаемое значение активации;  
 $a^{(L)output}$  – полученное значение активации;

# Формулы

$$3. \nabla C = \frac{\delta C}{\delta w_{kj}^{(L)}} = \frac{\delta C}{\delta a_i^{(L)output}} * \frac{\delta a_i^{(L)output}}{\delta z_i^{(L)}} * \frac{\delta z_i^{(L)}}{w_{kj}} - \text{вычисление значения}$$

градиента ошибки на текущем слое, при значении  $w$  – веса на входе нейрона

$$4. w_{kj}^{(L)new} = w_{kj}^{(L)old} - \eta * \nabla C - \text{вычисление нового веса}$$

$\eta$  – шаг обучения (*Learning rate*)

# Ссылки на нейронки

- ▶ Генерируем котиков: <https://thiscatdoesnotexist.com/>
- ▶ Генерируем истории (на русском): <https://porfirevich.ru/>
- ▶ Генератор приключений (англ.): <https://play.aidungeon.io/>
- ▶ Умные фильтры фотошопа: <https://helpx.adobe.com/ru/photoshop/using/neural-filters.html>

# В следующих сериях...

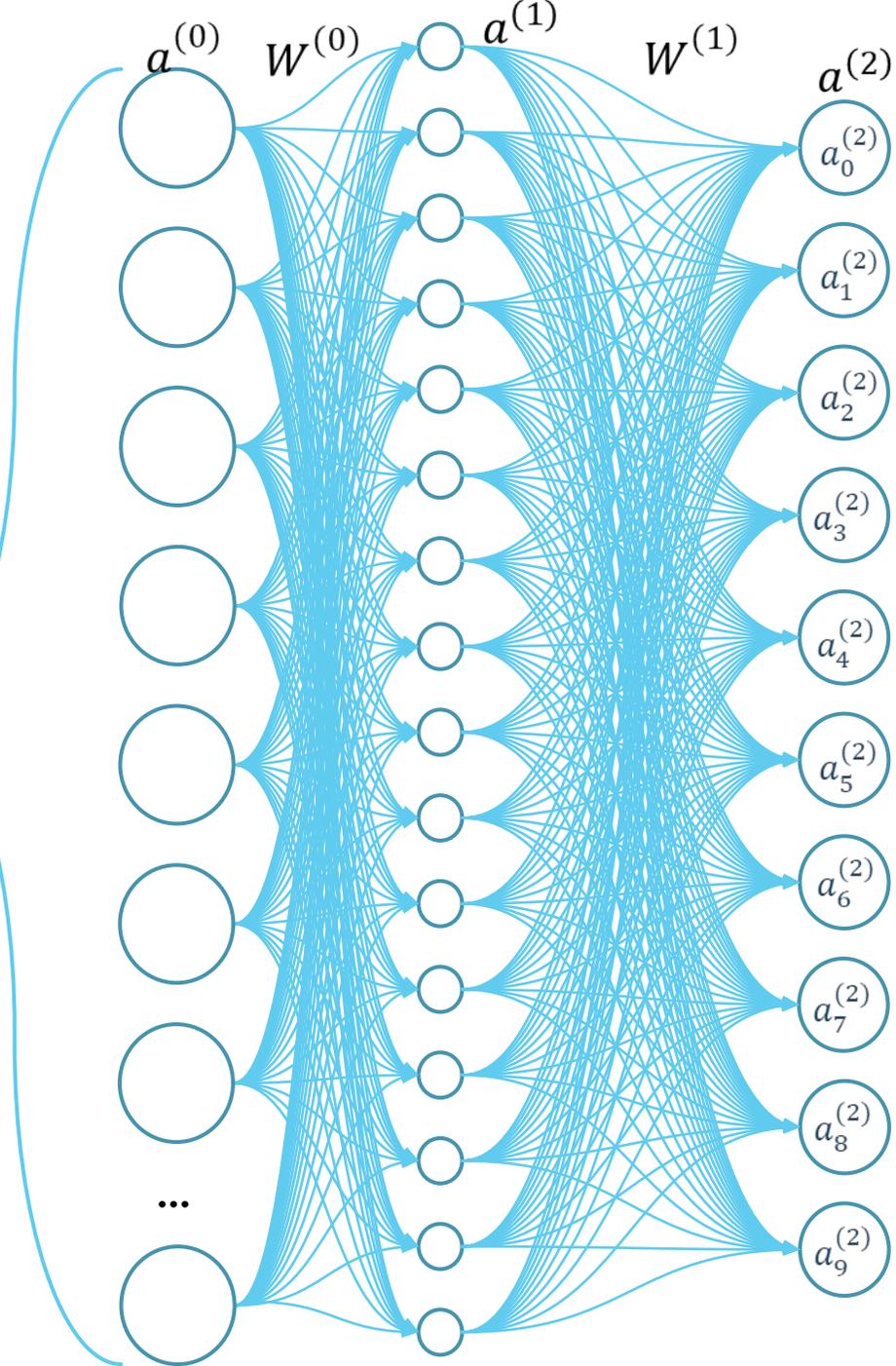
(на практической)

1. Реализация простой нейросети на Python;
2. Задача распознавания рукописных цифр или «Hello, world!» от области ИНС.

# Распознавание рукописного текста

- ▶ Изображение 28x28 пикселя = 784 нейронов на входном слое;
- ▶ Один скрытый слой с произвольным  $N$  количеством нейронов;
- ▶ Выходной слой с  $M=10$  нейронами на каждую цифру от 0 до 9.

784  
нейрона



Функция активации нейрона

$$a_k^{(L)} = \sigma(W_{kj}^{(L-1)} * a_j^{(L-1)} + b_k)$$

Спасибо за внимание!