

# ОБУЧАЮЩИЙ КУРС «ПРОЕКТИРОВАНИЕ НЕЙРОННЫХ СЕТЕЙ»



Руководитель  
Попов С.В.

# Цели и задачи учебного курса для студентов СПО

## ■ Цели:

Дать представление студентам о работе нейронных сетей. Выработать навыки создания нейронных сетей с помощью предложенной методики и оригинального ПО.

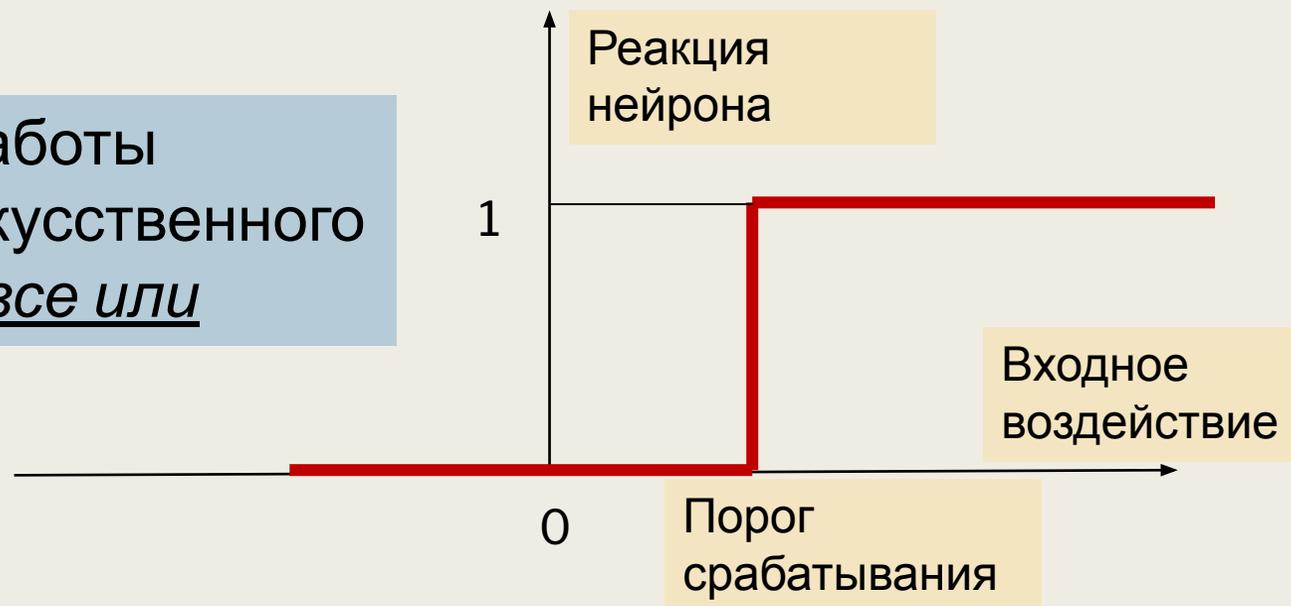
## ■ Задачи:

- 1) Изучить конструкции нейронных сетей и принципы их обучения.
- 2) Освоить приемы обучения нейронных сетей на создаваемых выборках данных.
- 3) Освоить приемы использования нейронных сетей при решении реальных задач распознавания.
- 4) Приобрести навыки программирования на языке Python.

# История

Термин «искусственная нейронная сеть» появился сравнительно недавно. В 1943 году ученые попытались воссоздать известные к тому времени процессы головного мозга, вследствие чего появилась модель нейрона. Исходно искусственный нейрон реализовывал т.н. пороговую функцию. В ходе экспериментов оказалось, что сеть из нескольких нейронов способна к обучению, путем изменения своей структуры.

Принцип работы первого искусственного нейрона – все или ничего



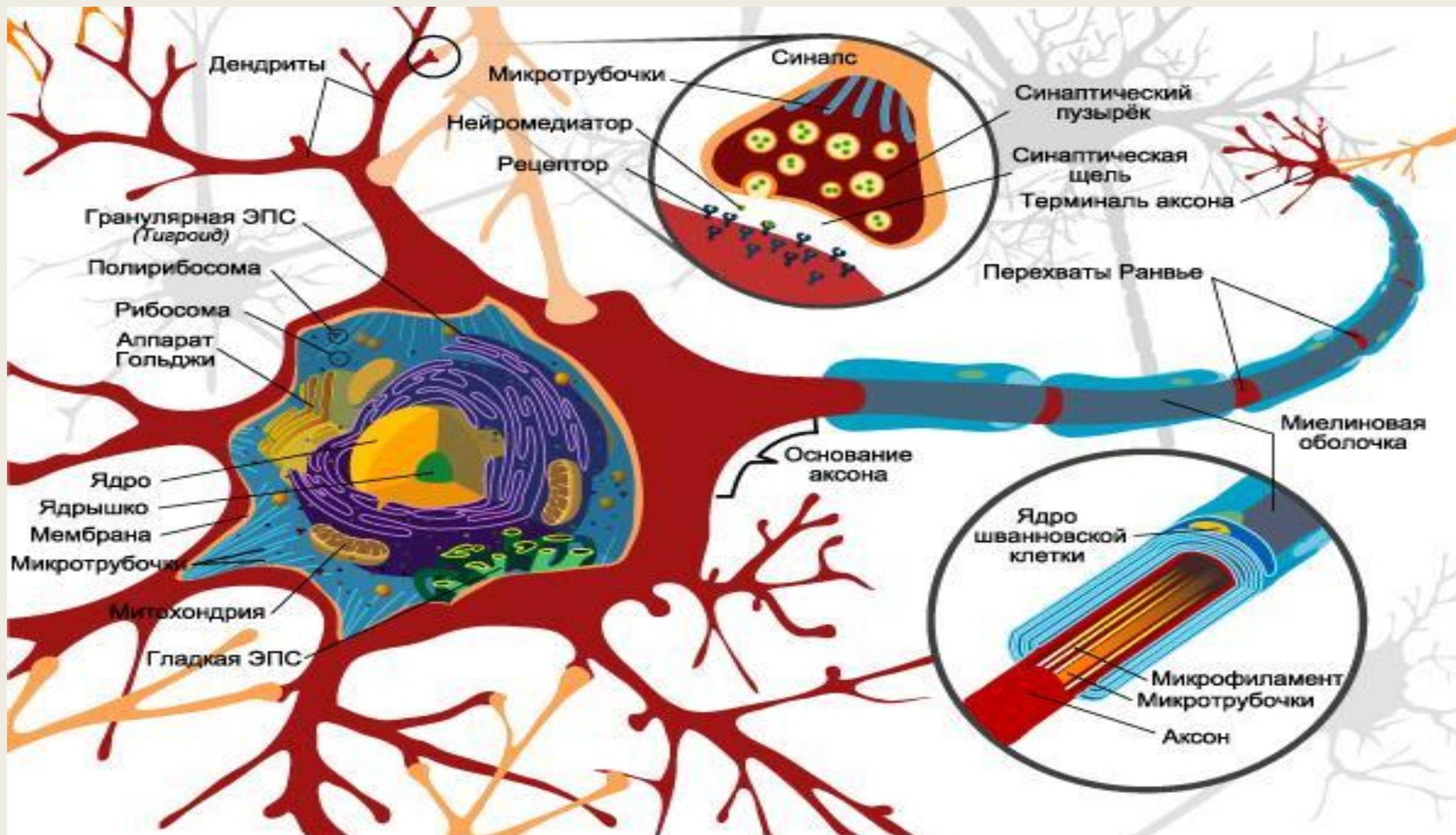
Пороговая функция - простейшая модель нейрона головного мозга

# Современное представление нейрона - сигмоида

Принцип работы современного искусственного нейрона – сигмоида



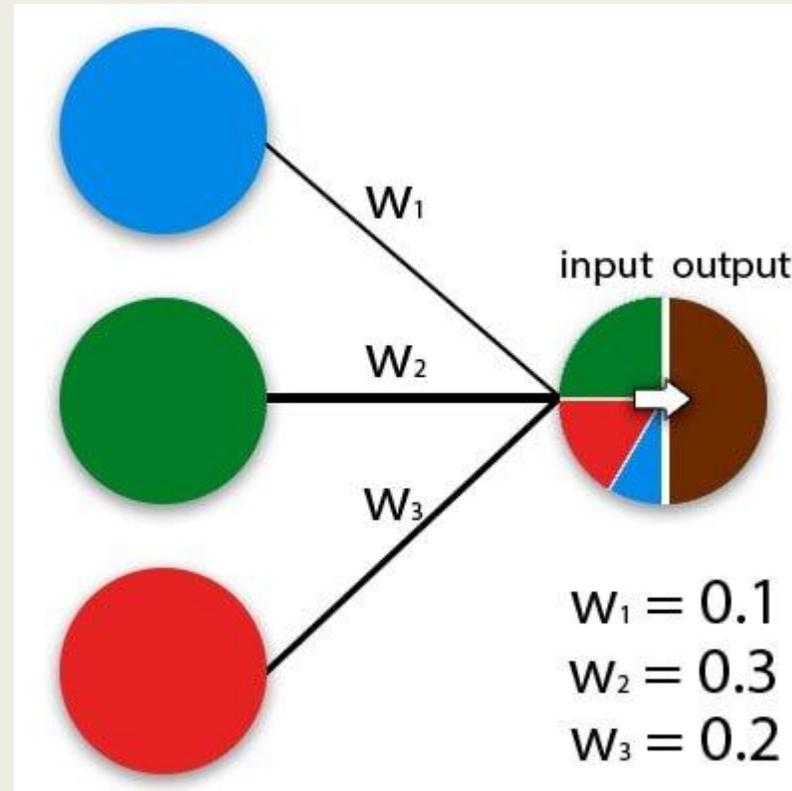
# Нейрон головного мозга



# Нейронные сети

**Нейронная сеть** – есть объединение нескольких нейронов (их может быть несколько тысяч), выход каждого нейрона подается на вход других нейронов. **Входные** нейроны сети **воспринимают** входной образ, **выходные** – демонстрируют **реакцию** на входной раздражитель. Связи между нейронами усиливают или ослабляют выходной сигнал нейрона. Их окончательные значения формируются в процессе обучения сети.

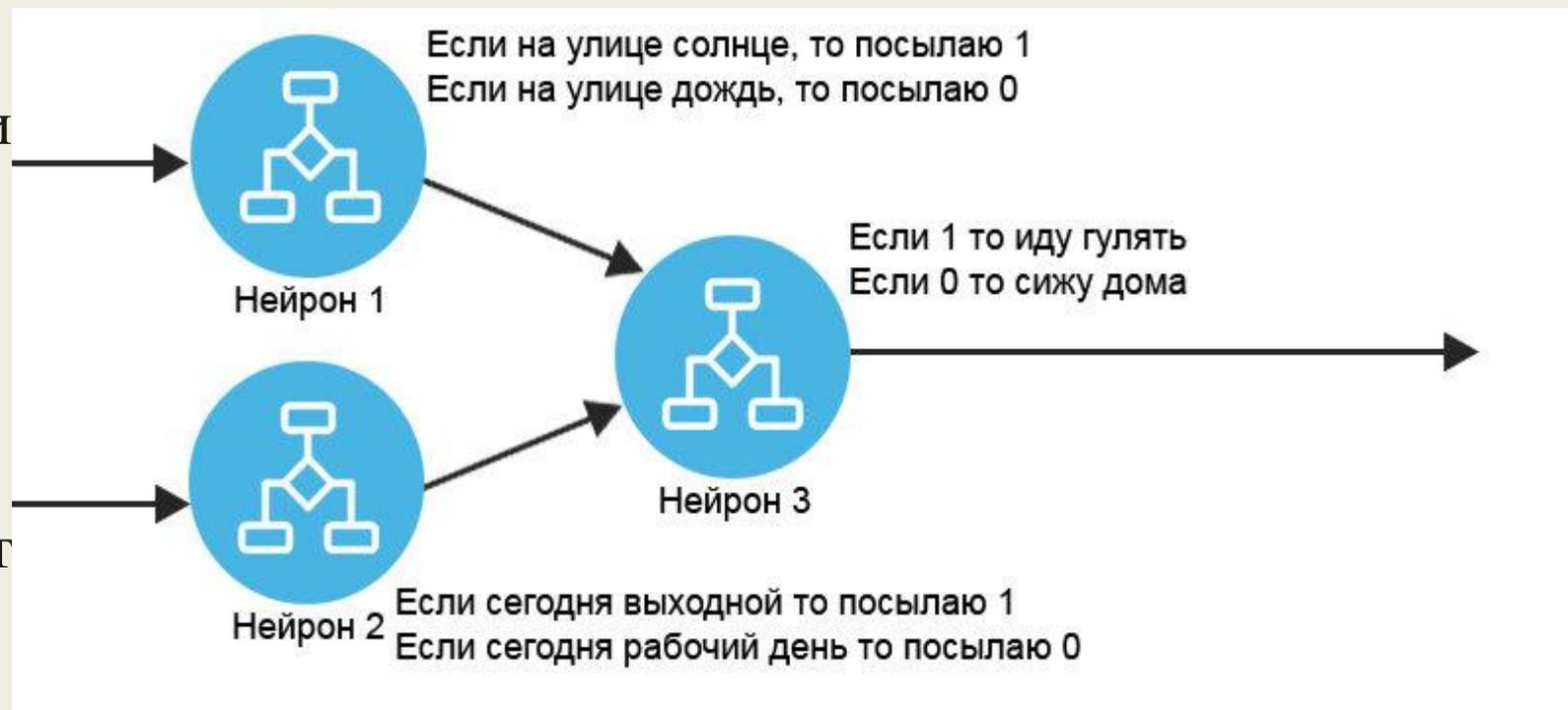
**Обучение сети** состоит в подборе весов нейронных связей, чтобы выходная реакция сети совпадала с целевым значением. Например, чтобы изображение кошки вызывало реакцию сети, отличную от реакции на изображение собаки.



Нейроны соединены друг с другом, транслируя собственные выходные сигналы на вход других нейронов

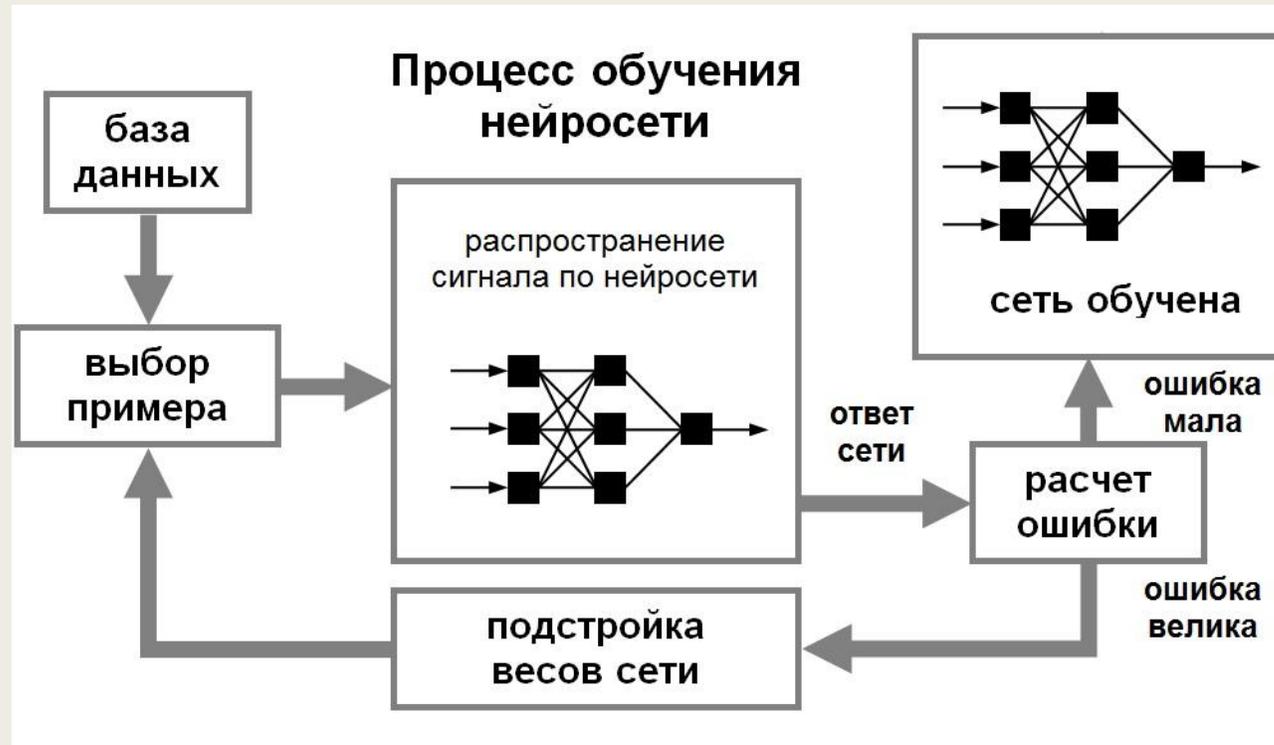
# Пример нейронной сети

Пусть человек любит гулять в солнечные дни и в выходные дни, когда у него много времени. Но он не любит гулять под дождем. Примерно такая нейросеть выдает 1 лишь тогда, когда на улице солнце или выходной день.



# Процесс обучения нейросети

Обучение сети – состоит в подстройке весов нейронных связей.  
Удачное обучение нейросети означает, что она выдает ожидаемую реакцию на входной раздражитель.



# Сеть с одним нейроном



# Простейшая нейронная сеть – иллюстрация формирования веса связей

График ошибки при обучении:  
(Предсказание – Цель)<sup>2</sup>



Градиентный метод поиска минимума –  
используется при обучении сети

# Реализация градиентного спуска

```
Вход = 0.8          # Входное значение единственного нейрона
Цель = 1.1         # Ожидаемая реакция нейрона
Вес = np.random.rand() # Случайный (начальный!) вес связи нейрона
for i in range(10): # Осуществляем 10 итераций подбора веса
    Предсказание = Вход * Вес # Реакция нейрона (произведение входа на вес)
    Ошибка = (Предсказание - Цель) ** 2 # Ошибка (квадратичная функция)
    Разность = Предсказание - Цель # Разность между реакцией и ожидаемым
    # значением (указывает наклон графика ошибки)
    Корректировка = Разность * Вход # Корректировка веса
    Вес -= Корректировка # Новое значение веса
if Ошибка < 0.01 :
    print(« Итерация = ", i, "Финальное значение веса = ", Вес)
```

# Поиск необходимого веса

Исходный вес = 0.7294971369350512

Ошибка = 0.2666713255840295 Предсказание = 0.583597709548041

=====

Очередной Вес = 1.1426189692966187

Ошибка = 0.034560603795690155 Предсказание = 0.914095175437295

=====

Очередной Вес = 1.2913428289467828

Ошибка = 0.004479054251921433 Предсказание = 1.0330742631574263

=====

Очередной Вес = 1.3448834184208418

Ошибка = 0.0005804854310490181 Предсказание = 1.0759067347366735

=====

Итерация = 3 Финальное значение веса = 1.364158030631503

# Более сложная предметная область – распознавание конфигураций

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

На выделенных конфигурациях сеть должна выдавать 1, на остальных – 0.  
Обучающими входами сети являются векторы: (1, 6, 7), (2, 7, 9), (8, 1, 3) и (4, 3, 9),  
которым  
приписана реакция 1 и несколько входов, описывающих другие конфигурации,  
имеющие нулевую реакцию.

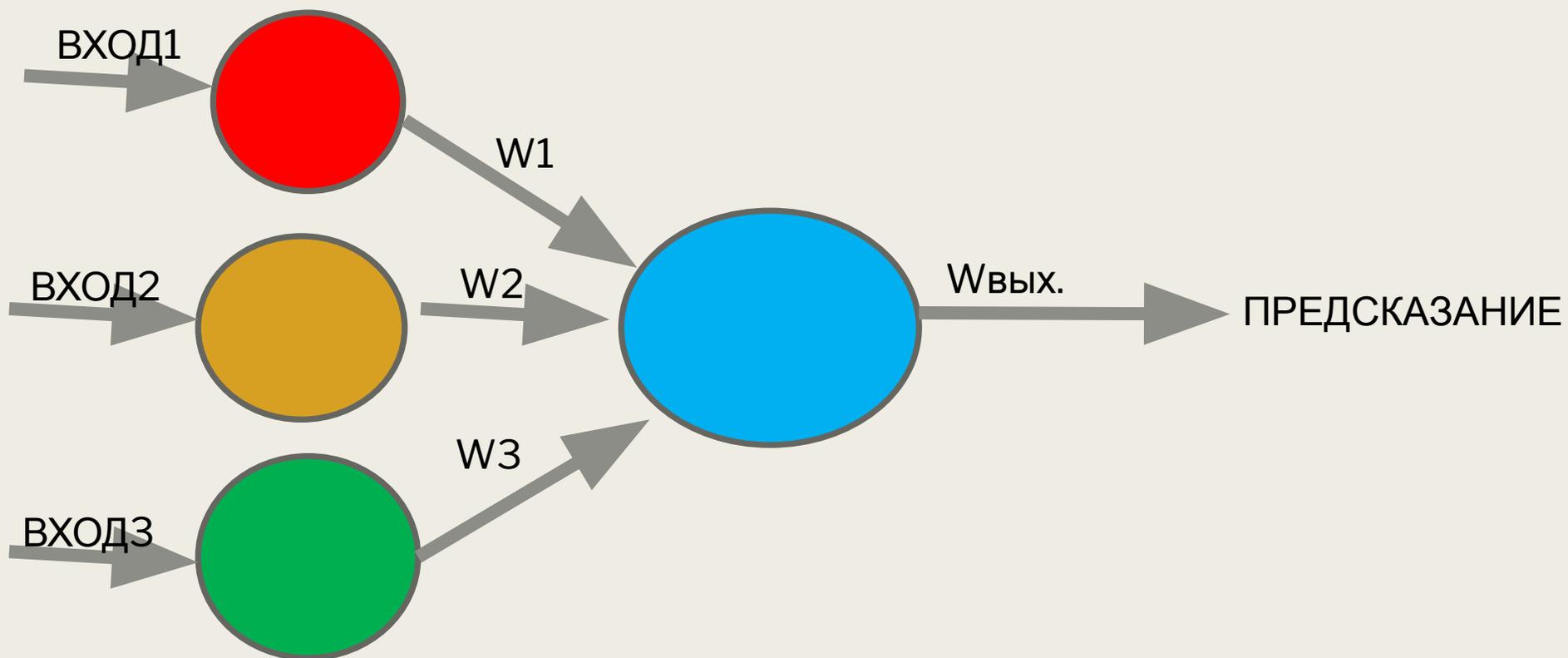
Финал: Ошибка = 0.00246895352855757

Веса = [ 0.29998269 0.41414685 -0.34044844]

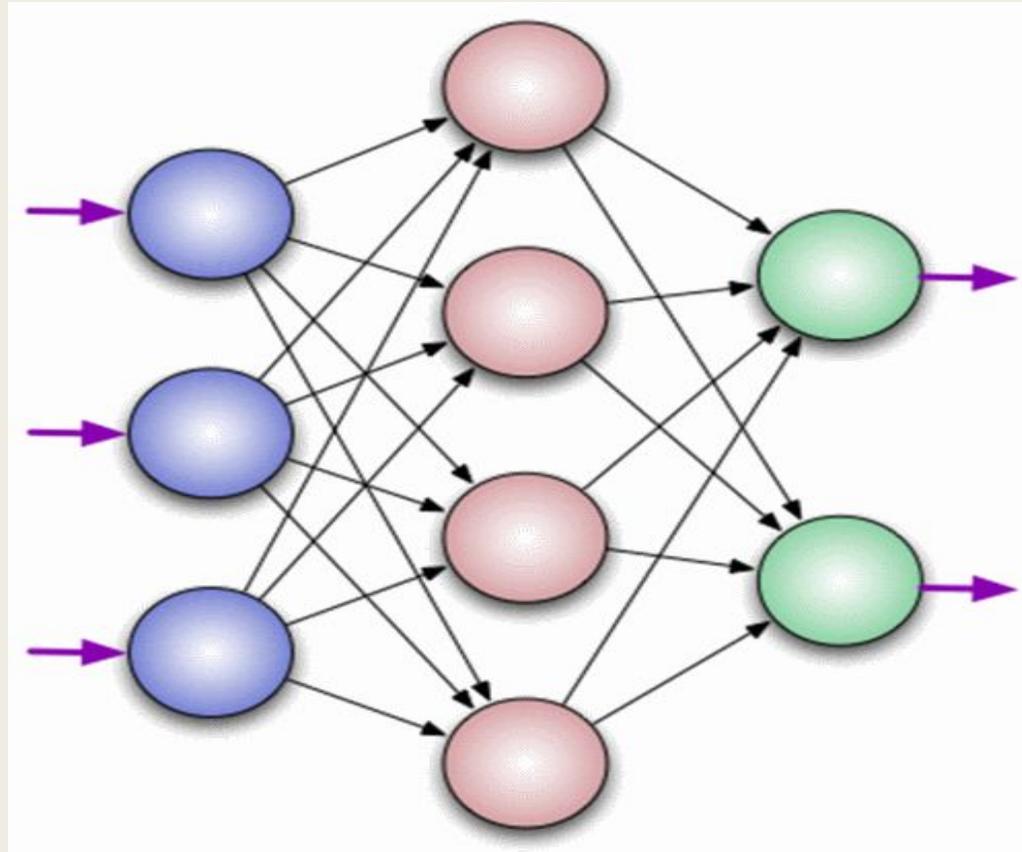
===== Проверяем настройку сети на векторе (3, 4, 8) =====

Проверка Ошибка = 0.027906385288831312

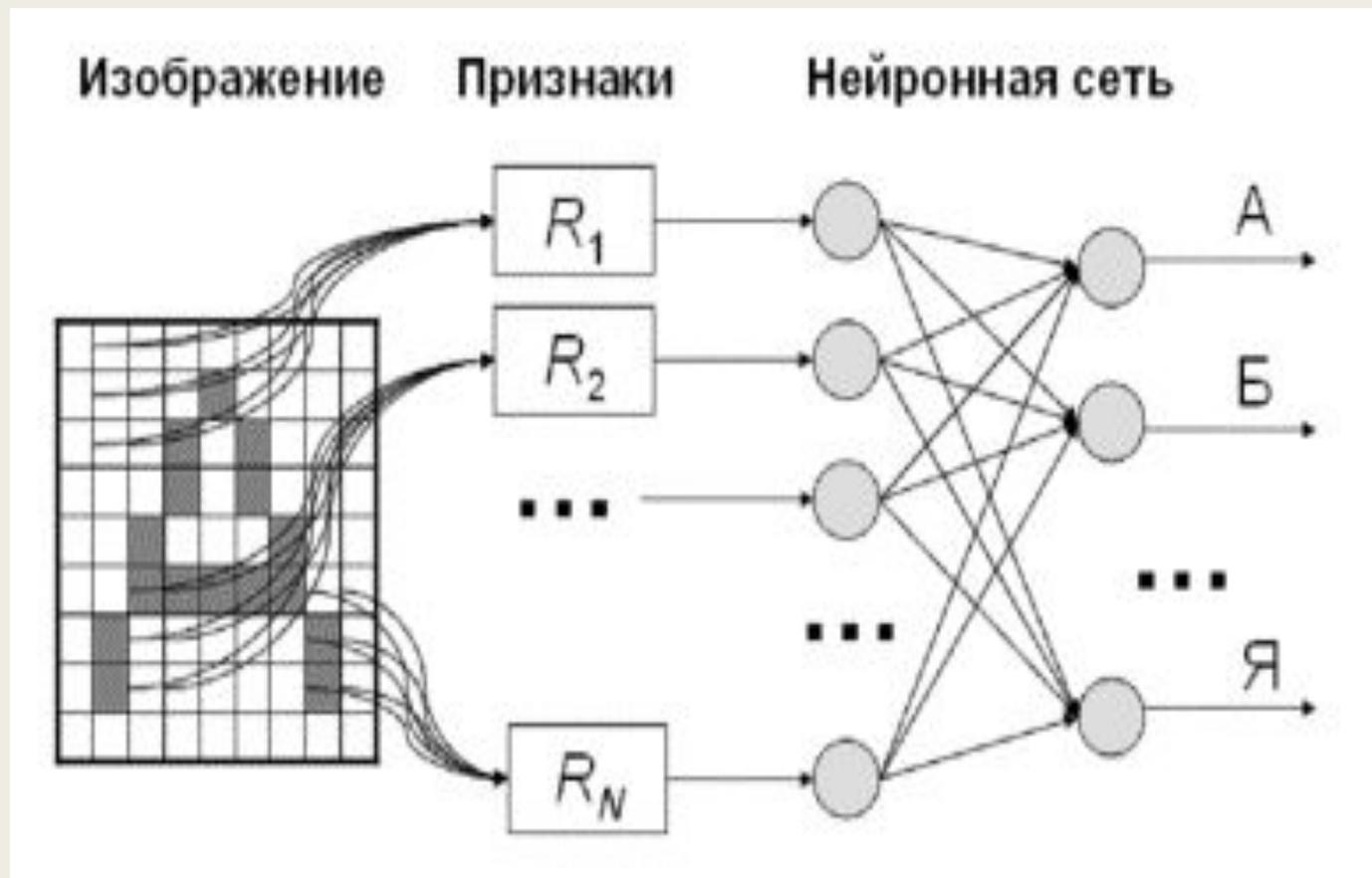
# Нейронная сеть для распознавания ситуаций



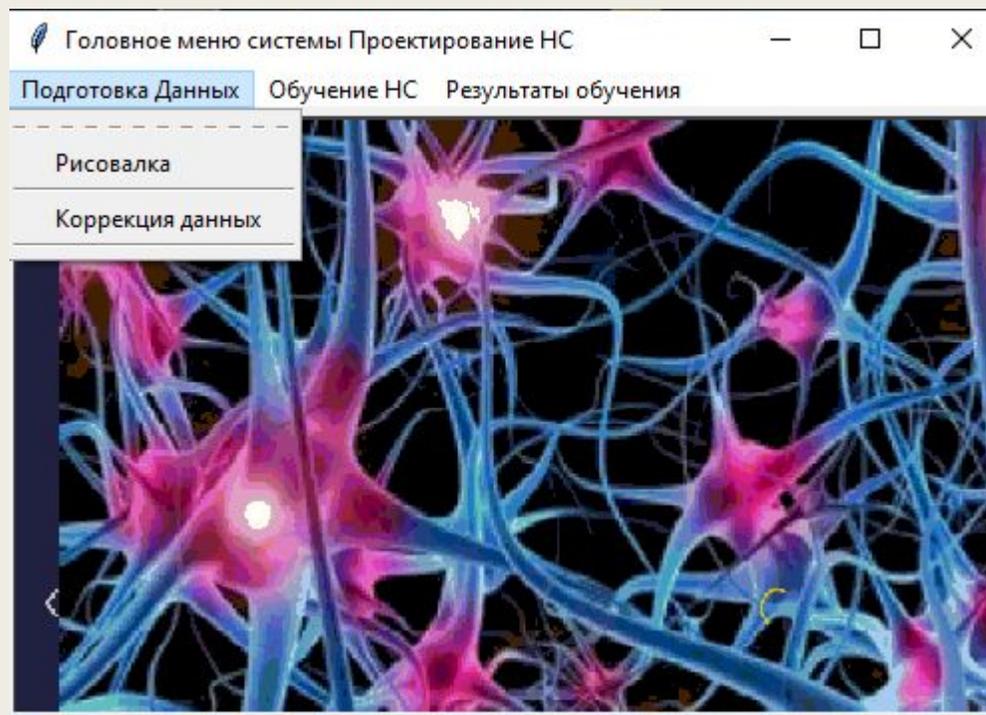
Общий вид нейронной сети. Используется тот же градиентный поиск, но уже в векторном пространстве



# Пример распознавания текста нейронной сетью



# Главная форма учебного курса «Проектирование нейронных сетей»



# Подготовка обучающих данных с помощью поля 30 на 30 точек

Покрасим поле

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59
60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89
90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119
120	121	122	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149
150	151	152	153	154	155	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179
180	181	182	183	184	185	186	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209
210	211	212	213	214	215	216	217	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269
270	271	272	273	274	275	276	277	278	279	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299
300	301	302	303	304	305	306	307	308	309	310	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329
330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359
360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389
390	391	392	393	394	395	396	397	398	399	400	401	402	403	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419
420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449
450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	466	467	468	469	470	471	472	473	474	475	476	477	478	479
480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	497	498	499	500	501	502	503	504	505	506	507	508	509
510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	528	529	530	531	532	533	534	535	536	537	538	539
540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559	560	561	562	563	564	565	566	567	568	569
570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	590	591	592	593	594	595	596	597	598	599
600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	622	623	624	625	626	627	628	629
630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	652	653	654	655	656	657	658	659
660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675	676	677	678	679	680	681	682	683	684	685	686	687	688	689
690	691	692	693	694	695	696	697	698	699	700	701	702	703	704	705	706	707	708	709	710	711	712	713	714	715	716	717	718	719
720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	745	746	747	748	749
750	751	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	776	777	778	779
780	781	782	783	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	807	808	809
810	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837	838	839
840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864	865	866	867	868	869
870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895	896	897	898	899

Transform Strach Сглаживание InFile Чистка Обучаем









## Обработка изображений в интересах обучения нейронной сети - 2

Затем обучающая выборка подается на вход спроектированной трехслойной нейронной сети и происходит ее обучение на этой выборке.

Параметр	Значение
Число входных нейронов =	900
Число выходных нейронов =	33
Число скрытых нейронов =	400
Коэффициент обучения =	0.3
Количество итераций =	1000
Файл с обучающей выборкой =	1000

Введи входные параметры сети

Произведено Обучение на выборке из 37 букв различного начертания с повторениями. Результат обучения: **Номер буквы в алфавите, буква, ошибка обучения**

['**2 Б** [0.00800006]', '**2 Б** [0.01497103]', '**3 В** [0.01335256]', '**3 В** [0.0181136]',  
'**4 Г** [0.01279627]', '**4 Г** [0.01634902]', '**16 Р** [0.01584977]', '**16 Р** [0.01667234]',  
'**1 А** [0.02205391]', '**12 М** [0.01168011]', '**1 А** [0.01315646]', '**2 Б** [0.00944931]',  
'**2 Б** [0.01592796]', '**14 О** [0.01389658]', '**14 О** [0.01616591]', '**14 О**  
[0.02212198]',  
'**15 П** [0.02969158]', '**16 Р** [0.01231214]', '**16 Р** [0.0146418]', '**17 С** [0.015533]',  
'**17 С** [0.01580154]', '**9 И** [0.00633291]', '**9 И** [0.01254909]', '**9 И** [0.02018677]',  
'**10 К** [0.01449295]', '**10 К** [0.01854573]', '**10 К** [0.01040025]', '**10 К**  
[0.01798245]',  
'**11 Л** [0.01519966]', '**11 Л** [0.01258915]', '**11 Л** [0.01449979]', '**12 М** [0.00972242]',  
'**12 М** [0.01810413]', '**12 М** [0.01679458]', '**18 Т** [0.01643644]', '**18 Т**  
[0.01468914]',  
'**18 Т** [0.01351758]']

# Работа с обученной сетью

1. Обученная сеть однозначно описывается своими весовыми коэффициентами. Для каждой обучающей выборки обучение может давать различные результаты.
2. Важно! По весовым коэффициентам сеть восстанавливается однозначно, и ее можно использовать в дальнейшем в практической деятельности.
3. Для проверки качества обучения используется контрольная выборка образов, как правило, реальных данных.

# Применение нейронных сетей

- Поисковые системы



Яндекс и Google не первый год используют нейронные сети для обучения собственных поисковых систем, делая их «умнее». Они адаптируются под конкретного пользователя, узнают о его предпочтениях и выдают максимально релевантные результаты.

# Применение нейронных сетей

Голосовые  
ассистенты



Системы распознавания речи достигли такого уровня, что позволяют распознать вопрос и предоставить информацию по нему, или выполнить ту или иную функцию. Эти голосовые ассистенты внедряются в технику, поэтому мы можем голосом попросить включить музыкальную композицию, притушить свет в комнате, открыть окно на проветривание и поставить таймер на приготовление еды в мультиварке.

# Книги про нейронные сети



В книге дается информация для читателей, которые хотят разобраться в основах нейронных сетей. В ходе прочтения можно научиться программировать сети, не имея опыта и каких-либо углубленных знаний в математике. Автор предоставляет материал для изучения области с нуля и без использования машинного обучения.

## Преимущества:

- ❖ подходит новичкам;
- ❖ не требует сложных расчетов и пониманий математических алгоритмов;
- ❖ автором сделаны выводы, которые помогают укрепить материал, и не усложняют понимание сути вопроса.

# Книги про нейронные сети



Книга выступает в роли справочника по введению в практику и теорию по нейронным сетям. Она предназначена для тех, кто только начинает разбираться в этой сфере, и хочет освоить более качественный материал. В книге освещены основные теоретические аспекты и основы, которые необходимы для понимания системы в самом начале без достаточной базы знаний.

В практической части автор пошагово описывает этапы создания кода, на основе которого создается нейронная сеть на языке Python.

Создаем нейронную сеть. Рашид Тарик

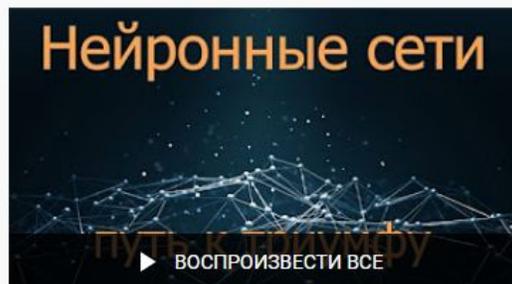
# Youtube канал

На Youtube есть такой канал под названием selfedu([https://www.youtube.com/c/selfedu\\_rus/featured](https://www.youtube.com/c/selfedu_rus/featured) )

В основном, содержит видеоуроки по математике, алгоритмам обработки данных (в том числе по нейронным сетям), по языкам программирования: C, C++, Java, Python (Питон), JavaScript, HTML, CSS, а также по их многочисленным библиотекам.



# Youtube канал



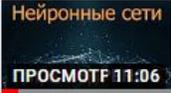
## Нейронные сети на Python. Уроки

33 видео • 23 949 просмотров • Обновлено сегодня



selfedu

ПОДПИСАТЬСЯ

-  **Нейронные сети** Нейронные сети: краткая история триумфа  
selfedu
-  **НЕЙРОННЫЕ СЕТИ #1** Структура и принцип работы полносвязных нейронных сетей | #1 нейросети на Python  
selfedu
-  **НЕЙРОННЫЕ СЕТИ #2** Персептрон - возможности классификации образов, задача XOR | #2 нейросети на Python  
selfedu
-  **НЕЙРОННЫЕ СЕТИ #3** Back propagation - алгоритм обучения по методу обратного распространения | #3 нейросети на Python  
selfedu
-  **НЕЙРОННЫЕ СЕТИ #4** Ускорение обучения, начальные веса, стандартизация, подготовка выборки | #4 нейросети на Python  
selfedu

# Библиографические ссылки

- <https://future2day.ru/nejronnye-seti/>
- <https://www.youtube.com/playlist?list=PLA0M1Bcd0w8yv0XGiF1wjerjSZVSrYbjh>
- <https://proglib.io/p/pishem-neyroset-na-python-s-nulya-2020-10-07>
- <https://progerlib.ru/neural-networks-books>
- <https://zen.yandex.ru/media/nuancesprog/glubokie-neiroseti-rukovodstvo-dlia-nachinaiuscih-5f7b00a2952c3b370e936002>
- Ф.М. ГАФАРОВ, А.Ф. ГАЛИМЯНОВ ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ И ИХ ПРИЛОЖЕНИЯ
- <https://zen.yandex.ru/media/scienceeveryday/neiroseti-cto-eto-takoe-i-zachem-oni-nujny-5b0457dcad0f22d5a951a9be>
- <https://neurohive.io/ru/tutorial/prostaja-nejronnaja-set-python/>
- <https://future2day.ru/ii/>