

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования «Вятский государственный университет»
Институт математики и информационных систем
Факультет автоматики и вычислительной техники
Кафедра электронных вычислительных машин

Исследование многослойного персептрона с обучением по методу с
обратным распространением ошибки

Отчёт
Лабораторная работа № 1 по дисциплине
«Системы обработки знаний»

Выполнила студент группы ИВТб-41 _____/Седов М. Д./
Проверил доцент кафедры ЭВМ _____/Ростовцев В. С./

Киров 2021

Цель

Изучить алгоритм обратного распространения ошибки (ОРО) в процессе обучения нейронной сети при вариативных параметрах обучения. Работа выполняется с помощью программы BackPropagate 3.0.0.

1 Задание

Обучающая выборка представлена в приложении А. Используется выходная функция $out = X_1 - X_2^2 - X_3$. Переменные x_1 измеряется в промежутке $[-4;4]$ с шагом 0,1, переменная x_2 изменяется в промежутке $[-6;6]$ с шагом 0,1, x_3 - $[1;5]$.

2 Протокол выполнения

Во всех таблицах ниже цветом выделен наилучший результат. Значение минимальной ошибки не учитывалось при выборе.

2.1 Влияние нормализации на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000

В таблице 1 показаны результаты исследования.

Таблица 1 – Влияние нормализации на эффективность алгоритма ОРО

Тип нормализации	Результат
Без нормализации	Максимальная ошибка: 41,01 Минимальная ошибка: 0 Средняя ошибка: 24,6601 Среднеквадратичная ошибка: 53811,9493
[0;1]	Максимальная ошибка: 22,973084 Минимальная ошибка: 0,283357 Средняя ошибка: 10,695654459854 Среднеквадратичная ошибка: 9457,60845399253
[-0.5;0.5]	Максимальная ошибка: 24,699215 Минимальная ошибка: 2,395598 Средняя ошибка: 12,3325194817518 Среднеквадратичная ошибка: 12133,2649603782
[-1;1]	Максимальная ошибка: 41,009991 Минимальная ошибка: 0 Средняя ошибка: 24,6601454 Среднеквадратичная ошибка: 53811,9483094731

Применение нормализации [0;1] значительно сокращает величину ошибок.

2.2 Влияние выбора примеров на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000.

Нормализация: [0;1].

В таблице 2 показаны результаты исследования.

Таблица 2 – Влияние выбора примеров на эффективность алгоритма ОРО

Выбор примеров	Результат
Последовательный	Максимальная ошибка: 23,844323 Минимальная ошибка: 1,578702 Средняя ошибка: 11,395781379562 Среднеквадратичная ошибка: 10551,14052336443
Случайный	Максимальная ошибка: 22,973084 Минимальная ошибка: 0,283357 Средняя ошибка: 10,695654459854 Среднеквадратичная ошибка: 9457,60845399253

Случайный выбор примеров показывает лучший результат по сравнению с последовательным выбором примеров обучения.

2.3 Влияние крутизны функции на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000.

Нормализация: [0;1].

Выбор примеров: случайный.

В таблице 3 показаны результаты исследования.

Таблица 3 – Влияние крутизны функции на эффективность алгоритма ОРО

Крутизна функции	Результат
0,1	Максимальная ошибка: 24,658934 Минимальная ошибка: 2,346057 Средняя ошибка: 12,3473078686131 Среднеквадратичная ошибка: 12158,0145082307
0,5	Максимальная ошибка: 22,973084 Минимальная ошибка: 0,283357 Средняя ошибка: 10,695654459854 Среднеквадратичная ошибка: 9457,60845399253
1,0	Максимальная ошибка: 17,447951 Минимальная ошибка: 0,02191999999 Средняя ошибка: 4,3755617810219 Среднеквадратичная ошибка: 2116,10844434154
3,0	Максимальная ошибка: 18,083525 Минимальная ошибка: 0,008578 Средняя ошибка: 4,40751371532847 Среднеквадратичная ошибка: 2380,44954314138

Наилучшие результаты – при крутизне функции 1,0.

2.4 Влияние смещения на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000.

Нормализация: [0;1].

Выбор примеров: случайный.

Крутизна функции: 1.

В таблице 4 показаны результаты исследования.

Таблица 4 – Влияние смещения на эффективность алгоритма ОРО

Смещение	Результат
0,1	Максимальная ошибка: 17,03812 Минимальная ошибка: 0,030581 Средняя ошибка: 5,0331838759 Среднеквадратичная ошибка: 2972,56755363546
0,5	Максимальная ошибка: 17,515156 Минимальная ошибка: 0,021265999999 Средняя ошибка: 4,6555025912 Среднеквадратичная ошибка: 2570,09050288
1,0	Максимальная ошибка: 17,447951 Минимальная ошибка: 0,02191999999 Средняя ошибка: 4,3755617810219 Среднеквадратичная ошибка: 2116,10844434154
2,0	Максимальная ошибка: 17,860393 Минимальная ошибка: 0,02604099999 Средняя ошибка: 5,03921960583 Среднеквадратичная ошибка: 2578,47202008895

Наилучшие результаты наблюдаются при смещении 1,0. Наименьшая максимальная ошибка наблюдается при смещении 0,1, а наименьшие минимальная, средняя и среднеквадратическая ошибка – при смещении 1,0.

2.5 Влияние скорости обучения на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000.

Нормализация: [0;1].

Выбор примеров: случайный.

Крутизна функции: 1.

Смещение: 1.

В таблице 5 показаны результаты исследования.

Таблица 5 – Влияние скорости обучения на эффективность алгоритма ОРО

Скорость обучения	Результат
0,01	Максимальная ошибка: 23,889278 Минимальная ошибка: 2,165691 Средняя ошибка: 11,59712904 Среднеквадратичная ошибка: 10733,392310847
0,1	Максимальная ошибка: 17,447951 Минимальная ошибка: 0,02191999999 Средняя ошибка: 4,3755617810219 Среднеквадратичная ошибка: 2116,10844434154
0,5	Максимальная ошибка: 18,811113 Минимальная ошибка: 0,203837 Средняя ошибка: 5,30231670802 Среднеквадратичная ошибка: 2756,81088118633
1,0	Максимальная ошибка: 16,189081 Минимальная ошибка: 0,148908 Средняя ошибка: 5,3890004598 Среднеквадратичная ошибка: 2646,28935286

Увеличение скорости обучения во всех случаях ведет к снижению средней и среднеквадратичной ошибки. Наилучшие результаты наблюдаются при скорости 1,0.

2.6 Влияние количества нейронов в скрытом слое на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000.

Нормализация: [0;1].

Выбор примеров: случайный.

Крутизна функции: 1.

Смещение: 1.

Скорость обучения: 1.

В таблице 6 показаны результаты исследования.

Таблица 6 – Влияние количества нейронов в скрытом слое на эффективность алгоритма ОРО

Количество нейронов в скрытом слое	Результат
1	Максимальная ошибка: 22,028578 Минимальная ошибка: 0,02193 Средняя ошибка: 6,3331904 Среднеквадратичная ошибка: 3951,444869943
5	Максимальная ошибка: 16,189081 Минимальная ошибка: 0,148908 Средняя ошибка: 5,3890004598 Среднеквадратичная ошибка: 2646,28935286
10	Максимальная ошибка: 17,310105 Минимальная ошибка: 0,02306 Средняя ошибка: 5,55307477372263

Приоритет был отдан количеству нейронов в скрытом слое, равному 5.

2.7 Влияние момента на эффективность алгоритма ОРО

Число входов: 3.

Число выходов: 1.

Циклов обучения: 1000.

Нормализация: [0;1].

Выбор примеров: случайный.

Крутизна функции: 1.

Смещение: 1.

Скорость обучения: 1.

Количество нейронов в скрытом слое: 5.

В таблице 7 показаны результаты исследования.

Таблица 7 – Влияние момента на эффективность алгоритма ОРО

Момент	Результат
Без момента	Максимальная ошибка: 16,189081 Минимальная ошибка: 0,148908 Средняя ошибка: 5,3890004598 Среднеквадратичная ошибка: 2646,28935286
0,1	Максимальная ошибка: 17,091992 Минимальная ошибка: 0,16309 Средняя ошибка: 4,64099294693 Среднеквадратичная ошибка: 2353,293484
0,5	Максимальная ошибка: 46,99824 Минимальная ошибка: 5,988492 Средняя ошибка: 22,1473975401 Среднеквадратичная ошибка: 45280,0604049
1,0	Максимальная ошибка: 41,099726 Минимальная ошибка: 0,00044 Средняя ошибка: 24,8340739 Среднеквадратичная ошибка: 53982,802417522

Наилучшие результаты наблюдаются при обучении без момента и обучении с моментом 0,1. Наименьшие максимальная ошибка наблюдаются при обучении без момента, а наименьшие средняя и среднеквадратическая ошибка – при обучении с моментом 0,1.

3 Ручной расчет

Для более подробного изучения алгоритма ОРО в режиме трассировки был сделан один проход (включающий прямое и обратное распространение), а затем те же самые действия были произведены вручную.

На рисунке 1 показана трассировка первого прохода при обучении сети.

На рисунке 2 показана структура сети.

Инициализация весов синапсов случайным образом...

Нейрон[1][1]
w[1, 1, 1] = 0,696
w[1, 1, 2] = -0,524
w[1, 1, 3] = 0,542
Вес смещения:
w[1, 1, 4] = 1

Нейрон[1][2]
w[1, 2, 1] = 0,406
w[1, 2, 2] = 0,194
w[1, 2, 3] = -0,998
Вес смещения:
w[1, 2, 4] = 1

Нейрон[1][3]
w[1, 3, 1] = -0,286
w[1, 3, 2] = -0,412
w[1, 3, 3] = -0,426
Вес смещения:
w[1, 3, 4] = 1

Нейрон[1][4]
w[1, 4, 1] = 0,492
w[1, 4, 2] = 0,708
w[1, 4, 3] = -0,54
Вес смещения:
w[1, 4, 4] = 1

Нейрон[1][5]
w[1, 5, 1] = -0,484
w[1, 5, 2] = 0,872
w[1, 5, 3] = -0,98
Вес смещения:
w[1, 5, 4] = 1

Нейрон[2][1]
w[2, 1, 1] = -0,356
w[2, 1, 2] = -0,388
w[2, 1, 3] = -0,188
w[2, 1, 4] = 0,416
w[2, 1, 5] = 0,972
Вес смещения:
w[2, 1, 6] = 1

Выбираем допустимый образ из обучающего множества...
0,953191
0,870213
0,97234
0,852979

Подаем сигнал на вход нейронной сети...

Нейрон[0][1]
Аксон = 0,953191

Нейрон[0][2]
Аксон = 0,870213

Нейрон[0][3]
Аксон = 0,97234

Прямая волна...

Нейрон[1][1]
Взвешенная сумма = 1,734437604
Аксон = 0,8499791588

Нейрон[1][2]
Взвешенная сумма = 0,585421548
Аксон = 0,642313948

Нейрон[1][3]
Взвешенная сумма = -0,045357222
Аксон = 0,4886626381

Нейрон[1][4]
Взвешенная сумма = 1,560017176
Аксон = 0,8263558176

Нейрон[1][5]
Взвешенная сумма = 0,344588092
Аксон = 0,585304593

Нейрон[2][1]
Взвешенная сумма = 1,269001116
Аксон = 0,6535092485

Обратная волна - подсчет локальной ошибки
нейронов...

```

Подсчет локальной ошибки нейронов на выходе
нейронной сети...
Желаемый сигнал на выходе:
0,852979
Прогнозируемый сигнал на выходе нейронной сети:
0,6535092485

Нейрон[2][1]
Локальная ошибка = -0,02258345768

Подсчет локальной ошибки нейронов в скрытых слоях
нейронной сети...

Нейрон[1][1]
Локальная ошибка = 0,001025180431

Нейрон[1][2]
Локальная ошибка = 0,002013128604

Нейрон[1][3]
Локальная ошибка = 0,001060876788

Нейрон[1][4]
Локальная ошибка = -0,001348065807

Нейрон[1][5]
Локальная ошибка = -0,005328044684

Коррекция весов синапсов...

w[1, 1, 1] = 0,6970251804
w[1, 1, 2] = -0,5229748196
w[1, 1, 3] = 0,5430251804
Вес смещения:
w[1, 1, 4] = 1,00102518

w[1, 2, 1] = 0,4080131286
w[1, 2, 2] = 0,1960131286
w[1, 2, 3] = -0,9959868714
Вес смещения:
w[1, 2, 4] = 1,002013129

w[1, 3, 1] = -0,2849391232
w[1, 3, 2] = -0,4109391232
w[1, 3, 3] = -0,4249391232
Вес смещения:
w[1, 3, 4] = 1,001060877

w[1, 4, 1] = 0,4906519342
w[1, 4, 2] = 0,7066519342
w[1, 4, 3] = -0,5413480658
Вес смещения:
w[1, 4, 4] = 0,9986519342

w[1, 5, 1] = -0,4893280447
w[1, 5, 2] = 0,8666719553
w[1, 5, 3] = -0,9853280447
Вес смещения:
w[1, 5, 4] = 0,9946719553

w[2, 1, 1] = -0,3785834577
w[2, 1, 2] = -0,4105834577
w[2, 1, 3] = -0,2105834577
w[2, 1, 4] = 0,3934165423
w[2, 1, 5] = 0,9494165423
Вес смещения:
w[2, 1, 6] = 0,9774165423

```

Рисунок 1 – Трассировка первого прохода

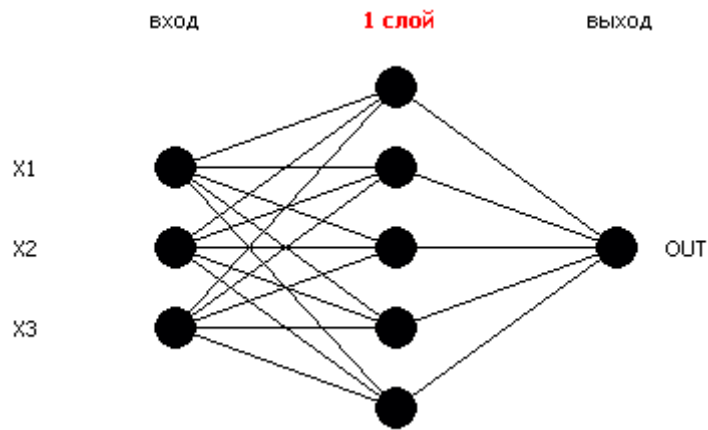


Рисунок 2 – Структура сети

В качестве активационной функции взята

$$F(S_i) = \frac{1}{1 + e^{-\alpha S_i}}, \quad (1)$$

где S_i – взвешенная сумма входов i -го нейрона (с учетом смещения);

α – коэффициент крутизны. Для скрытого слоя $\alpha = 1$, для выходного $= 0,5$.

В таблице 8 показан расчет прямой волны.

№ слоя	№ нейрона	№ выхода	Входной сигнал x_j	Весовой коэффициент w_{ij}	Смещение w_{i0}	Вес смещения	$w_{ij} \cdot x_j$	Взвешенная сумма S_i	Выход нейрона $y_i = F(S_i)$
Вход	1	1	0,953191	-	-	-	-	-	0,953191
	2	1	0,870213	-	-	-	-	-	0,870213
	3	1	0,97234	-	-	-	-	-	0,97234
1	1	1	0,953191	0,696	0,5	1	0,663420936	1,734437604	0,8499791588
		2	0,870213	-0,524			-0,455991612		
		3	0,97234	0,542			0,52700828		
	2	1	0,953191	0,406	0,5	1	0,386995546	0,585421548	0,642313948
		2	0,870213	0,194			0,168821322		
		3	0,97234	-0,998			-0,97039532		
	3	1	0,953191	-0,286	0,5	1	-0,272612626	-0,045357222	0,488662638
		2	0,870213	-0,412			-0,358527756		
		3	0,97234	-0,426			-0,41421684		
	4	1	0,953191	0,492	0,5	1	0,468969972	1,560017176	0,826355818
		2	0,870213	0,708			0,616110804		
		3	0,97234	-0,54			-0,5250636		
	5	1	0,953191	-0,484	0,5	1	-0,461344444	0,344588092	0,585304593
		2	0,870213	0,872			0,758825736		
		3	0,97234	-0,98			-0,9528932		
Выход	1	1	0,8499791588	-0,356	0,5	1	-0,302592581	1,269001116	0,653509249
		2	0,642313948	-0,388			-0,249217812		
		3	0,488662638	-0,188			-0,091868576		
		4	0,826355818	0,416			0,34376402		
		5	0,585304593	0,972			0,568916064		
		93							

Для расчета ошибок необходимо найти производную функции (1) по S_i :

$$F'(S_i) = \frac{\alpha e^{-aS_i}}{(e^{-aS_i} + 1)^2}. \quad (2)$$

Тогда ошибка единственного нейрона выходного слоя будет найдена как

$$\gamma = (y - t) * F'(S),$$

где y – фактическое значение его выхода;

t – желаемый сигнал на выходе.

Ошибка i -го нейрона скрытого слоя будет найдена как

$$\gamma_i = \gamma_j * F'(S_i) * \omega_{ij}.$$

где γ_j – ошибка выходного слоя;

ω_{ij} – синаптическая связь между i -м нейроном скрытого слоя и j -м нейроном выходного слоя.

В таблице 9 показан расчет ошибок.

Таблица 9 – Расчет ошибок

№ слоя	№ нейрона	S_i	$F'(S_i)$	Ошибка
Выход	1	1,269001116	0,113217455	-0,022583458
1	1	1,734437604	0,127514588	0,00102518
	2	0,585421548	0,22974674	0,002013129
	3	-0,045357222	0,249871464	0,001060877
	4	1,560017176	0,14349188	-0,001348066
	5	0,344588092	0,242723126	-0,005328045

Коррекция веса синапса производится по следующей формуле:

$$\omega_{ij}(t + 1) = \omega_{ij}(t) - \alpha * \gamma_j F'(S_j) y_j.$$

Коррекция веса смещения производится по следующей формуле:

$$T_j(t + 1) = T_j(t) - \alpha * \gamma_j F'(S_j).$$

В таблице 10 показан расчет новых весов.

Значения в таблицах 8,9 полностью совпадают со значениями на рис. 1.

Скорректированные веса в таблице 10 совпадают с искомыми с точностью не менее трех знаков после запятой; погрешность можно объяснить ошибками округления и расчетов с плавающей запятой в разных средах.

Таким образом, ручной расчет выполнен верно.

Таблица 10 – Расчет новых весов.

№ слоя	№ нейрона	№ выхода	Предыдущий весовой коэффициент $w_{ij}(t)$	Предыдущий вес смещения $T_j(t)$	Новый весовой коэффициент $w_{ij}(t+1)$	Новый вес смещения $T_j(t+1)$
1	1	1	0,696	1	0,696111114	1,000111114
		2	-0,524		-0,523888886	
		3	0,542		0,542111114	
	2	1	0,406	1	0,405702924	1,000297076
		2	0,194		0,193702924	
		3	-0,998		-0,998297076	
	3	1	-0,286	1	-0,286129536	1,000129536
		2	-0,412		-0,412129536	
		3	-0,426		-0,426129536	
	4	1	0,492	1	0,492159847	0,999840153
		2	0,708		0,708159847	
		3	-0,54		-0,539840153	
	5	1	-0,484	1	-0,483243061	0,999243061
		2	0,872		0,872756939	
		3	-0,98		-0,979243061	
Выход	1	1	-0,356	1	-0,357791114	0,99916454
		2	-0,388		-0,389791114	
		3	-0,188		-0,189791114	
		4	0,416		0,414208886	
		5	0,972		0,970208886	

4 Выводы

В ходе выполнения лабораторной работы был исследован алгоритм ОРО при обучении многослойной НС, а также влияние различных параметров на качество обучения.

Наибольшее влияние на качество обучения показали нормализация, крутизна функции и величина момента. Влияние других параметров были значительно меньше. Случайный выбор примеров улучшил результаты работы, его использование предпочтительно для избегания привыкания сети.

Оптимальные параметры, выбранные для решения данной задачи:

Число входов: 3;

Число выходов: 1;

Циклов обучения: 1000;

Нормализация: [0;1];

Выбор примеров: случайный;

Крутизна функции: 1;

Смещение: 1;

Скорость обучения: 1;

Количество нейронов в скрытом слое: 5.

Работа алгоритма ОРО была проверена ручным расчетом одного прохода. Вычисленные вручную значения примерно совпали с автоматическими. Разница возникла из-за округлений и погрешностей в вычислениях.

Приложение А
(обязательное)
Обучающая выборка

x1	x2	x3	out
-4	-6	1	-41
-3,9	-6	1,1	-41
-3,8	-6	1,2	-41
-3,7	-5,9	1,3	-39,81
-3,6	-5,8	1,4	-38,64
-3,5	-5,7	1,5	-37,49
-3,4	-5,6	1,6	-36,36
-3,3	-5,5	1,7	-35,25
-3,2	-5,4	1,8	-34,16
-3,1	-5,3	1,9	-33,09
-3	-5,2	2	-32,04
-2,9	-5,1	2,1	-31,01
-2,8	-5,1	2,2	-31,01
-2,7	-5,1	2,3	-31,01
-2,6	-5,1	2,4	-31,01
-2,5	-5,1	2,5	-31,01
-2,4	-5,1	2,6	-31,01
-2,3	-5,1	2,7	-31,01
-2,2	-5,1	2,8	-31,01
-2,1	-5,1	2,9	-31,01
-2	-5	3	-30
-1,9	-5	3,1	-30
-1,8	-5	3,2	-30
-1,7	-5	3,3	-30
-1,6	-5	3,4	-30
-1,5	-5	3,5	-30
-1,4	-5	3,6	-30
-1,3	-5	3,7	-30
-1,2	-4,9	3,8	-29,01
-1,1	-4,8	3,9	-28,04
-1	-4,7	4	-27,09
-0,9	-4,6	4,1	-26,16
-0,8	-4,5	4,2	-25,25
-0,7	-4,4	4,3	-24,36
-0,6	-4,3	4,4	-23,49
-0,5	-4,2	4,5	-22,64
-0,4	-4,1	4,6	-21,81
-0,3	-4	4,7	-21
-0,2	-3,9	4,8	-20,21
-0,1	-3,8	4,9	-19,44
2,41E-15	-3,7	5	-18,69
0,1	-3,6	1	-13,86
0,2	-3,5	1,1	-13,15
0,3	-3,4	1,2	-12,46
0,4	-3,3	1,3	-11,79

0,5	-3,2	1,4	-11,14
0,6	-3,1	1,5	-10,51
0,7	-3	1,6	-9,9
0,8	-2,9	1,7	-9,31
0,9	-2,8	1,8	-8,74
1	-2,7	1,9	-8,19
1,1	-2,6	2	-7,66
1,2	-2,5	2,1	-7,15
1,3	-2,4	2,2	-6,66
1,4	-2,3	2,3	-6,19
1,5	-2,2	2,4	-5,74
1,6	-2,1	2,5	-5,31
1,7	-2	2,6	-4,9
1,8	-1,9	2,7	-4,51
1,9	-1,8	2,8	-4,14
2	-1,7	2,9	-3,79
2,1	-1,6	3	-3,46
2,2	-1,5	3,1	-3,15
2,3	-1,4	3,2	-2,86
2,4	-1,3	3,3	-2,59
2,5	-1,2	3,4	-2,34
2,6	-1,1	3,5	-2,11
2,7	-1	3,6	-1,9
2,8	-0,9	3,7	-1,71
2,9	-0,8	3,8	-1,54
3	-0,7	3,9	-1,39
3,1	-0,7	4	-1,39
3,2	-0,7	4,1	-1,39
3,3	-0,6	4,2	-1,26
3,4	-0,5	4,3	-1,15
3,5	-0,4	4,4	-1,06
3,6	-0,3	4,5	-0,99
3,7	-0,2	4,6	-0,94
3,8	-0,1	4,7	-0,91
3,9	0	4,8	-0,9
4	0,1	4,9	-0,91
1	0,2	5	-4,04
1,1	0,3	1	0,01
1,2	0,4	1,1	-0,06
1,3	0,5	1,2	-0,15
1,4	0,6	1,3	-0,26
1,5	0,7	1,4	-0,39
1,6	0,8	1,5	-0,54
1,7	0,9	1,6	-0,71
1,8	1	1,7	-0,9
1,9	1,1	1,8	-1,11
2	1,2	1,9	-1,34
2,1	1,3	2	-1,59
2,2	1,4	2,1	-1,86
2,3	1,5	2,2	-2,15

2,4	1,6	2,3	-2,46
2,5	1,7	2,4	-2,79
2,6	1,8	2,5	-3,14
2,7	1,9	2,6	-3,51
2,8	2	2,7	-3,9
2,9	2,1	2,8	-4,31
3	2,2	2,9	-4,74
3,1	2,3	3	-5,19
3,2	2,4	3,1	-5,66
3,3	2,5	3,2	-6,15
3,4	2,6	3,3	-6,66
3,5	2,7	3,4	-7,19
3,6	2,8	3,5	-7,74
3,7	2,9	3,6	-8,31
3,8	3	3,7	-8,9
3,9	3,1	3,8	-9,51
4	3,2	3,9	-10,14
-4	3,3	4	-18,89
-3,9	3,4	4,1	-19,56
-3,8	3,5	4,2	-20,25
-3,7	3,6	4,3	-20,96
-3,6	3,7	4,4	-21,69
-3,5	3,8	4,5	-22,44
-3,4	3,9	4,6	-23,21
-3,3	4	4,7	-24
-3,2	4,1	4,8	-24,81
-3,1	4,2	4,9	-25,64
-3	4,3	5	-26,49
-2,9	4,4	1	-23,26
-2,8	4,5	1,1	-24,15
-2,7	4,6	1,2	-25,06
-2,6	4,7	1,3	-25,99
-2,5	4,8	1,4	-26,94
-2,4	4,9	1,5	-27,91
-2,3	5	1,6	-28,9
-2,2	5,1	1,7	-29,91
-2,1	5,2	1,8	-30,94
-2	5,3	1,9	-31,99
-1,9	5,4	2	-33,06
-1,8	5,5	2,1	-34,15
-1,7	5,6	2,2	-35,26
-1,6	5,7	2,3	-36,39
-1,5	5,8	2,4	-37,54
-1,4	5,9	2,5	-38,71
-1,3	6	2,6	-39,9