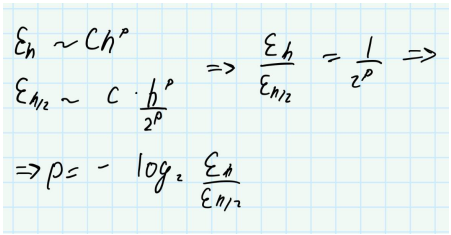
Методы численного интегрирования

i.vdovin1

October 2022

Составим таблицы для каждого метода.
Шагом считаем увеличение количества точек на прямой в 2 раза. Очевидно, что чем больше точек, тем ближе мы подходим к определению интеграла (диаметр разбиения становится всё ближе к 0).
Точность (Эпсилон) считаем как разность интегралов, полученных на предыдущем шаге и на текущем.
Таблица имеет 4 колонки: N - количество точек разбиения отрезка, eps - разница с прошлым интегралом(Мы начинаем как бы со 2 шага, поэтому она определена уже на 1 шаге), p - скорость сходимости(чем она выше, тем лучше и тем более хороший этот метод в общем случае). Можно найти с помщью метода, пописанного на паре через выражение 2 эпсилон:



И последний столбец - само значение интеграла на каждом шаге

1 Метод прямоугольников

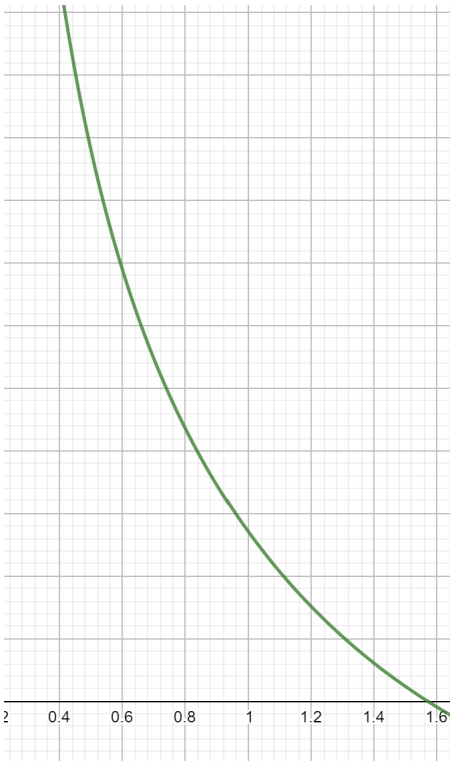
N	eps	p	integral
4	5.81658e-05	null	0.199001
8	1.02531e-05	-2.5041043915764525	0.198991
16	1.8893e-06	-2.440140588649939	0.198989
32	4.06137e-07	-2.217814038375686	0.198988
64	9.42618e-08	-2.107221848187764	0.198988
128	2.27136e-08	-2.053120060180647	0.198988
256	5.57531e-09	-2.0264311029980306	0.198988

На данной таблице видно, что p сходится к -2. Это для eps = 1e-8.
Однако, я заметил, что если уменьшить эпсилон, то метод перестаёт сходиться и скачет где-то в -15, -14 степени. вот так:
Ну и на 28 итерации у меня получается 2^{28} точек, ну и очевидно, что на этом этапе компьютеру не хватает вычислительных

```
14 1.336375454741301e-12
15 3.3603675397841926e-13
16 7.979727989493313e-14
17 2.2315482794965646e-14
18 6.5503158452884236e-15
19 7.993605777301127e-15
20 1.0574874309554616e-14
21 9.18709552877317e-15
22 3.552713678800501e-15
23 1.3183898417423734e-14
24 3.630429290524262e-14
25 1.715294573045867e-14
26 1.2934098236883074e-14
27 1.5182299861749016e-14
28 4.18554080283684e-14
```

мощностей
По логике, чем меньше мы берём промежутки, тем меньше отличаются интегралы. Так почему так происходит? Моё

предположение, что функция растёт неравномерно, в каких-то местах мы уже достигли нужной точности, а в каки-то нет.



2 Метод трапеций

Тут ничего необычного, метод аналогичен методу прямоугольников: Сходимость схожа с методом Прямоугольников.

N	eps	p	integral
4	0.000157463	null	0.198962
8	2.10153e-05	-2.9055045931319627	0.198983
16	3.79558e-06	-2.4690460780380428	0.198987
32	8.13074e-07	-2.222862607395268	0.198988
64	1.88567e-07	-2.1083069512584207	0.198988
128	4.54297e-08	-2.0533729309324262	0.198988
256	1.11508e-08	-2.026492319472249	0.198988
512	2.76231e-09	-2.0131979648309164	0.198988

Имеется та же проблема со сходимостью для высокой точности

3 Метод Симпсона

Тут ситуация интереснее, метод сходится за значительно меньшее количество итераций

N	eps	p	integral
4	1.37106e-05	null	0.198988
8	1.69668e-07	-6.336432323003417	0.198988
16	5.65922e-09	-4.905968328985275	0.198988

За счёт скорости сходимости, тут нет проблем с высокой точностью. И для eps = 1e-16, выдается следующая таблица:
Тут видно, что p сходится к -4. Значения p, появляющиеся в конце и выбивающиеся из общей какртины обуславливаются

N	eps	p	integral
4	1.37106e-05	null	0.198988
8	1.69668e-07	-6.336432323003417	0.198988
16	5.65922e-09	-4.905968328985275	0.198988
32	2.66458e-10	-4.4086207288018215	0.198988
64	1.45493e-11	-4.194887358032029	0.198988
128	8.51347e-13	-4.095059657777784	0.198988
256	5.13201e-14	-4.052152147720504	0.198988
512	3.35842e-15	-3.933666272129601	0.198988
1024	1.66533e-16	-4.3339007365534385	0.198988
2048	5.27356e-16	1.6629650127224294	0.198988
4096	8.04912e-16	0.6100534816839867	0.198988
8192	8.32667e-17	-3.273018494406416	0.198988

достижением высокой точности, где эпсилон уже почти перестаёт меняться. По приколу (и чтобы проверить поведение p) я решил поставить функцию, рост которой сильно выше, но поведение знаков не отличается от исходной

N	eps	p	integral
16	6.48541e-05	-4.6374894535462445	0.976921
32	3.18523e-06	-4.347726708115894	0.976918
64	1.75608e-07	-4.180968129642718	0.976918
128	1.02976e-08	-4.091972817821933	0.976918
256	6.23273e-10	-4.04630681521868	0.976918
512	3.83322e-11	-4.02323423862722	0.976918
1024	2.37677e-12	-4.011486345876932	0.976918
2048	1.48992e-13	-3.9956934444412777	0.976918
4096	8.54872e-15	-4.123382415505282	0.976918
8192	6.66134e-16	-3.681824039973745	0.976918

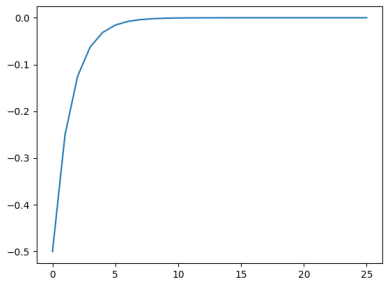
Тут явно видно, что p сходится к 4. И так же при достижении высокой точности наблюдаются скачки

4 Производная

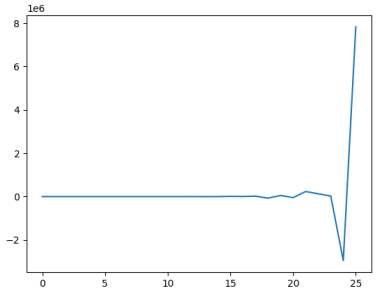
Функкцию я считаю исходя из стандартного определения производной в точке, а именно $\lim_{\Delta\to} \frac{(f(x+\Delta)-f(x))}{\Delta}$. Функция у которой я беру производную - x^2, в точке x = 1 (Потому что просто считать и не является константой после дифференцирования). Если у нас имеются все данные и они точные, то получим следующее:
Отлично сходится при увеличении Δ

f	eps
3	-1
2.5	-0.5
2.25	-0.25
2.125	-0.125
2.0625	-0.0625
2.03125	-0.03125
2.01562	-0.015625
2.00781	-0.0078125
2.00391	-0.00390625

2.00391	-0.00390625
2.00195	-0.00195312
2.00098	-0.000976562
2.00049	-0.000488281
2.00024	-0.000244141
2.00012	-0.00012207
2.00006	-6.10352e-05
2.00003	-3.05176e-05
2.00002	-1.52588e-05
2.00001	-7.62939e-06
2	-3.8147e-06
2	-1.90735e-06
2	-9.53674e-07
2	-4.76837e-07
2	-2.38419e-07
2	-1.19209e-07
2	-5.96046e-08
2	-2.98023e-08
2	-1.49012e-08

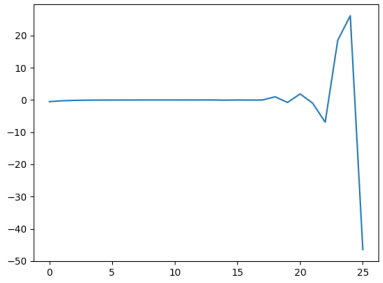


Однако, при добавленнии случайной погрешности в данные (значения функции). Ситуация меняется и начиная с некоторого Δ становится некорректной. Подберём оптимальный Δ Для разных данных. В 1 случае погрешность достаточно большая находится от [-0.1; 0.1]
Тут видно, что для большой погрешности eps становится большим уже с 3 шага.



f	eps
3.11716	-1.11716
2.51949	-0.519493
2.47205	-0.472053
1.27684	0.723163
5.12121	-3.12121
2.63126	-0.631262
-1.18445	3.18445
-19.222	21.222
-43.4595	45.4595
-36.1234	38.1234

Рассмотрим ещё вариант с более адекватной погрешностью, а именно [-0,000001; 0,000001] Тут мы видим следующее:



3	-0.999999
2.5	-0.499999
2.25	-0.250002
2.12499	-0.124988
2.06251	-0.0625059
2.03125	-0.0312484
2.01563	-0.0156271
2.00769	-0.00768558
2.00421	-0.0042066
2.0018	-0.00179617
2.00233	-0.00233358
2.00003	-3.19634e-05
1.99904	0.000964206
1.99624	0.00376266
1.98699	0.013015
2.04923	-0.0492294

Видно, что метод начинает расходиться на 12 шаге. Таким образом, можно сделать вывод, что скачки эпсилон надо отслеживать. И после их появления выходить из функции и выводить оптимальное значение