

WSI - ćwiczenie 1.

Zagadnienie przeszukiwania i podstawowe podejścia do niego

Kornelia Błaszczuk

Numer indeksu: 331361

Decyzje projektowe

- Gradient liczony jest przy użyciu różnic skośnych, gdzie epsilon używany jest jako wielkość przesunięcia w obliczeniu gradientu.
- Rozmiar kroku jest stały w czasie całej iteracji.
- Dodana została normalizacja wektora gradientu, zachodząca dla takiego, którego rozmiar wynosi co najmniej 2.
- W funkcji *solve* klasy *GradientDescent* znajduje się parametr określający maksymalną ilość iteracji. Jego obecność tam, a nie jako inny hiperparametr, jest uzasadniona możliwością stworzenia tylko jednego obiektu klasy *GradientDescent*, przy możliwości dobrania liczby maksymalnej iteracji do potrzeb danego problemu.

Hiperparametry

- **Epsilon** - Minimalna zmiana wartości funkcji; poniżej jej algorytm przerywa działanie.
- **Learning Rate** - Współczynnik uczenia (rozmiar kroku gradientu).
- **Normalize Threshold** - Gdy gradient przekracza ten próg, zostaje znormalizowany.

Eksperymenty

Przeprowadzone przeze mnie eksperymenty miały na celu sprawdzenie wpływu różnych danych wejściowych na otrzymywaną wartość minimum danej funkcji oraz ilość iteracji potrzebnych do jej uzyskania.

Dla każdego eksperymentu został przygotowany zestaw dwóch punktów startów na każdą funkcję.

Wpływ hiperparametrów

Poniżej zaprezentowane zostały wyniki eksperymentów związanych z wartością parametrów. Startowe wartości hiperparametrów:

Epsilon	1 e^{-4}
Learning Rate	1 e^{-2}
Normalize Threshold	1 e^{-2}

* Max Iterations domyślnie wynosi 1e^4

Wpływ punktów początkowych

Mamy dwie funkcję

$$f(x) = 0.5x^4 + x,$$

która ma minimum dla $x = -0.79369896$ o wartości $f(x) = -0.5952753944834231$

$$g(x) = 1 - 0.6 \exp\{-x_1^2 - (x_2 + 1)^2\} - 0.4 \exp\{-(x_1 - 1.75)^2 - (x_2 + 2)^2\},$$

która ma minimum dla $x = [0.02195075, -1.01254615]$ o wartości $g(x) = 0.39276746842424715$

➔ Wartość spodziewanego minimum została przeze mnie obliczona przy pomocy biblioteki [SciPy](#).

	Wartość x_{\min}	Znaleziona wartość minimum	Różnica względem minimum spodziewanego	Iteracje
$x_0 = [10.0]$ funkcji $f(x)$ *	-1.64905038e+27	3.6974788264558765e+108	3.6974788264558765e+108	9
$x_0 = [8.0]$ funkcji $f(x)$	-0.79630341	-0.5952625624787795	1.283200920154659e-05	134
$x_0 = [7.897]$ funkcji $f(x)$	-0.79626284	-0.5952629598296053	1.243465837574842e-05	133
$x_0 = [6.0]$ funkcji $f(x)$	-0.79113467	-0.5952629790040236	1.241548395747305e-05	275
$x_0 = [5.671]$ funkcji $f(x)$	-0.79110448	-0.5952626855121215	1.2708975859587746e-05	277
$x_0 = [4.0]$ funkcji $f(x)$	-0.7910876	-0.5952625198573047	1.2874630676340537e-05	279
$x_0 = [3.1891]$ funkcji $f(x)$	-0.79112446	-0.5952628801387082	1.2514349272851888e-05	279
$x_0 = [2.0]$ funkcji $f(x)$	-0.79111123	-0.5952627514349795	1.264305300152646e-05	276
$x_0 = [1.87]$ funkcji $f(x)$	-0.79108519	-0.5952624960613275	1.2898426653529249e-05	275
$x_0 = [0.0]$ funkcji $f(x)$	-0.79105202	-0.5952621672539298	1.3227234051282899e-05	187
$x_0 = [-0.8]$ funkcji $f(x)$	-0.79628473	-0.5952627462459859	1.2648241995205645e-05	23
$x_0 = [-2.0]$ funkcji $f(x)$	-0.7962852	-0.5952627416354315	1.2652852549588367e-05	132
$x_0 = [-3.1]$ funkcji $f(x)$	-0.79628608	-0.5952627329579326	1.2661530048418967e-05	135
$x_0 = [-4.54]$ funkcji $f(x)$	-0.79632344	-0.5952623640878276	1.3030400153435373e-05	135
$x_0 = [-5.0]$ funkcji $f(x)$	-0.79628502	-0.5952627434064812	1.2651081499837069e-05	135
$x_0 = [-6.0]$ funkcji $f(x)$	-0.796242	-0.5952631615009842	1.2232986996818518e-05	131

$x_0 = [-7.35]$ funkcji $f(x)$	-0.79112585	-0.5952628935928601	1.2500895120948208e-05	242
$x_0 = [-8.53]$ funkcji $f(x)$	-0.79108923	-0.5952625358855536	1.2858602427501253e-05	280
$x_0 = [-9.0]$ funkcji $f(x)$	-0.791062	-0.5952622665623124	1.3127925668654505e-05	278
$x_0 = [-10.0]$ funkcji $f(x)$	-0.79105095	-0.5952621565102769	1.3237977704139148e-05	282
$x_0 = [-15.0]$ funkcji $f(x)^*$	-1.92284249e+24	6.835099509867542e+96	6.835099509867542e+96	4
$x_0 = [2.0, -3.0]$ funkcji $g(x)$	[1.70069158, -1.97491343]	0.5883623923231265	0.19559492389887934	107
$x_0 = [1.75, -4.0]$ funkcji $g(x)$	[1.69829494, -1.97370378]	0.5883479170247108	0.19558044860046364	203
$x_0 = [0.6812, 1.7812]$ funkcji $g(x)$ **	[0.6812, 1.7812]	0.9998349817930005	0.6070675133687533	0
$x_0 = [1.0, 1.0]$ funkcji $g(x)$	[0.02490982, -1.00547592]	0.39280305766602014	3.558924177299305e-05	223
$x_0 = [1.5, 0.0]$ funkcji $g(x)$	[0.02768141 -1.01227142]	0.39278602911352767	1.8560689280522347e-05	181
$x_0 = [4.5, 0.75]$ funkcji $g(x)$ **	[4.5, 0.75]	0.999998919718134	0.6072324235475662	0
$x_0 = [0.9987, 0.9127]$ funkcji $g(x)$	[0.02139649, -1.01353108]	0.3927682474834984	7.79059251276859e-07	216
$x_0 = [100.0, 0.97]$ funkcji $g(x)$ ***	[100.0, 0.97]	1.0	0.6072325315757529	0
$x_0 = [0.0, 0.0]$ funkcji $g(x)$	[0.02074223 -1.00941198]	0.39277388846064754	6.420036400389684e-06	101
$x_0 = [-1.0, -1.0]$ funkcji $g(x)$	[0.01970581, -1.01181524]	0.39277051814085656	3.0497166094156647e-06	102
$x_0 = [-0.8, -2.0]$ funkcji $g(x)$	[0.01826203, -1.01585289]	0.392782228504916	1.4760080669196096e-05	128

$x_0 = [0.02, -1.0]$ funkcji g(x)	[0.02103732, -1.00994605]	0.39277180059912314	4.332174875998351e-06	1
$x_0 = [0.871, -1.2]$ funkcji g(x)	[0.02971918, -1.01604635]	0.3928071447861231	3.967636187596213e-05	87
$x_0 = [0.0198, -7.0]$ funkcji g(x)	[0.0198, -7.0]	0.9999999999997216	0.6072325315754744	0

* Widzimy, że dla $x_0 = 10.0$ i $x_0 = 15.0$ algorytm zaczyna się rozbiegać, co powoduje wynik przedstawiony w tabelce. Rozwiązaniem jest zmniejszenie wartości learning rate (z $1e-2$ na $1e-3$). Wtedy otrzymamy poniższy rezultat:

$x_0 = 10.0$

Znalezione minimum funkcji f: [-0.76632798]

Liczba iteracji dla funkcji f: 2219

Znaleziona wartość minimum: -0.5938916644345763

$x_0 = -15.0$

Znalezione minimum funkcji f: [-0.81929545]

Liczba iteracji dla funkcji f: 815

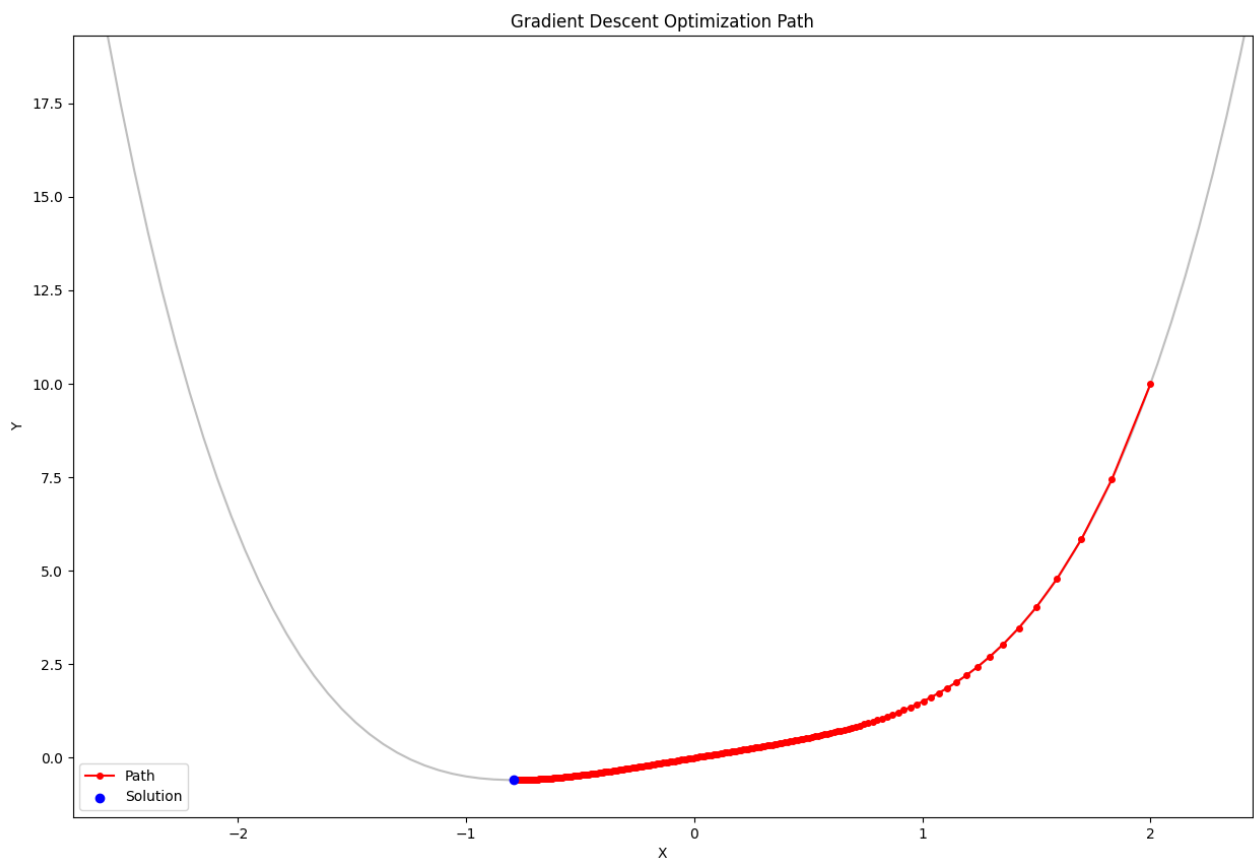
Znaleziona wartość minimum: -0.5940105017910143

Widać, że rezultatem jest otrzymanie poprawnego wyniku kosztem znacznie większej liczby iteracji.

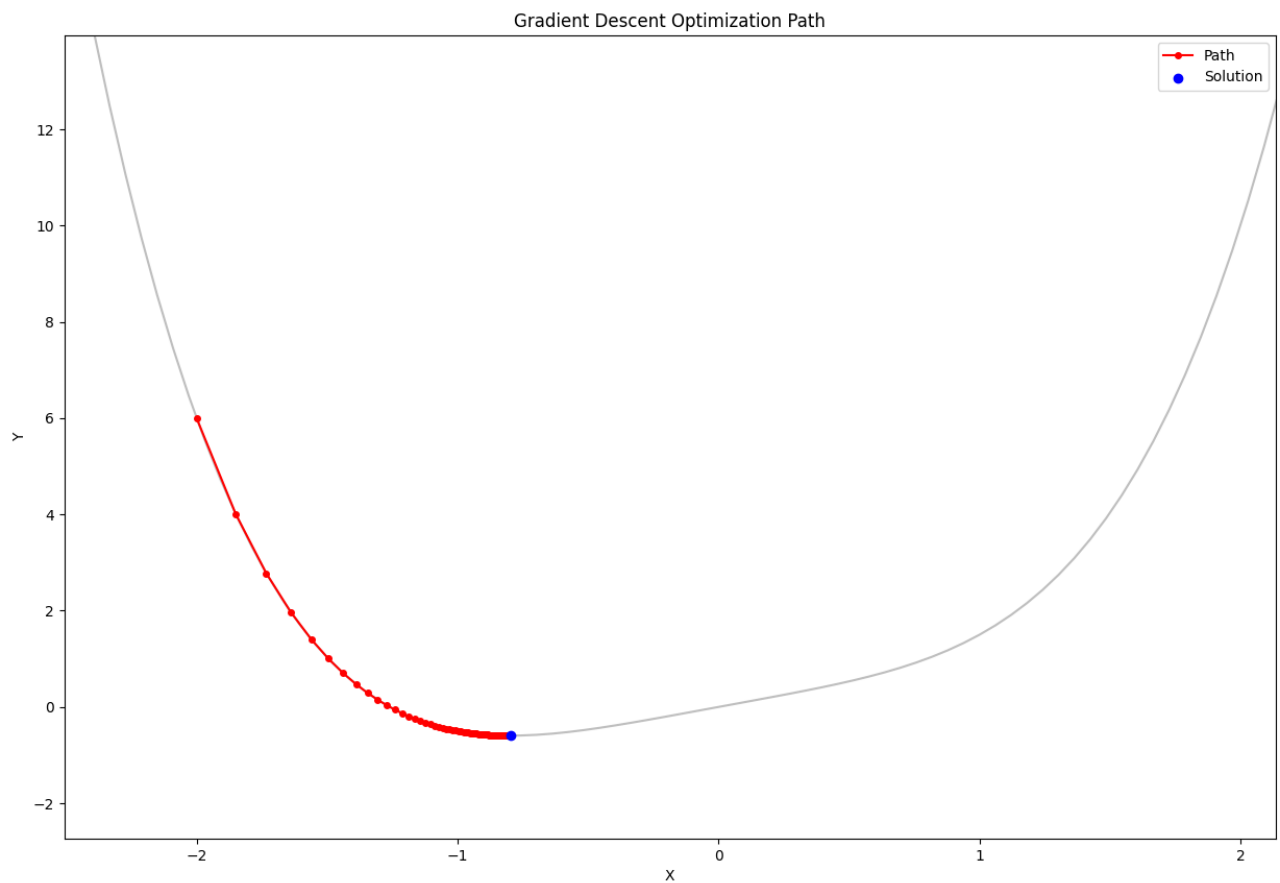
** W przypadku dla $x_0 = [0.6812, 1.7812]$ i $x_0 = [4.5, 0.75]$ dla funkcji g nie ma żadnych iteracji. Jest to spowodowane wartością epsilon, która jest większa niż gradient, więc algorytm się kończy.

*** Dla $x_0 = [100.0, 0.97]$ i $x_0 = [0.0198, -7.0]$ dla funkcji g algorytm ma 0 iteracji, dlatego, że gradient w tym punkcie wynosi 0. Jest to spowodowane tym, że funkcja g(x) jest w tej okolicy prawie stała, funkcja jest płaska.

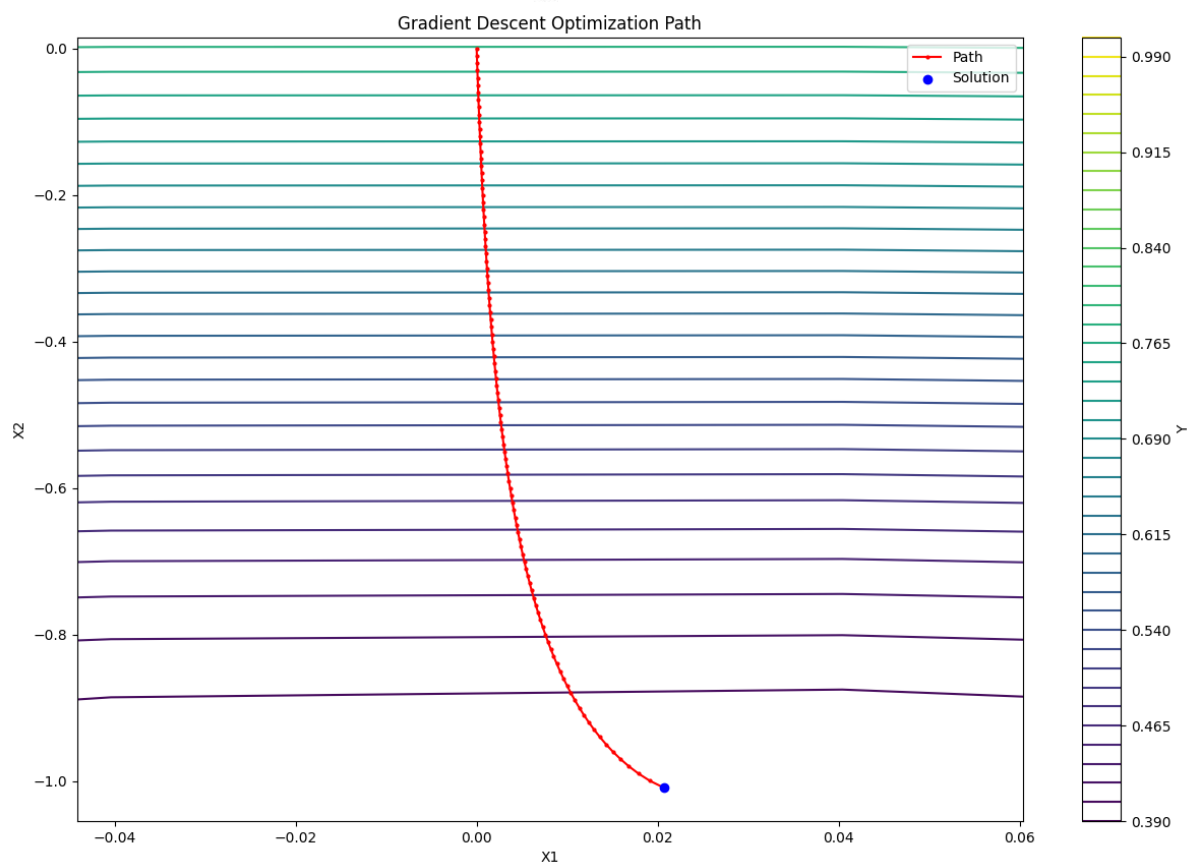
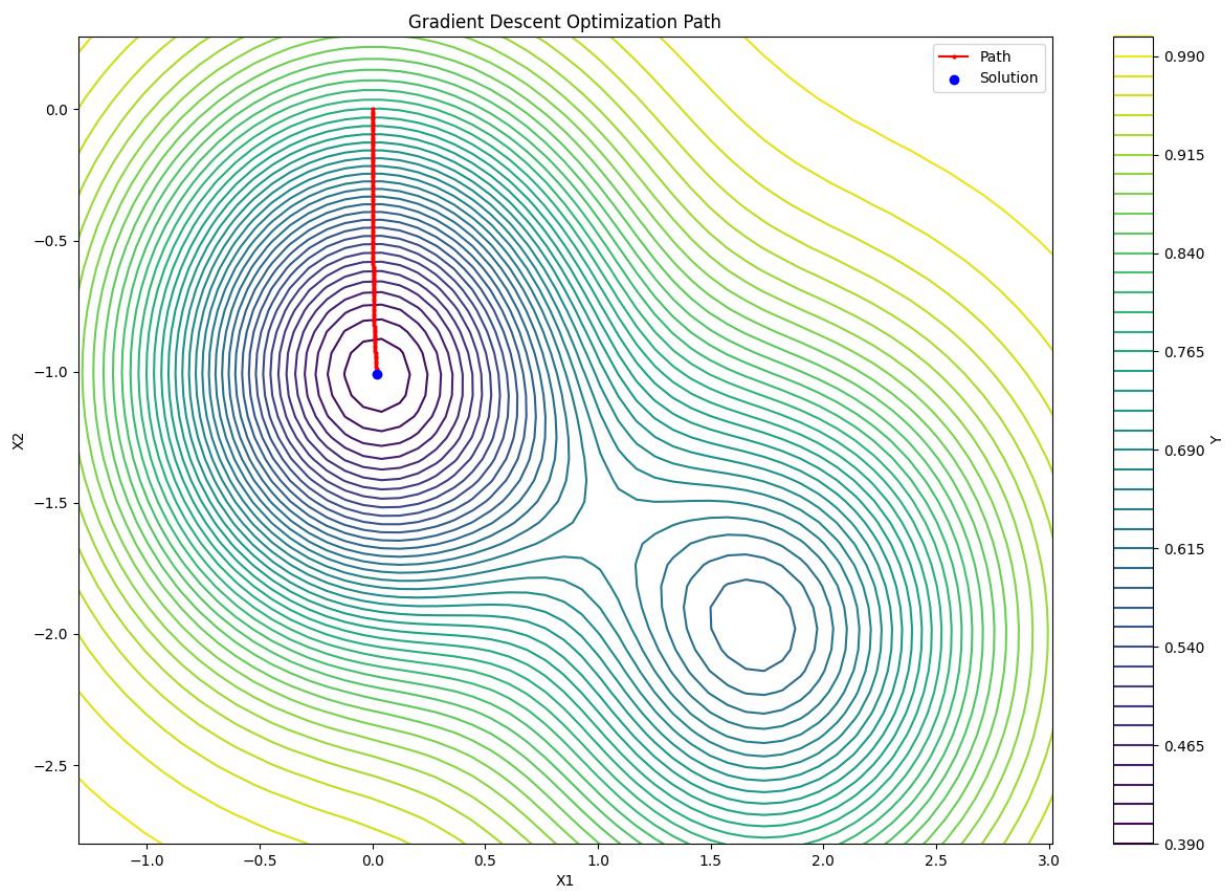
Wykres $f(x)$ dla $x_0 = 2.0$:



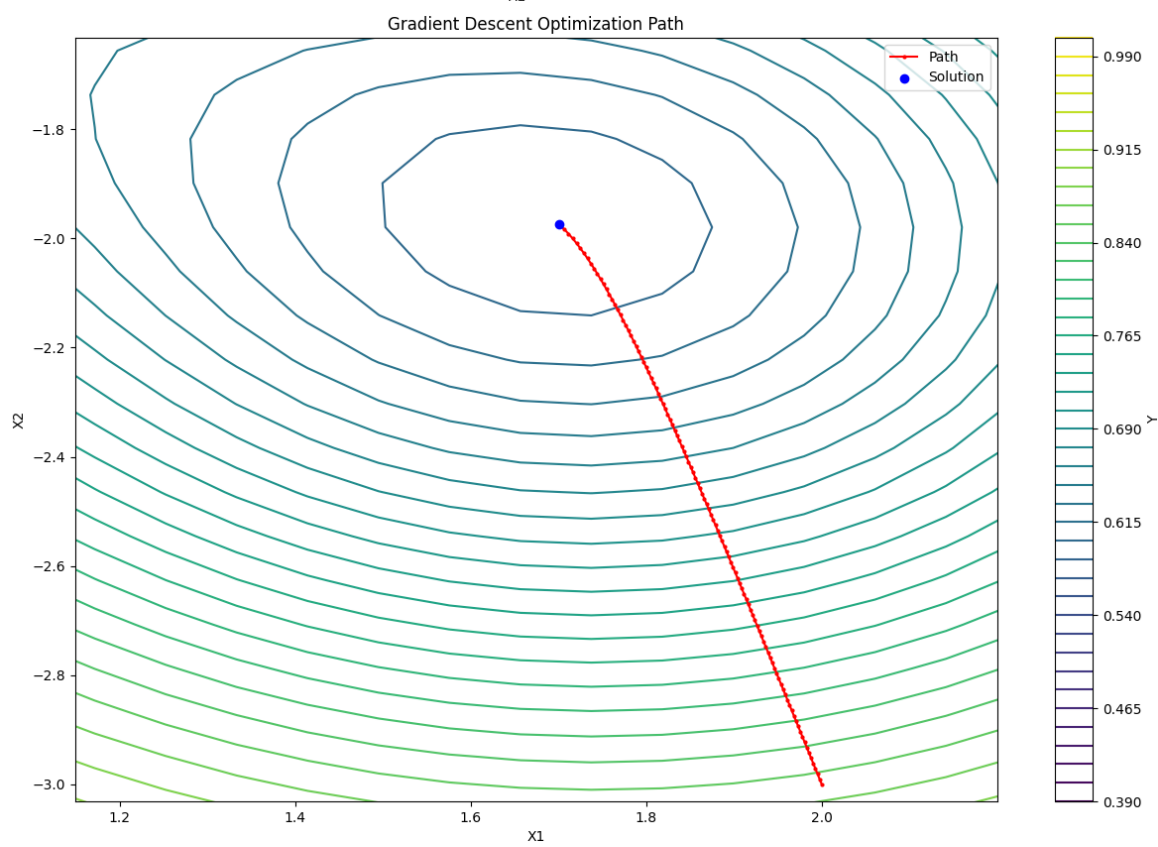
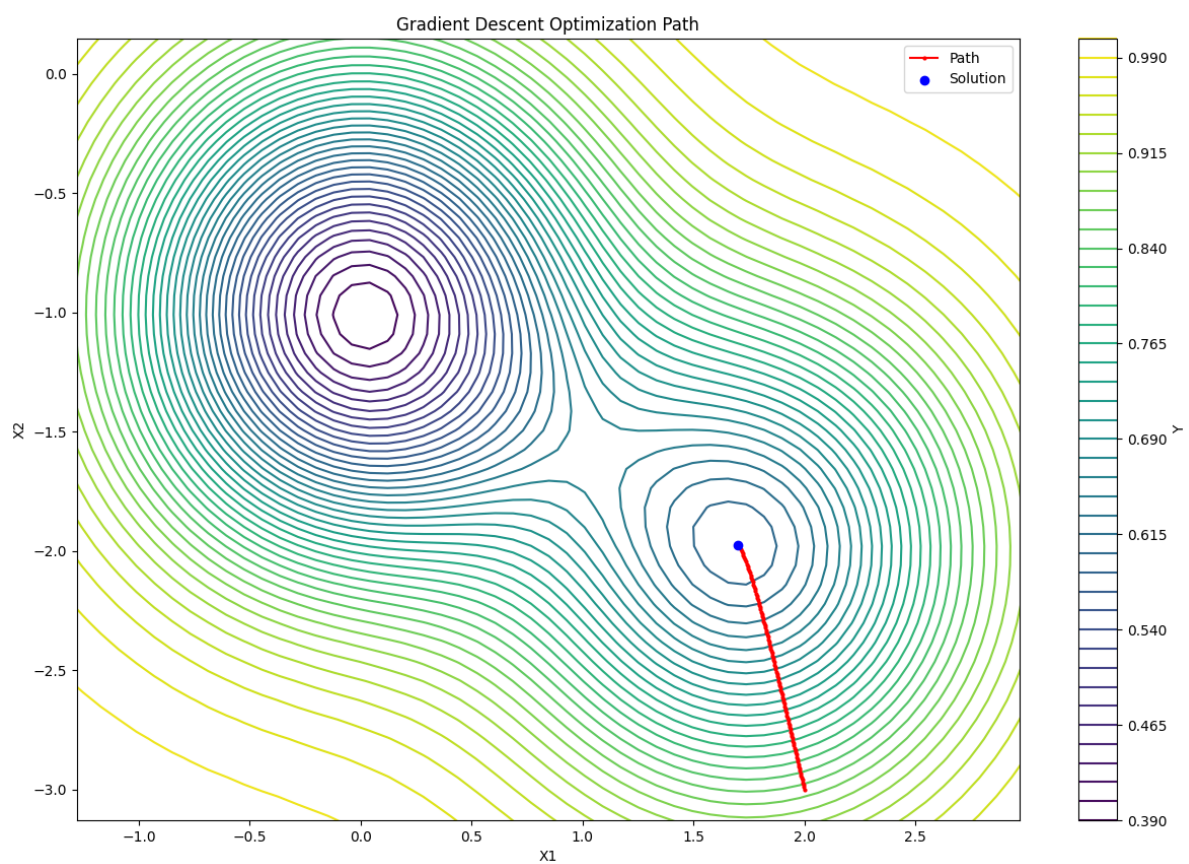
Wykres $f(x)$ dla $x_0 = -2.0$



Wykresy $g(x)$ dla $x_0 = [2.0, -3.0]$



Wykresy $g(x)$ dla $x_0 = [0.0, 0.0]$



Widzimy, że postać funkcji g może powodować wpadnięcie w minimum lokalne (siodółka), gradient nie jest w stanie się z niego wydostać. Co więcej jest to funkcja w znacznej mierze płaska, co powoduje wartość gradientu bliską zero, a więc niemożność znalezienia minimum.

Dla funkcji f dla losowych wartości punktu początkowego algorytm znajduje minimum globalne. Jednak z racji, że funkcja ta bardzo szybko rośnie, jej ramiona są blisko osi OY , to przy punktach startowych, które są dalej od minimum, należy dostosować, zmniejszyć wielkość kroku, aby nie miało miejsce przeskalowanie.

Epsilon (Minimalna zmiana wartości funkcji)

$$f(x) = 0.5x^4 + x$$

Epsilon	Wartość x_{\min} dla $x_0 = [2.0]$	Znalezione minimum	Różnica względem minimum spodziewanego	Iteracje
1e-3	-0.7663837	-0.5938972287462	0.0013781657418542315	214
1e-4	-0.7911112	-0.59526275143497	1.264305300152646e-05	276
1e-5	-0.7934432	-0.595275269367832	1.25120149019331e-07	336
1e-6	-0.793675	- 0.5952753932588075	1.229173540728823e-09	396
1e-7	-0.7936979	- 0.5952753944750396	1.29414257088456e-11	455
1e-8	-0.7937002	-0.595275394487947	3.4083846855992306e-14	515
1e-9	-0.7937005	- 0.5952753944880735	9.248157795127554e-14	574
1e-10	-0.79370042	- 0.5952753944880518	7.072120666862247e-14	537
1e-15	-0.77667537	- 0.5947353929128174	0.0005400015751636511	223

Epsilon	Wartość x_{\min} dla $x_0 = [-2.0]$	Znalezione minimum	Różnica względem minimum spodziewanego	Iteracje
1e-3	-0.8185588	-0.5940829985509	0.0011923959371736848	74
1e-4	-0.7962852	-0.5952627416354315	1.2652852549588367e-05	132
1e-5	-0.79395584	-0.5952752712727689	1.23215212144423e-07	192
1e-6	-0.79372681	-0.5952753931826298	1.305351271518873e-09	251
1e-7	-0.79370313	-0.5952753944752603	1.2720713371550119e-11	311
1e-8	-0.79370078	-0.595275394487949	3.2085445411667024e-14	371
1e-9	-0.79370055	-0.5952753944880733	9.225953334635051e-14	430
1e-10	-0.7937007	-0.5952753944880171	3.608224830031759e-14	383
1e-15	-0.80928581	-0.5948103015549289	0.00046509293305219135	82

$$g(x) = 1 - 0.6 \exp\{-x_1^2 - (x_2 + 1)^2\} - 0.4 \exp\{-(x_1 - 1.75)^2 - (x_2 + 2)^2\}$$

Epsilon	Wartość x_{\min} dla $x_0 = [2.0, -3.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1e-3	[1.70069151, -1.97491346]	0.5883623921554586	0.19559492373121146	107
1e-4	[1.70069158, -1.97449134]	0.5883623923231265	0.19559492389887934	107
1e-5	[1.69436617, -1.96846368]	0.5883276424705376	0.19556017404629045	409
1e-6	[1.69322362, -1.9675681]	0.5883268468639683	0.19555937843972115	781
1e-7	[1.69310172, -1.9674872]	0.5883268388360564	0.19555937041180926	1155
1e-8	[1.69308921, -1.96747957]	0.5883268387555798	0.19555937033133264	1530
1e-9	[1.69308805, -1.96747889]	0.588326838754794	0.19555937033054682	1820
1e-10	[1.69308896, -1.96747946]	0.5883268387553309	0.19555937033108373	1559
1e-15	[1.70986122 - 2.02966724]	0.5898268520978625	0.1970593836736153	106

Epsilon	Wartość x_{\min} dla $x_0 = [0.0, 0.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1e-3	[0.02074227, -1.00941198]	0.392773888419858	6.41999561085127e-06	101
1e-4	[0.02074223, -1.00941198]	0.39277388846064754	6.420036400389684e-06	101
1e-5	[0.02157568, -1.01174465]	0.3927679070591522	4.386349050644611e-07	218
1e-6	[0.02190217, -1.01246661]	0.39276747277232116	4.348074011950587e-09	421
1e-7	[0.02194285, -1.01253472]	0.39276746845487437	3.062722297997311e-11	625
1e-8	[0.02194765, -1.01254119]	0.392767468410848	1.3399170661898552e-11	831
1e-9	[0.02194816, -1.01254177]	0.39276746841040777	1.3839374091162426e-11	1004
1e-10	[0.02194797, -1.01254154]	0.39276746841049665	1.3750500738041183e-11	10000 (wartość graniczna)
1e-15	[0.02292822, -0.98537777]	0.3932063800723782	0.0004389116481310351	10000 (wartość graniczna)

Spadek wartości epsilon wiąże się ze wzrostem liczby iteracji. Zauważmy, że wartości otrzymywanych przez nas minimum funkcji w danym punkcie są sobie bliskie. Co ciekawe wyniki najbliższe spodziewanemu minimum globalnemu otrzymujemy dla epsilon rzędu 1e-8/1e-9.

Learning Rate (Rozmiar kroku gradientu)

Przy wartości rozmiaru kroku gradientu rzędu $1e-4$ obliczenia tracą swoją dokładność, nie jest on wartością uniwersalną (dla pewnych punktów początkowych otrzymujemy prawidłowe wartości, ale w większości punkty otrzymane nie są poprawne).

$$f(x) = 0.5x^4 + x$$

Learning Rate	Wartość x_{\min} dla $x_0 = [2.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1e-1	-0.79350976	- 0.5952753257210824	6.87668986198986e-08	26
1e-2	-0.79111123	- 0.5952627514349795	1.264305300152646e-05	276
1e-3	-0.76632321	- 0.5938911873618627	0.0013842071261183087	2163

Learning Rate	Wartość x_{\min} dla $x_0 = [-2.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1e-1	-0.79352403	- 0.5952753356228415	5.8865139540564826e-08	18
1e-2	-0.7962852	- 0.5952627416354315	1.2652852549588367e-05	132
1e-3	-0.81927553	- 0.5940124907022704	0.0012629037857107006	756

$$g(x) = 1 - 0.6 \exp\{-x_1^2 - (x_2 + 1)^2\} - 0.4 \exp\{-(x_1 - 1.75)^2 - (x_2 + 2)^2\}$$

Learning Rate	Wartość x_{\min} dla $x_0 = [2.0, -3.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1e-1	[1.65474017, -2.03458711]	0.5907788730671168	0.1980114046428697	10000 (wartość graniczna)
1e-2	[1.70069158, -1.97491343]	0.5883623923231265	0.19559492389887934	107
1e-3	[1.70310834, -1.97918064]	0.5884027936526326	0.19563532522838545	1065

Learning Rate	Wartość x_{\min} dla $x_0 = [0.0, 0.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1e-1	[0.02968568, -0.99900135]	0.392915137500505	0.0001476690762578503	10000 (wartość graniczna)
1e-2	[0.02074223, -1.00941198]	0.39277388846064754	6.420036400389684e-06	101

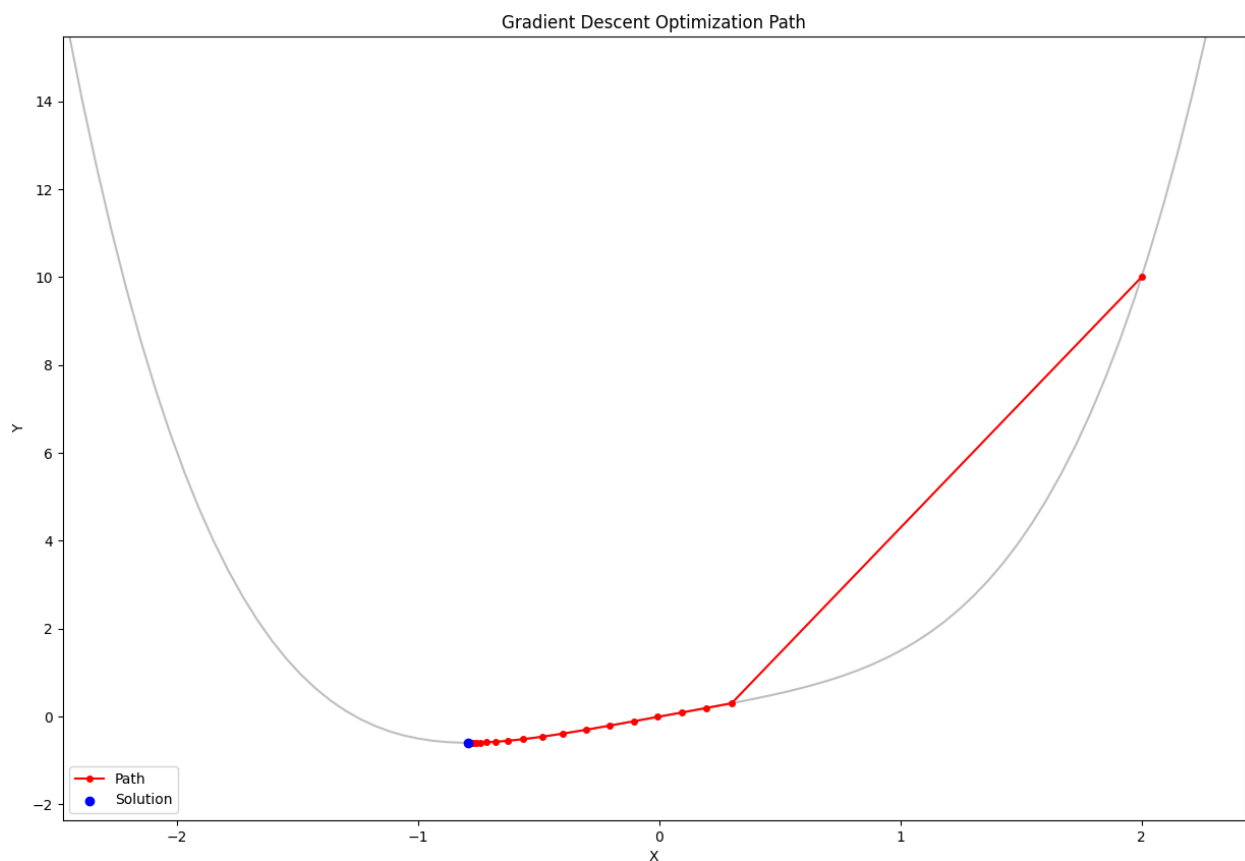
1e-3	[0.02012069, -1.00447013]	0.39280716173917324	3.9693314926092516e-05	1005
------	-------------------------------	---------------------	------------------------	------

Liczba iteracji rośnie wraz ze spadkiem wartości współczynnika uczenia. Algorytm działa poprawnie dla wartości współczynnika uczenia pomiędzy $1e-1$ a $1e-3$. Przy wykorzystaniu mniejszych wartości, różnica między kolejnymi x , jest mniejsza niż wartość domyślnego hiperparametru epsilon.

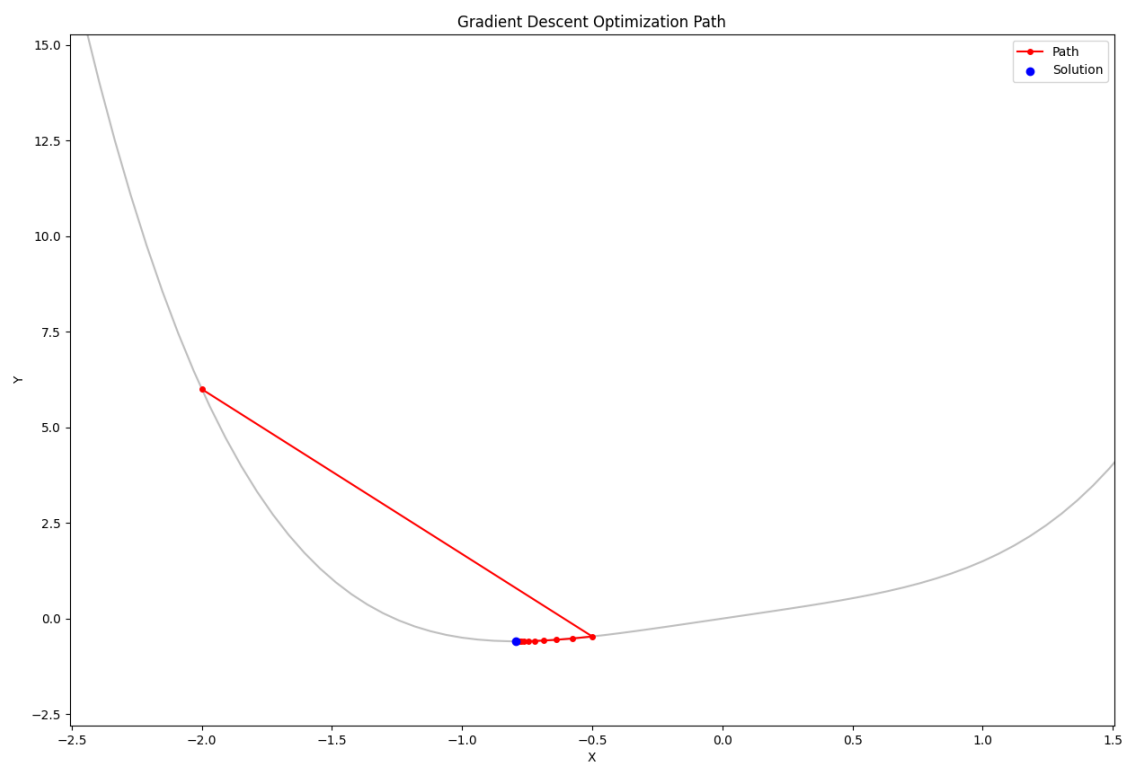
Na poniższych wykresach widać algorytm gradientu prostego dla współczynnika uczenia równego $1e-1$. Widać na nich, że z racji na wykonywaną długość kroku funkcja czasami przybliża się i oddala od minimum, aby w końcu je osiągnąć.

W przypadku wykresu dla funkcji $f(x)$ i początkowym $x = -2.0$ widzimy, że kolejny krok jest po drugiej stronie minimum i po tej stronie następują kolejne punkty w drodze do osiągnięcia minimum. Inaczej wyglądało to na wykresie dla wartości startowych hiperparametrów.

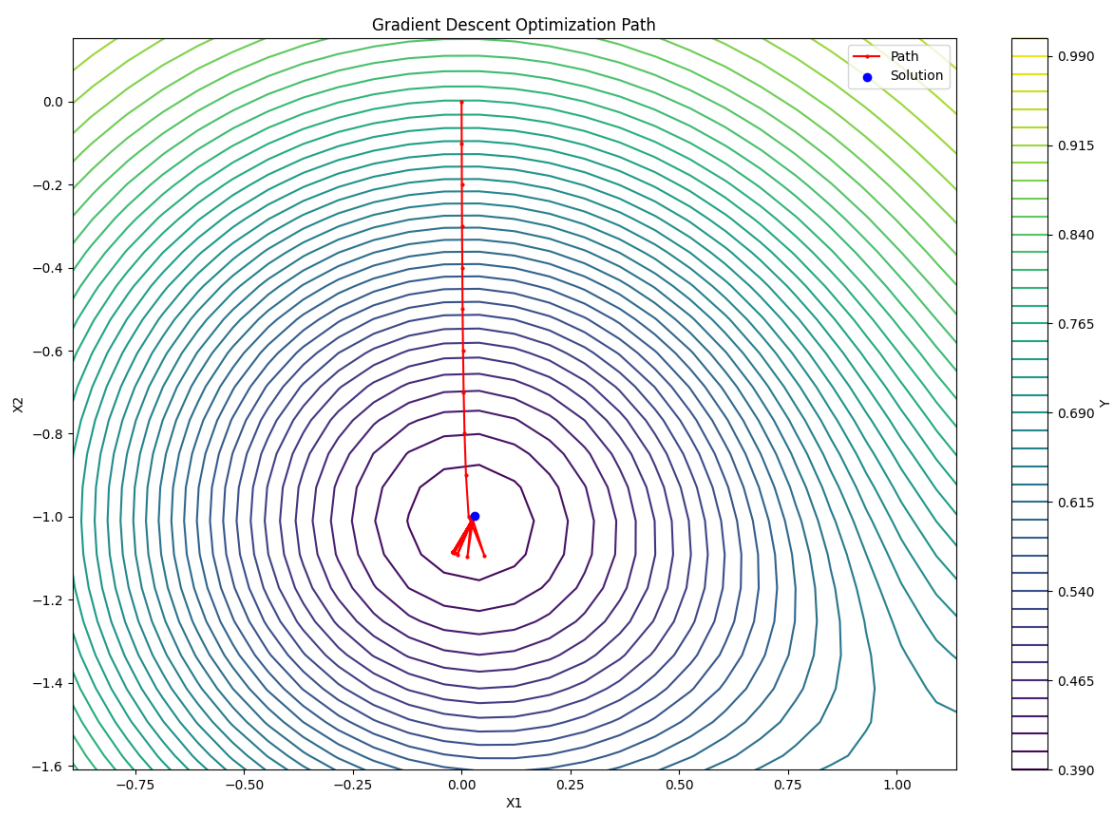
Wykres dla funkcji f , $x_0 = 2.0$



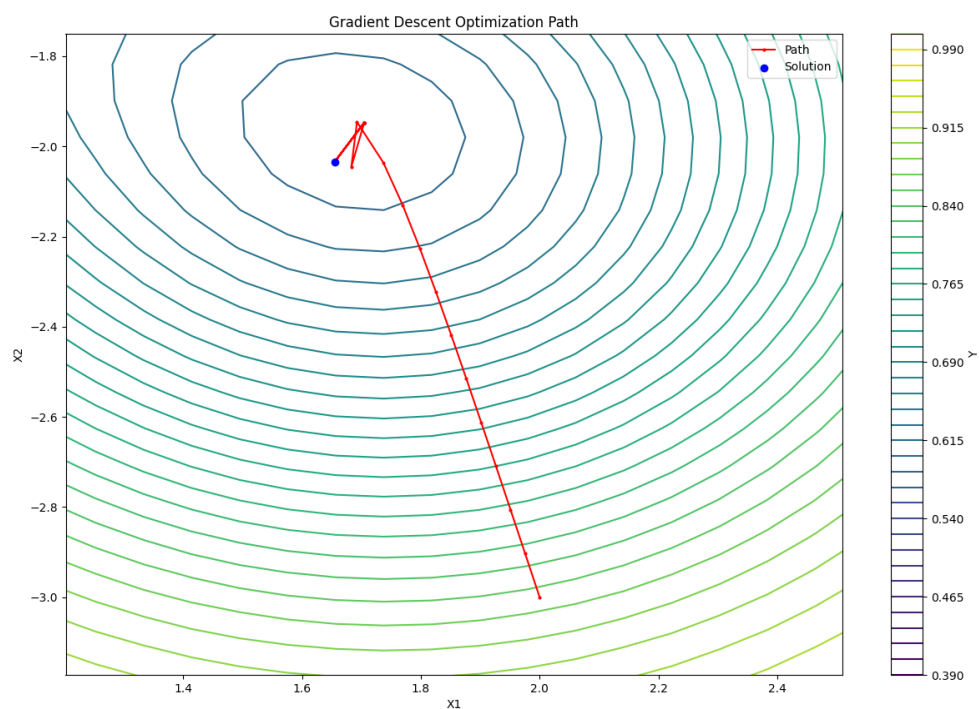
Wykres funkcji f dla $x_0 = -2.0$



Wykres dla funkcji g, $x_0 = [0.0, 0.0]$



Wykres dla funkcji g dla $x_0 = [2.0, -3.0]$



Normalize Threshold (Limit normalizacji)

Prezentujemy wyniki dla $g(x)$. Funkcję jednowymiarowe nie korzystają z normalizacji.

$$g(x) = 1 - 0.6 \exp\{-x_1^2 - (x_2 + 1)^2\} - 0.4 \exp\{-(x_1 - 1.75)^2 - (x_2 + 2)^2\}$$

Normalize Threshold	Wartość x_{\min} dla $x_0 = [2.0, -3.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1	[1.70311998, -1.97926077]	0.5884035158990457	0.19563604747479857	693
1e-1	[1.70310474, -1.97920792]	0.588402996064161	0.19563552763991388	423
1e-2	[1.70069158, -1.97491343]	0.5883623923231265	0.19559492389887934	107
None	[1.70311998, -1.97926077]	0.5884035158990457	0.19563604747479857	693

Normalize Threshold	Wartość x_{\min} dla $x_0 = [0.0, 0.0]$	Znalezione minimum	Różnica względem minimum	Iteracje
1	[0.02009112 -1.00420155]	0.3928097954745812	4.232705033407891e-05	459
1e-1	[0.02007786, -1.00419836]	0.3928098486163308	4.2380192083635304e-05	287
1e-2	[0.02074223, -1.00941198]	0.39277388846064754	6.420036400389684e-06	101
None	[0.02009112, -1.00420155]	0.3928097954745812	4.232705033407891e-05	459

Normalizacja pozwala nam znacząco zmniejszyć liczbę iteracji potrzebnych do uzyskania minimum funkcji. Straty dokładności funkcji nie są znaczące. Widać, że jest to rozwiązanie opłacalne rozważając wynik i liczbę iteracji.

Wnioski

Algorytm gradientu słabo radzi sobie z funkcjami płaskimi (funkcja g).

Najlepsze parametry dla algorytmu zostały przedstawione przeze mnie na początku. Z moich eksperymentów widać, jak duży wpływ w niektórych przypadkach mogą mieć wartości hiperparametrów. Warto dobrać ich wartości pod swój problem, swoje punkty startowe oraz swoją postać funkcji.

Algorytm gradientu prostego nie daje również dokładnej wartości minimum. W tabelach przedstawiłam, jakie rzędu są to odchylenia. Mimo to, nie są one znaczące, a wręcz bardzo małe.