Numerical Optimization - Assignment #1

Viktor Loreth- viktor.loreth@pm.me

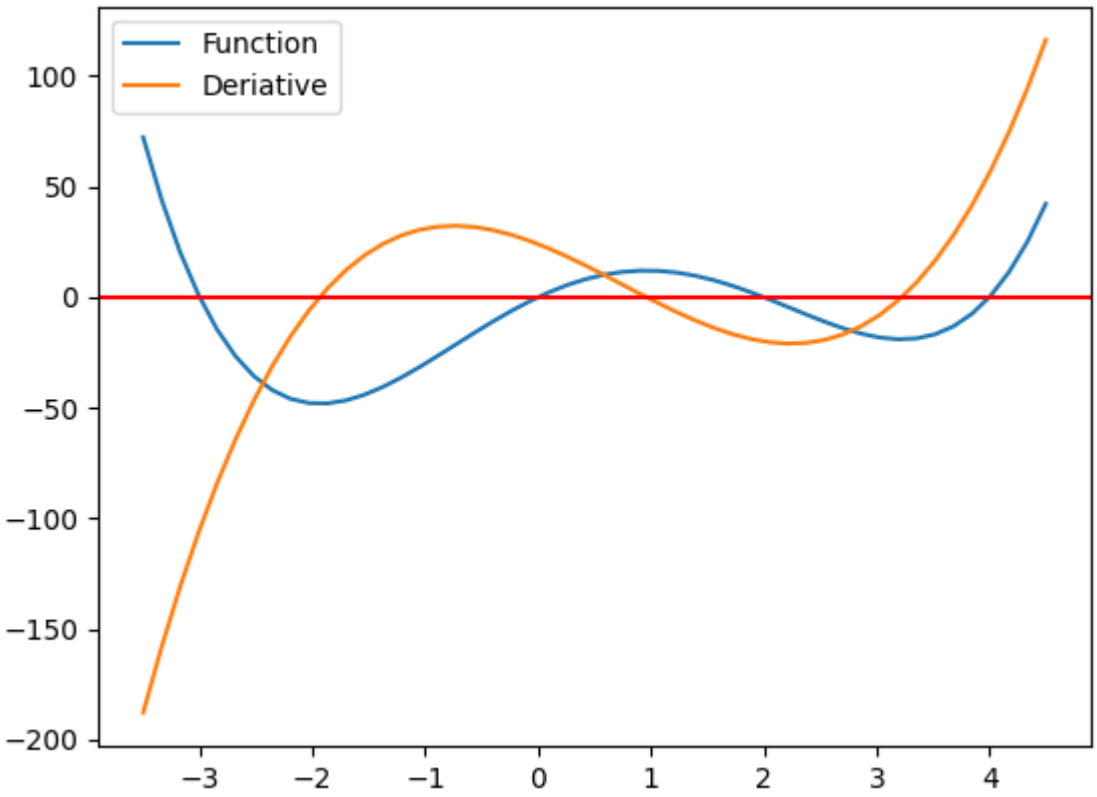
May 25 2022

Introduction:

Epsilon used for all functions is $10^{-}5$. I solved 39 problems + 4 problems from the rosenbrock function and the second function.

1 $f(x) = x^4 - 3x^3 - 10x^2 + 24x; x0 = 2; n = 4$

Known solutions: x1 = -1,935362601; x2 = 3,223663518



This function is very basic to observe for our different functions. The optimization methods only using the gradient will find the local minimum, the ones using the hessian will find the global minimum.

1.1 Steepest Descent

Time taken: 0.010087251663208008 seconds
Number of Iterates: 36
Found solution: 3.2236635185507594
Iterates: [2.0, 2.8, 2.95, 3.06, 3.13, 3.17, 3.19, 3.21, 3.22, 3.22]
found solution - real solution = 5.507594380560477e-10
f(x) = -19.05933819584762

1.2 Newton

Time taken: 0.32587623596191406 seconds
Number of Iterates: 78
Found solution: -1.9353626102299228
Iterates: [2.0, -2.18, -2.13, -2.1, -2.06, -2.04, -2.02, -2.0, -1.99, -1.98]
found solution - real solution = -9.229922826037296e-09
f(x) = -48.127806954042704

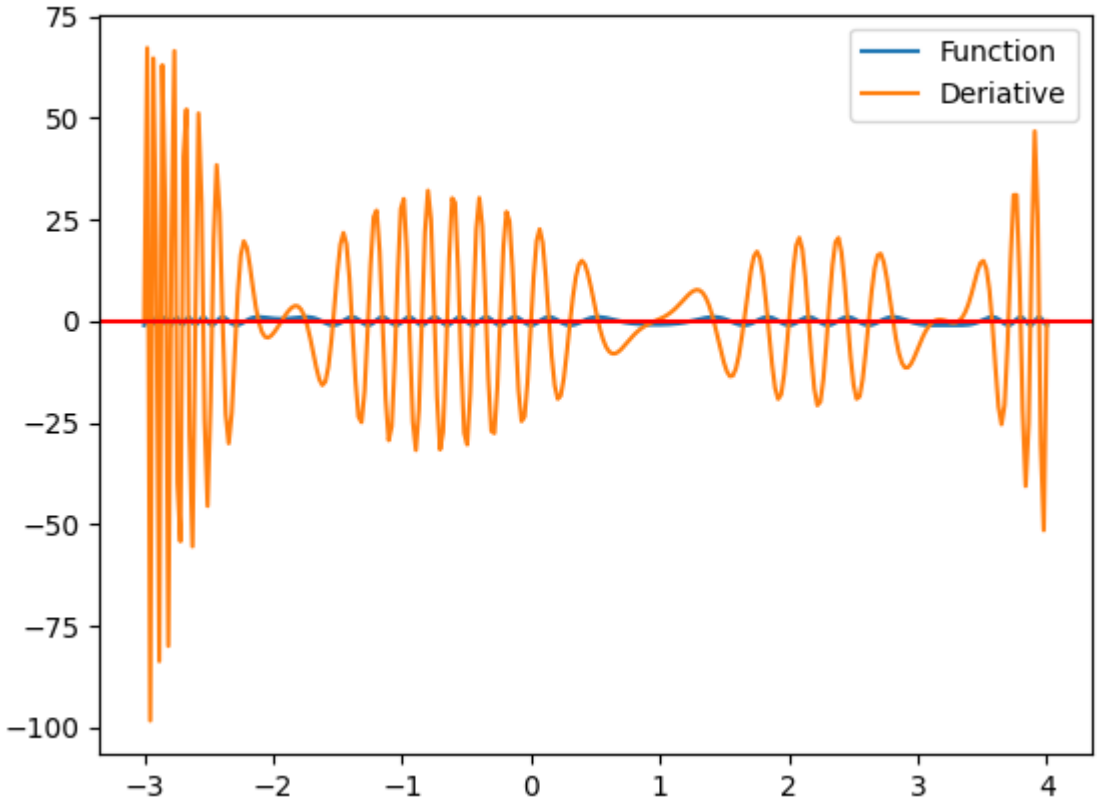
1.3 Conjugate Gradient

Time taken: 0.005005359649658203 seconds
Number of Iterates: 18
Found solution: -1.9353626004931248
Iterates: [2.0, -2.21, -2.16, -2.0, -1.95, -1.94, -1.94, -1.94, -1.94, -1.94]
found solution - real solution = 5.068752084724792e-10
f(x) = -48.127806954042704

1.4 Quasi-Newton

Time taken: 0.13312029838562012 seconds
Number of Iterates: 79
Found solution: 3.223663508436486
Iterates: [2.0, 2.8, 2.91, 2.99, 3.04, 3.08, 3.11, 3.14, 3.16, 3.17]
found solution - real solution = -9.56351398073707e-09
f(x) = -19.059338195847616

2 $f(x) = -cos(x^4 - 3x^3 - 10x^2 + 24x; x0 = 2.2; n = 4)$



Known solutions: 2; x2 = 2.3031656

2.1 Steepest Descent

Time taken: 0.0040018558502197266 seconds
Number of Iterates: 10
Found solution: 2.303165692228743
Iterates: [2.2, 2.31, 2.3, 2.3, 2.3, 2.3, 2.3, 2.3, 2.3, 2.3]
found solution - real solution = 9.222874330916397e-08
f(x) = -1.0

2.2 Steepest Descent

Time taken: 0.7251002788543701 seconds
Number of Iterates: 18
Found solution: 1.999999992612371
Iterates: [2.2, 1.98, 1.99, 2.0, 2.0, 2.0, 2.0, 2.0, 2.0, 2.0]
found solution - real solution = -7.3876289530261374e-09
f(x) = -0.9999999999999891

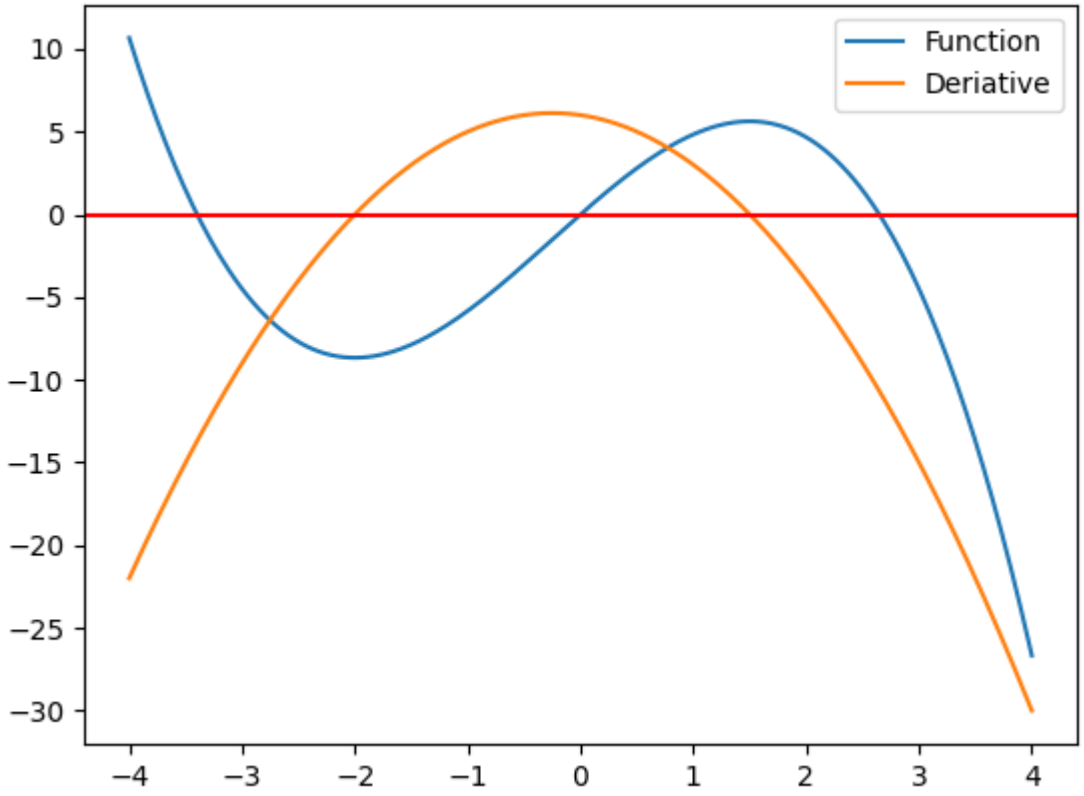
2.3 Conjugate Gradient

Time taken: 0.004038810729980469 seconds
Number of Iterates: 11
Found solution: 1.9999999999588742
Iterates: [2.2, 1.95, 1.98, 2.0, 2.0, 2.0, 2.0, 2.0, 2.0, 2.0]
found solution - real solution = -4.112576945658475e-11
f(x) = -1.0

2.4 Quasi-Newton

Time taken: 0.3731093406677246 seconds
Number of Iterates: 16
Found solution: 2.3031656997379537
Iterates: [2.2, 2.31, 2.3, 2.3, 2.3, 2.3, 2.3, 2.3, 2.3, 2.3]
found solution - real solution = 9.973795389583984e-08
f(x) = -0.9999999999999876

3 $f(x) = x^4 - 2/3 * x^3 - x^2/2 + 2 * x; x0 = -4; n = 4)$



Known solutions: -2:

3.1 Steepest Descent

Time taken: 0.006062030792236328 seconds
Number of Iterates: 40
Found solution: -2.0000000008417382
Iterates: [-4.0, -3.12, -2.7, -2.35, -2.19, -2.1, -2.06, -2.03, -2.02, -2.01]
found solution - real solution = -8.417382346692648e-10
f(x) = -8.666666666666668

3.2 Newton Method

Time taken: 0.12862801551818848 seconds
Number of Iterates: 39
Found solution: -2.000000008145913
Iterates: [-4.0, -3.35, -2.89, -2.56, -2.36, -2.22, -2.13, -2.08, -2.05, -2.03]
found solution - real solution = -8.145912833157354e-09
f(x) = -8.666666666666668

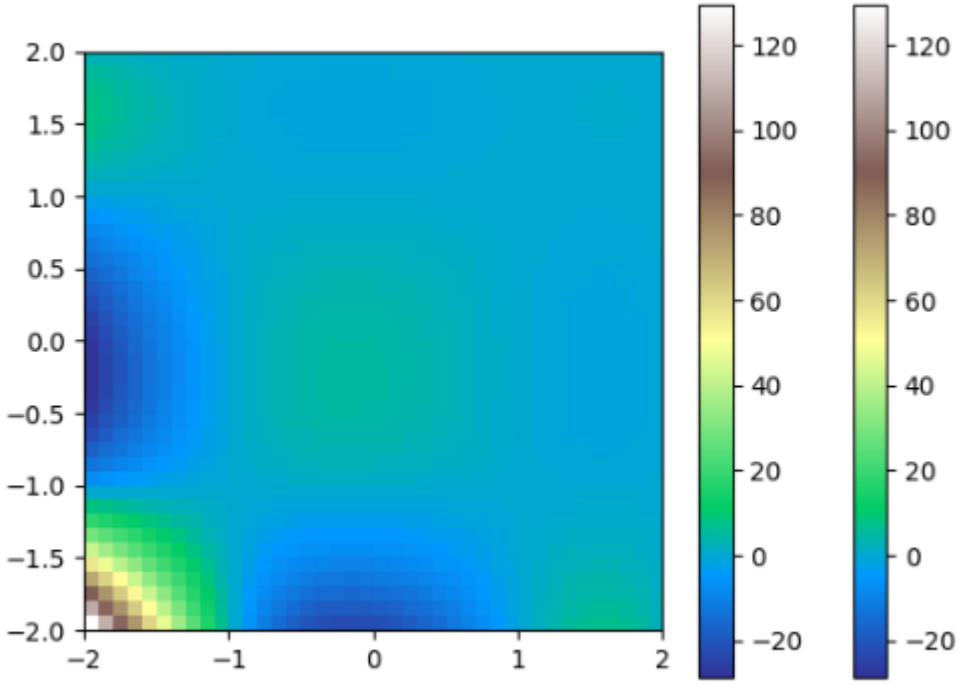
3.3 Conjugate Gradient

Time taken: 0.004003763198852539 seconds
Number of Iterates: 30
Found solution: -2.0000000007739045
Iterates: [-4.0, -3.12, -2.51, -2.25, -2.12, -2.06, -2.03, -2.01, -2.01, -2.0]
found solution - real solution = -7.739044960430874e-10
f(x) = -8.666666666666668

3.4 Quasi-Newton

Time taken: 0.019097089767456055 seconds
Number of Iterates: 39
Found solution: -2.0000000061684666
Iterates: [-4.0, -3.12, -2.73, -2.46, -2.28, -2.17, -2.1, -2.06, -2.04, -2.02]
found solution - real solution = -6.168466626377267e-09
f(x) = -8.666666666666668

4 $f(x) = (1-x1)*(-1-x1)*(2-x1)*(1-x2)*(2-x2)*(-1,1-x2); x0 = [1.2, 1.3]; n = 3*3$



Known solutions: [[1, 1], [-0.21524, 1.546813]]:

4.1 Steepest Descent

Time taken: 0.45076489448547363 seconds
Number of Iterates: 22
Found solution: [-0.21524886 1.54681282]
Iterates: [array([1.2, 1.3], dtype=float32), array([0.91, 1.14]), array([0.08, 1.63]), array([0. , 1.55]), array([-0.13, 1.54]), array([-0.18, 1.55]), array([-0.2 , 1.54]), array([-0.21, 1.55]), array([-0.21, 1.55]), array([-0.21, 1.55])]
found solution - real solution = [-8.86437445e-06 -1.75303386e-07]
f(x) = -1.3856681436300882

4.2 Newton Method

Time taken: 0.801116943359375 seconds
Number of Iterates: 14
Found solution: [1.00000028 1.00000163]
Iterates: [array([1.2, 1.3], dtype=float32), array([1.02, 1.1]), array([1.01, 1.04]), array([1. , 1.02]), array([1. , 1.01]), array([1., 1.]), array([1., 1.]), array([1., 1.]), array([1., 1.]), array([1., 1.])]
found solution - real solution = [2.77556160e-07 1.63290744e-06]
f(x) = 1.9035368820317917e-12

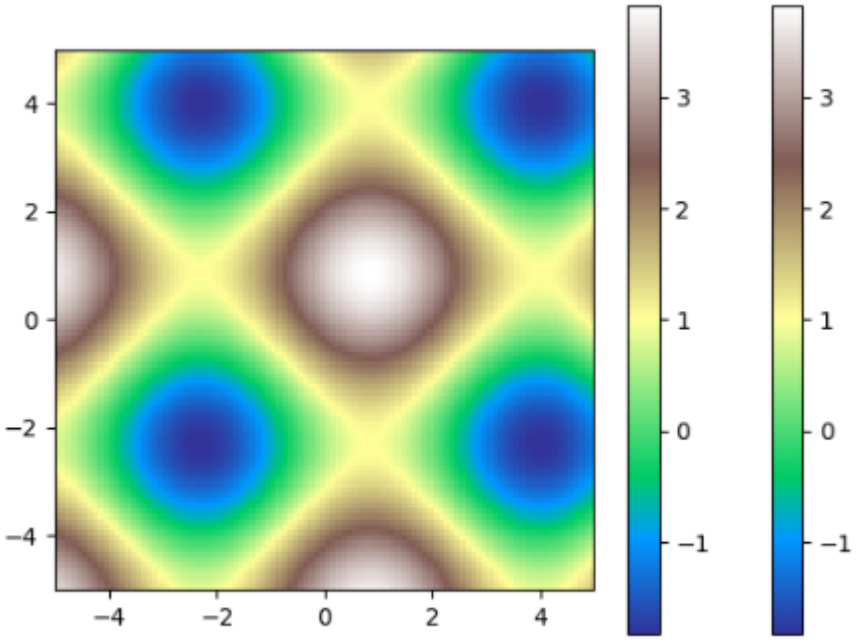
4.3 Conjugate Gradient

Time taken: 0.41343212127685547 seconds
Number of Iterates: 398
Found solution: [-0.21524794 1.54681343]
Iterates: [array([1.2, 1.3], dtype=float32), array([0.91, 1.14]), array([-0.22, 1.23]), array([-0.37, 1.27]), array([-0.45, 1.3]), array([-0.5 , 1.33]), array([-0.54, 1.36]), array([-0.56, 1.4]), array([-0.59, 1.44]), array([-0.6 , 1.47])]
found solution - real solution = [-7.93953119e-06 4.29741695e-07]
f(x) = -1.385668143620145

4.4 Quasi-Newton

Time taken: 1.0235044956207275 seconds
Number of Iterates: 18
Found solution: [-0.21525096 1.54681215]
Iterates: [array([1.2, 1.3], dtype=float32), array([0.91, 1.14]), array([-0.25, 1.29]), array([-0.35, 1.32]), array([-0.29, 1.47]), array([-0.24, 1.52]), array([-0.23, 1.53]), array([-0.22, 1.54]), array([-0.22, 1.54]), array([-0.22, 1.55])]
found solution - real solution = [-1.09609160e-05 -8.45230645e-07]
f(x) = -1.385668143632735

5 $f(x) = sin(x1) + cos(x1) + sin(x2) + cos(x2); x0 = [0, 0]$



5.1 Steepest Descent

Time taken: 0.8553903102874756 seconds
Number of Iterates: 14
Found solution: [-2.3561899 -2.3561899]
Iterates: [array([0., 0.], dtype=float32), array([-2.07, -2.07]), array([-2.24, -2.24]), array([-2.31, -2.31]), array([-2.34, -2.34]), array([-2.35, -2.35]), array([-2.35, -2.35]), array([-2.36, -2.36]), array([-2.36, -2.36]), array([-2.36, -2.36])]
found solution - real solution = [3.08953441e-09 3.08953441e-09]
f(x) = -1.8284271247163524

5.2 Newton Method

Time taken: 1.2379250526428223 seconds
Number of Iterates: 14

Found solution: [3.92699556 3.92699556]
Iterates: [array([0., 0.], dtype=float32), array([4.22, 4.22]), array([4.04, 4.04]), array([3.97, 3.97]), array([3.95, 3.95]), array([3.93, 3.93]), array([3.93, 3.93]), array([3.93, 3.93]), array([3.93, 3.93]), array([3.93, 3.93])]
found solution - real solution = [-1.91256344e-09 -1.91256344e-09]
f(x) = -1.8284271247144015

5.3 Conjugate Gradient

Time taken: 0.6085433959960938 seconds
Number of Iterates: 14
Found solution: [-2.35618996 -2.35618996]
Iterates: [array([0., 0.], dtype=float32), array([-2.07, -2.07]), array([-2.24, -2.24]), array([-2.31, -2.31]), array([-2.34, -2.34]), array([-2.35, -2.35]), array([-2.35, -2.35]), array([-2.36, -2.36]), array([-2.36, -2.36]), array([-2.36, -2.36])]
found solution - real solution = [-5.78389892e-08 -5.78389892e-08]
f(x) = -1.8284271247171393

5.4 Quasi-Newton

Time taken: 1.0614640712738037 seconds
Number of Iterates: 14
Found solution: [-2.35618984 -2.35618984]
Iterates: [array([0., 0.], dtype=float32), array([-2.07, -2.07]), array([-2.24, -2.24]), array([-2.31, -2.31]), array([-2.34, -2.34]), array([-2.35, -2.35]), array([-2.35, -2.35]), array([-2.36, -2.36]), array([-2.36, -2.36]), array([-2.36, -2.36])]
found solution - real solution = [6.33881858e-08 6.33881858e-08]
f(x) = -1.8284271247155641

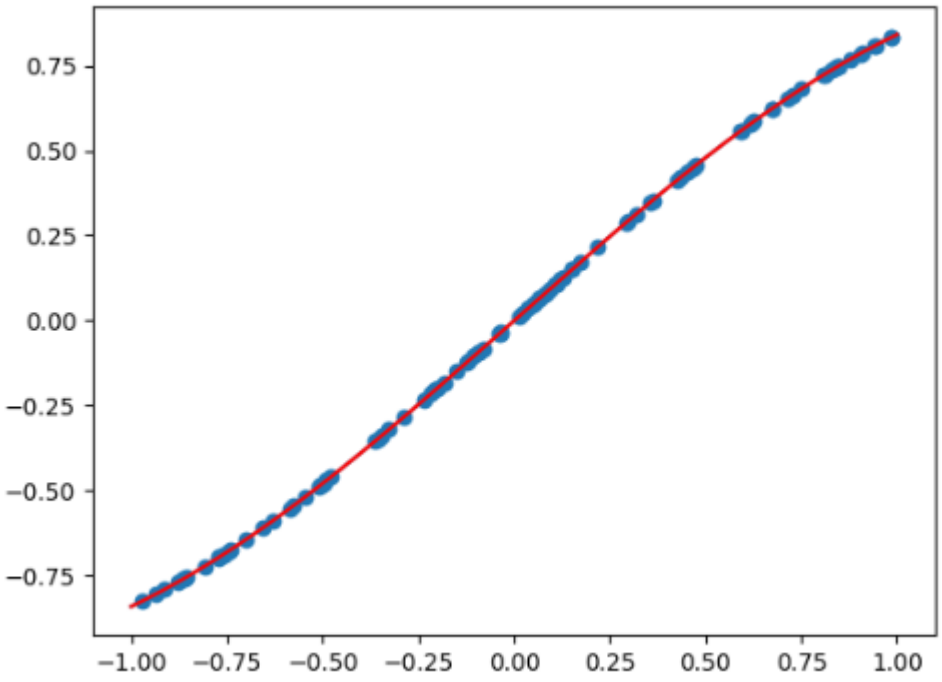
6 Least-Squares

Taylor expansion of sinus at x=0:

Y = X - X3/ 3! + X5/ 5! - X7/ 7! + X9/ 9! - X11/ 11! The Taylor expansion is not visible at any plots because they overlap. I will send the Taylor plots if required.

6.1 Run 1:

q = 1
Degree = 5
Data samples = 100
function g(x) = sin(x)



Steepest Descent:
Time taken: 1.0900278091430664 seconds
$[-0.00390368, 0.9153897, 0.00667812, 0.06866298, -0.00329793, -0.12835446]$

Newton Method

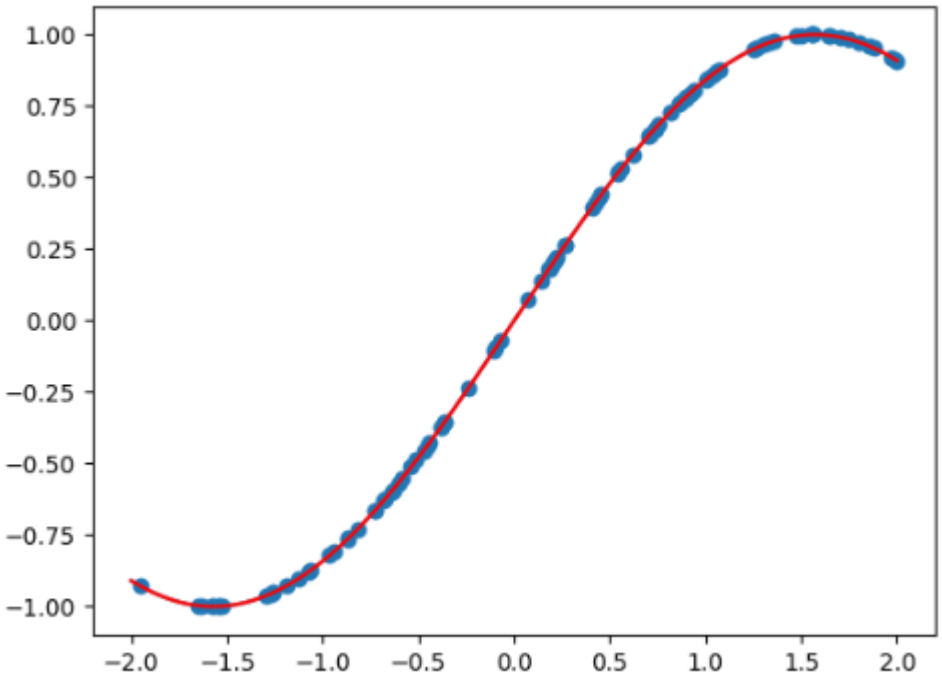
Time taken: 0.06806182861328125 seconds
$[-6.69211031e-08, 9.99987345e-01, 5.29360178e-07, -1.66539327e-01, -1.80933078e-06, 8.03249621e-03]$

Conjugate Gradient
Time taken: 0.20720601081848145 seconds
$[2.05163616e-05, 9.99985447e-01, 5.65408148e-06, -1.66537088e-01, 4.74219490e-06, 8.02917519e-03]$

Quasi-Newton
Time taken: 0.06836748123168945 seconds
$[-6.69211031e-08, 9.99987345e-01, 5.29360178e-07, -1.66539327e-01, -1.80933078e-06, 8.03249621e-03]$

6.2 Run 2:

q = 2
Degree = 8
Data samples = 100
function g(x) = sin(x)



Steepest Descent:
Time taken: 7.36026484823618098 seconds
-3.4418593939e-05, 9.99983741e-01, 6.3596476946e-06, -1.66604734e-01, -4.4392890032e-06, 3.8853423702e-03, 2.83720798237-07, -4.329581342e-04

Newton Method
Time taken: 0.10649871826171875 seconds
-1.21809212e-07, 9.99984957e-01, 1.23731611e-06, -1.66609792e-01, -1.45179621e-06, 8.27525551e-03, 3.10622229e-07, -1.76654196e-04

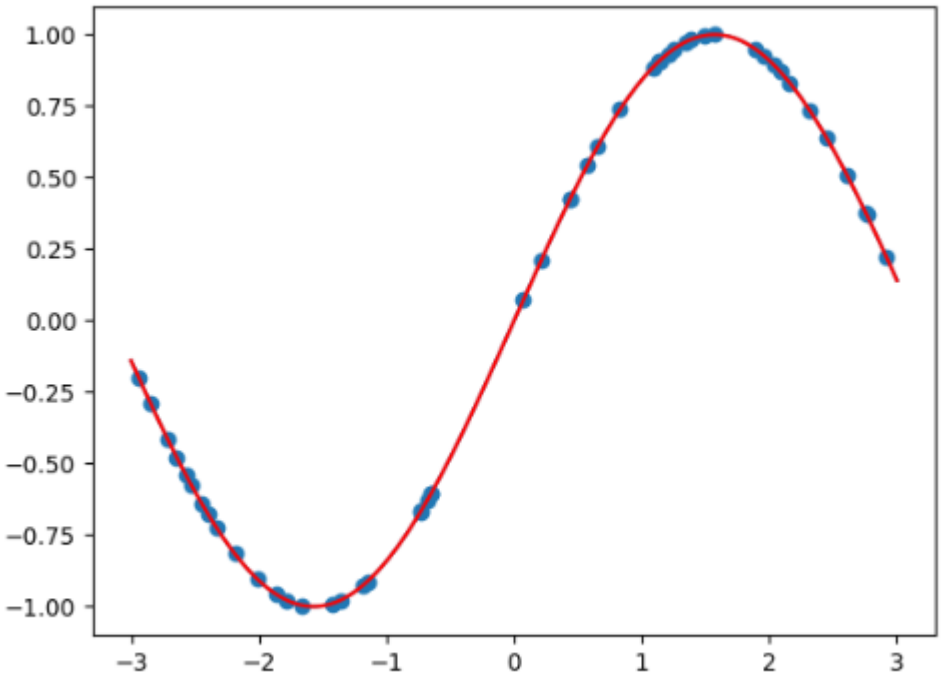
Conjugate Gradient
Time taken: 0.6406822204589844 seconds
-5.94995501e-07, 9.99982885e-01, 2.78638801e-06, -1.66605239e-01, -2.81794889e-06, 8.27217219e-03, 5.66448406e-07, -1.76119033e-04

Quasi-Newton
Time taken: 0.10578131675720215 seconds

-6.33098508e-07, 9.99982996e-01, 2.87671316e-06, -1.66605068e-01,-2.67221962e-06, 8.27239419e-03, 5.32034769e-07, -1.76219647e-04

6.3 Run 3:

q = 3
Degree = 8
Data samples = 100
function g(x) = sin(x)



Steepest Descent:
Time taken: 21.274384823618098 seconds
-3.3324262807-05, 9.99347655e-01, 6.8905093516-06, -1.66604734e-01, -4.8905093516-06, 3.8905093516-03, 2.8905093516-07, -4.8905093516-04

Newton Method
Time taken: 0.05431962013244629 seconds
2.91222840e-05, 9.99603417e-01, -5.55612184e-05, -1.66030610e-01, 1.82148071e-05, 8.05361969e-

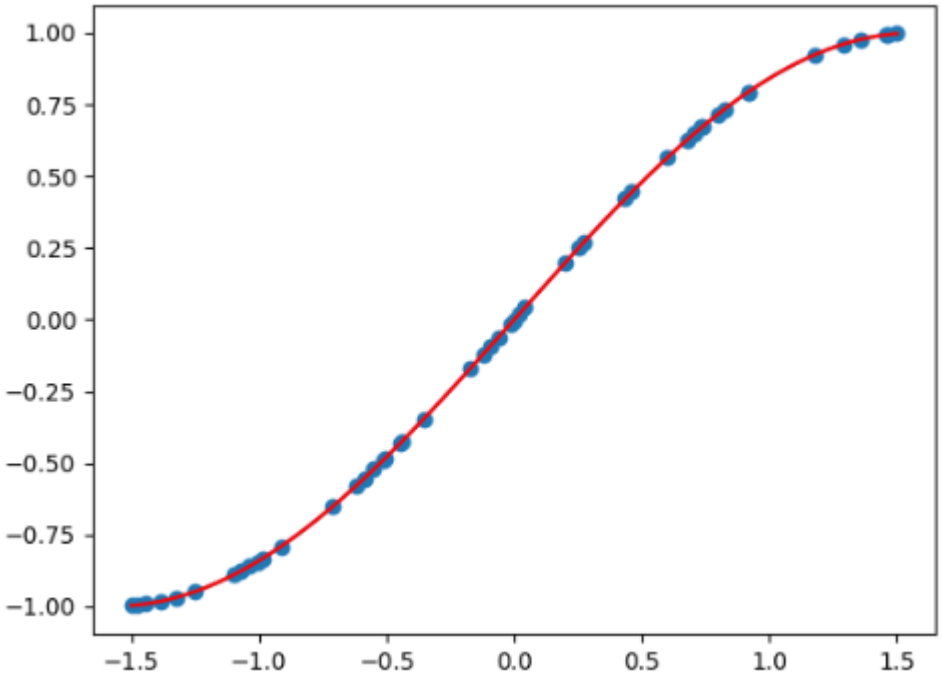
03, -1.52180735e-06, -1.52022857e-04

Conjugate Gradient
Time taken: 0.3100926876068115 seconds
3.08301564e-03, 9.99542733e-01, -4.02257055e-03, -1.65991202e-01, 1.08591640e-03, 8.05989769e-03, -7.92067181e-05, -1.53362859e-04

Quasi-Newton
Time taken: 0.05420684814453125 seconds
2.91222840e-05, 9.99603417e-01, -5.55612184e-05, -1.66030610e-01, 1.82148071e-05, 8.05361969e-03, -1.52180735e-06, -1.52022857e-04

6.4 Run 4:

q = 1.5 Degree = 5 Data samples = 50 function g(x) = sin(x)



Steepest Descent:
Time taken: 4.6615424156188965 seconds

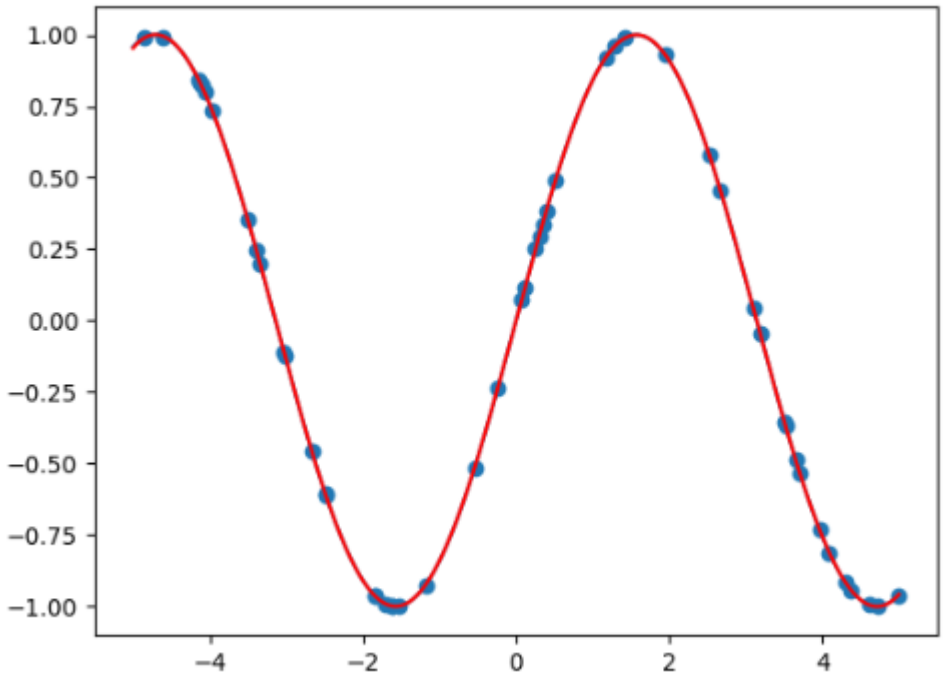
-0.00456991, 0.89584468, -0.00931609, 0.03921084, 0.00536012, -0.07497266

Newton Method
Time taken: 0.03515625 seconds
-1.43706894e-06, 9.99864318e-01, -5.11508376e-06, -1.66052981e-01, 5.13184111e-06, 7.68283731e-03

Conjugate Gradient
Time taken: 0.1232903003692627 seconds
-1.44026985e-06, 9.99864318e-01, -5.12001984e-06, -1.66052981e-01, 5.12373555e-06, 7.68283770e-03

Quasi-Newton
Time taken: 0.0378575325012207 seconds
-1.43706894e-06, 9.99864318e-01, -5.11508376e-06, -1.66052981e-01, 5.13184111e-06, 7.68283731e-03

6.5 Run 5:



q = 5
Degree = 10
Data samples = 50
function g(x) = sin(x)

Newton Method
Time taken: 0.09497189521789551 seconds
2.08366947e-04, 9.98525989e-01, -1.23074635e-04, -1.65488952e-01, 5.26111121e-05, 8.05772498e-03, -8.24593121e-06, -1.71012708e-04, 4.97244507e-07, 1.49367652e-06, -9.98667161e-09
3.5874194354204536e-06

Conjugate Gradient
Time taken: 8.144790410995483 seconds
2.47439941e-03, 9.97264042e-01, -3.24156452e-03, -1.65034235e-01, 9.43484141e-04, 7.98013925e-03, -1.00952482e-04, -1.63668884e-04, 4.40443081e-06, 1.28450768e-06, -6.61901634e-08

Quasi-Newton
Time taken: 0.09452700614929199 seconds

2.08366947e-04, 9.98525989e-01, -1.23074635e-04, -1.65488952e-01, 5.26111121e-05, 8.05772498e-03, -8.24593121e-06, -1.71012708e-04, 4.97244507e-07, 1.49367652e-06, -9.98667161e-09

7 Rosenbrock function

7.1 Steepest Descent:

Time taken: 1.8149046897888184 seconds
Number of Iterates: 1138
Found solution: [0.99998723 0.99997441]
Iterates: [array([-1.2, 1.], dtype=float32), array([-0.98, 1.09]), array([-1.03, 1.06]), array([-0.95, 0.89]), array([-0.94, 0.89]), array([-0.94, 0.89]), array([-0.94, 0.88]), array([-0.93, 0.88]), array([-0.93, 0.87]), array([-0.93, 0.87])]
found solution - real solution = [-1.27682324e-05 -2.55861232e-05]
f(x) = 1.632759756615874e-10

7.2 Newton Method

Time taken: 1.6465327739715576 seconds
Number of Iterates: 107
Found solution: [0.9999991 0.99999818]
Iterates: [array([-1.2, 1.], dtype=float32), array([-1.2 , 1.08]), array([-1.19, 1.13]), array([-1.18, 1.17]), array([-1.17, 1.19]), array([-1.16, 1.2]), array([-1.14, 1.2]), array([-1.13, 1.18]), array([-1.1 , 1.14]), array([-1.07, 1.09])]
found solution - real solution = [-8.97451744e-07 -1.81969718e-06]
f(x) = 8.66896357185098e-13

7.3 Quasi-Newton

Time taken: 1.4309754371643066 seconds
Number of Iterates: 110
Found solution: [0.999999 0.99999796]
Iterates: [array([-1.2, 1.], dtype=float32), array([-0.98, 1.09]), array([-0.85, 0.78]), array([-0.81, 0.69]), array([-0.78, 0.63]), array([-0.76, 0.59]), array([-0.74, 0.55]), array([-0.73, 0.52]), array([-0.72, 0.5]), array([-0.7 , 0.47])]
found solution - real solution = [-1.00396168e-06 -2.03595426e-06]
f(x) = 1.0865178322300813e-12

7.4 Conjugate Gradients

For Conjugate Gradients it stops after 20 iterates at [-0.32457923 0.04166611] because we can't find a suitable step-size. Due to the way conjugate gradients always finds direction

normal on each other, this might lead into a direction which we can't minimize at.

8 Second function

8.1 Steepest Descent

: Time taken: 1.1313400268554688 seconds
Number of Iterates: 156
Found solution: [-1.91602391e-10 9.99998769e-01]
Iterates: [array([-1.2, 1.], dtype=float32), array([-1.02, 0.78]), array([-0.88, 0.61]), array([-0.78, 0.47]), array([-0.71, 0.37]), array([-0.66, 0.28]), array([-0.62, 0.22]), array([-0.59, 0.17]), array([-0.57, 0.14]), array([-0.55, 0.11])]
found solution - real solution = [-1.0000000e+00 -1.2306783e-06]
f(x) = 6.058753393096457e-12

8.2 Conjugate Gradient

:

For Conjugate Gradient we end up bouncing between 2 vectors the whole time. Maybe one could decrease the step size to find a better solution. [0.38690017 0.14108287] [-0.01723019 0.6641161] [0.38690017 0.14108287] [-0.01757765 0.66456579] [0.38690017 0.14108287]

8.3 Quasi-Newton

Quasi Newton moves until the vectors below and afterwards starts bouncing as well. Wolfe conditions need to be adjusted! [-1.83223892 0.01926842] [-1.8336777 0.01741526] [-1.83223892 0.01926842] [-1.83367809 0.01741476] [-1.83223892 0.01926842]

8.4 Newton Method

Same as Quasi Newton. Adjust step length and search conditions! [-0.52841333 0.15678914] [-0.50548672 0.18927382] [-0.5284242 0.15677374]