Jacob Leonard

Math 467

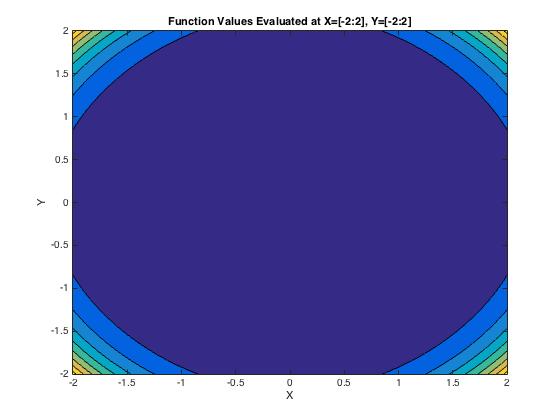
Project #1

Imaginary Analysis

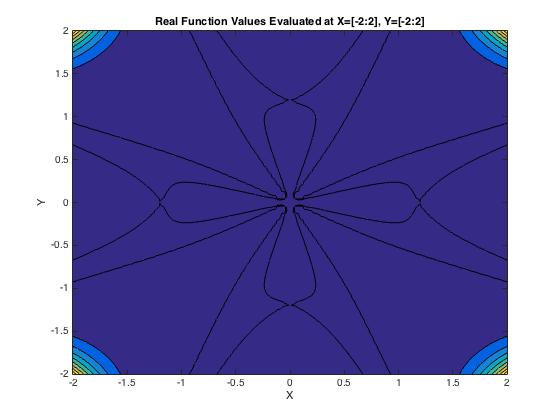
1. Analytically Compute the Gradient and Hessian of the Function:

Evaluating the function at all the points on the intervals:

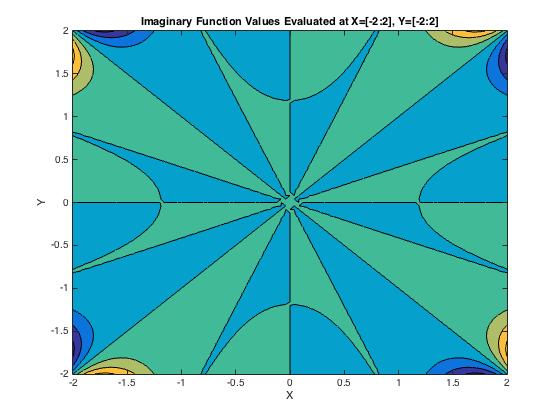
1. Graph of the function , z = (x+iy)



1. Graph of the real values



1. Graph of the imaginary values



1. Gradient:
2. Hessian:

1. Roots:
2. Gradient Along Root Values;
   1. = [0,0]

x = [-2,2], y = ix+1

x = [-2,2], y = ix-1

* 1. = [0,0]

x = [-2,2], y = ix+i

* 1. = [0,0]

x = [-2,2], y = ix-i

1. Gradient Near Root Values: D = +.04
   1. = [-1.4698, - 1.4698i]

x = [-2,2}, y = ix+1.04

* Increasing the y value along this root by .04 produces a negative real X gradient and a negative imaginary Y gradient
  1. = [1.1107, 1.1107i]

x = [-2,2}, y = ix-.96

* Increasing the y value along this root by .04 produces a real positive X gradient, and positive imaginary Y gradient
  1. = [-0.1789 - 1.2698i, 1.2698 - 0.1789i]

x = [-2,2}, y = ix+i+.04

* Increasing the y value along this root by .04 produces a negative X gradient, positive real valued Y gradient and negative imaginary Y gradient
  1. = [-0.1789 + 1.2698i, - 1.2698 - 0.1789i]

x = [-2,2}, y = ix-i+.04

* Increasing the y value along this root by .04 produces a negative Y gradient, negative real valued X gradient and positive imaginary X gradient

1. Gradient Near Root Values: D = -.04
   1. = [1.1107, 1.1107i]

x = [-2,2}, y = ix+.96

* Decreasing the y value along this root by .04 produces a positive gradient that is isometric to its parallel y= , meaning the function is symmetric about y = ix
  1. = [-1.4698 , - 1.4698i]

x = [-2,2}, y = ix-1.04

* Decreasing the y value along this root by .04 produces a positive gradient that is isometric to its parallel y= , meaning the function is symmetric about y = ix
  1. = [-0.1789 + 1.2698i , -1.2698 - 0.1789i]

x = [-2,2}, y = ix+i-.04

* Decreasing the y value along this root by .04 produces a gradient that is isometric to its parallel y=
  1. = [-0.1789 - 1.2698i , +1.2698 - 0.1789i]

x = [-2,2}, y = ix-i-.04

* Decreasing the y value along this root by .04 produces a gradient that is isometric to its parallel y=

1. Hessian Along Root Values:

x = [-2,2], y = ix+1

x = [-2,2], y = ix-1

* The hessian of the function along the parallel lines y = ix+1, y = ix+1 produces isometric matrices, meaning the function denatures at the same quadratic rate

x = [-2,2], y = ix+i

x = [-2,2], y = ix-i

* The hessian of the function along the parallel roots y=ix+1, y=ix-1 produces isometric matrices, meaning the function denatures at the same quadratic rate

1. Hessian Near Root Values: D = .04
   1. = [-44.8995, -44.8995i, -44.8995i , 44.8995]

x = [-2,2}, y = ix+1.04

* Increasing the y value along this root by .04 produces a highly negative hessian, except a positive quadratic Y term
  1. = [-21.7160, -21.7160i, -21.7160i ,21.7160 ]

x = [-2,2}, y = ix-.96

* Increasing the y value along this root by .04 produces a highly negative hessian, except a positive quadratic Y term
  1. = [30.6966 -11.4484i, 11.4484 +30.6966i,  
      11.4484 +30.6966i, -30.6966 +11.4484i]

x = [-2,2}, y = ix+i+.04

* Increasing the y value along this root by .04 produces a symmetric hessian with inverse quadratic terms
  1. = [30.6966 +11.4484i, -11.4484 +30.6966i,  
      -11.4484 +30.6966i, -30.6966 -11.4484i]

x = [-2,2}, y = ix-i+.04

* Increasing the y value along this root by .04 produces a symmetric hessian with inverse quadratic terms, large negative Y and large positive X

1. Hessian Near Root Values: D = -.04
   1. = [-21.7160 , -21.7160i, -21.7160i, 21.7160 ]

x = [-2,2}, y = ix+.96

* Decreasing the y value along this root by .04 produces a hessian equal to the line y = ix-.96, meaning the quadratic decomposition is symmetric about y = ix
  1. = [-44.8995, -44.8995i, -44.8995i , 44.8995]

x = [-2,2}, y = ix-1.04

* Decreasing the y value along this root by .04 produces a hessian equal to the line y = ix+1.04, meaning the quadratic decomposition is symmetric about y = ix
  1. = [30.6966 +11.4484i, -11.4484 +30.6966i,  
      -11.4484 +30.6966i, -30.6966 -11.4484i]

x = [-2,2}, y = ix+i-.04

* Decreasing the y value along this root by .04 produces a hessian equal to the line y = ix-i+.04, meaning the quadratic decomposition is symmetric about y = ix
  1. = [30.6966 -11.4484i, 11.4484 +30.6966i,  
      11.4484 +30.6966i, -30.6966 +11.4484i]

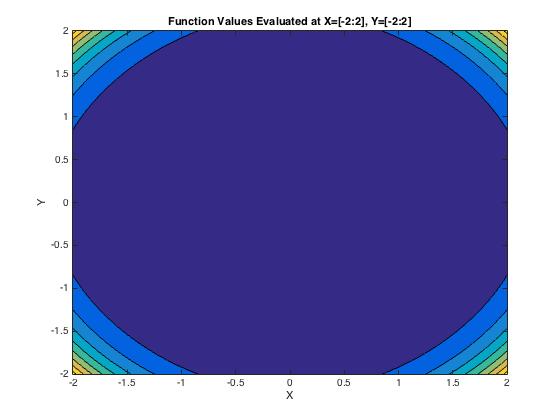
x = [-2,2}, y = ix-i-.04

* Decreasing the y value along this root by .04 produces a hessian equal to the line y = ix+i+.04, meaning the quadratic decomposition is symmetric about y = ix

Real Analysis

The above analysis looked at the function over the complex plane, but for the use of algorithms within Matlab, the following analysis will look at the function:

Evaluating the function at all points on the intervals:



1. Gradient:
2. Hessian:
3. Roots: z = [1,0], [-1,0], [0,1], [0,-1]
4. Gradient Along Root Values:
   1. = [0,0]

x = 1, y = 0

x = -1, y = 0

* 1. = [0,0]

x = 0, y = 1

* 1. = [0,0]

x = 0, y = -1

1. Gradient Near Root Values: D = .04
   1. = [0.0769, 1.2810]
   2. = [-0.0769, 1.2810]

* The gradient at these two points show the function is symmetric about y and opposite about x
  1. = [0, 1.5285]
  2. = [0, 1.0663]
* The gradient at these points shows that the function has an x gradient of 0 about x=0 and positive in the y direction

1. Gradient Near Root Values: D= -.04
   1. = [0.0769, -1.2810]

* The gradient at this point has the same X gradient and negative y gradient, meaning the function is negative symmetric about y=0
  1. = [-0.0769, -1.2810]
* The gradient at this point has the same X gradient and negative y gradient, meaning the function is negative symmetric about y=0
  1. = [0, -1.0663]
* The gradient at this point is the negative gradient at x = 0, y = -.96, meaning the function is negative symmetric about x = 0
  1. = [0, -1.5285]
* The gradient at this point is the negative gradient at x = 0, y = -.96, meaning the function is negative symmetric about x = 0

1. Hessian At Root Values:

* All of the hessians are equal at the roots of the function, showing that the minimum of the function is dictated by the quadratic terms, and denatures in the same way

1. Hessian Near Root Values: D = .04

* These two hessians have the same quadratic growth, but denature linearly inversely
* These two hessians both have zero linear terms and positive quadratic terms, and so both increase about x = 0, y 1

1. Hessian Near Root Values: D = -.04

* The hessian at this point is equal to the hessian at x = -1, y=.04, meaning the function is quadratically symmetric about y=-.04x
* The hessian at this point is equal to the hessian at x = 1, y=.04, meaning the function is quadratically symmetric about y=.04x
* The hessian at this point is equal to x=0, y=-.96, meaning the function is symmetric about x = 0
* The hessian at this point is equal to x=0, y=-.96, meaning the function is symmetric about x = 0

1. Fixed Step Size
   1. This code employs the use of a fixed step size to converge to the minimal solution
   2. The Fixed Step Size Gradient Method requires the step size to be larger than 0, but smaller than 2/lambda(max) if it is to converge. Larger step sizes require fewer steps to converge, but have less accuracy. Smaller lambda take more steps to converge, but increase the accuracy of the estimates
   3. The algorithm was developed to converge for any point that does not go to (0,0). The origin contains a local maximum, and is symmetrical in every direction. Therefore, if the algorithm reaches this point it is considered convergent given the availability of 4 equidistant minimums of the same value.
   4. Q = [1,3;1,3] , 0 = (1-x)(1-x)-9 = x2-2x-8 = (x-4)(x+2) -> x=4,-2
   5. In order to converge, alpha must be larger than 0 and less than .5
   6. For the following plots, I choose alpha = .05, .15, .3, .45 (yellow=high, blue=low)

Alpha = .05

* 1. The first plot shows the number of steps it took the algorithm to converge, with blue being low, and yellow being high. The second plot shows the real values of the function at the value the algorithm converged. The algorithm was designed to converge to either 0, the minimum of the function, or 1, which is the local maximum with 4-quadrant symmetry at (0,0)
  2. You can see from the two plots above that the size of the step involved in the algorithm determines how the various points converge. With a larger alpha, it takes fewer steps for a point to “converge”, but the accuracy of the algorithm to converge to the real minimum is reduced. With a larger alpha, the algorithm takes larger steps toward the origin, and therefore can’t correct its direction before getting stuck.
  3. With a smaller alpha, the number of steps it takes to get to a minimal value is much larger, but the number of points that converge to the minimum is closer to 100%. The points along the steepest gradient descent however, still get stuck at the origin.

1. Newton’s Method with Backtracking
   1. This code employs the use of Newton’s method with 1st and 2nd derivatives to determine the direction of the iterations, as well as backtracking to determine the optimal size of alpha at each iteration
   2. Values of B = .8 and A = .4 were chosen for the backtracking algorithm based on researched values, and the algorithm was considered to converge when the gradient or the hessian is equal to 0. If the algorithm tries to push the iterations into negative values, the algorithm stops, records the number of steps, and then records the value at that point

1. Conjugate gradient method with Fletcher-Reeves formula
   1. This code employs the use of conjugate gradient directions to dictate the descent of the algorithm to minimum solutions, with the fletcher reeves formula to determine the step sizes of the subsequent directions
   2. This method was not as effective, do to the way in which the surface is shaped. The conjugate gradient direction method follows the path of steepest descent in orthogonal directions. However, do to the rate at which the function changes values from its outermost corners to the center, the step size is drastically over estimated. Additionally, when the x and y values become sufficiently close to 0, the gradient dissolves to a constant, and value, and increases without bound. Therefore, if the algorithm iterates to any points near zero, the algorithm will converge, but not to the optimal value, and instead to 1.