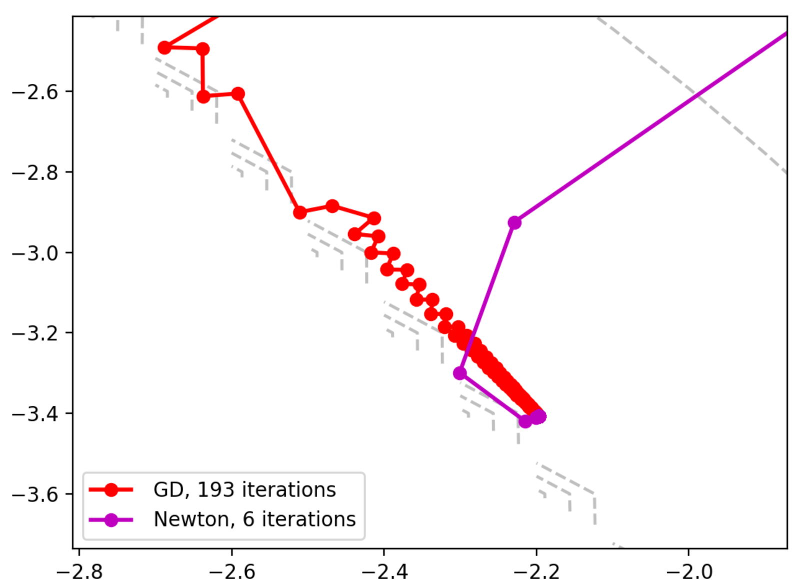
CSE592 Convex Optimization: Home Work 3

Ashik Vetrivelu (111652994)

April 11, 2018

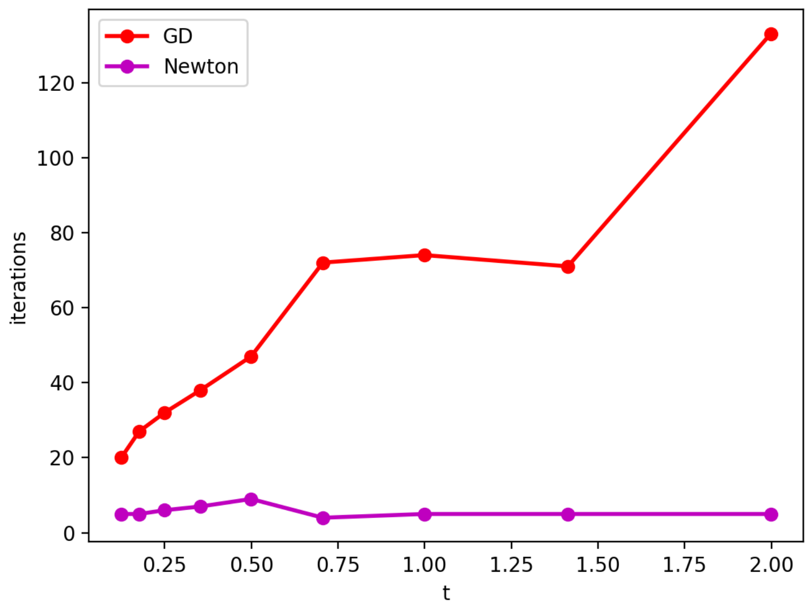
1.2.1 Newton vs Gradient Descent

Here, we ran the main_quadratic function on both the Newton and Gradient Descent functions. We can clearly see that the gradient descent takes many iterations to converge whereas the Newton takes less amount of iterations to converge. Precisely, gradient descent takes 193 iterations whereas Newton takes around 6 iterations. We'll further experiment about this in the next problem. Here the value of t was a bit too large for the Gradient Descent as we can see from the inner iterations.The inner iterations are more only when the t value is large.



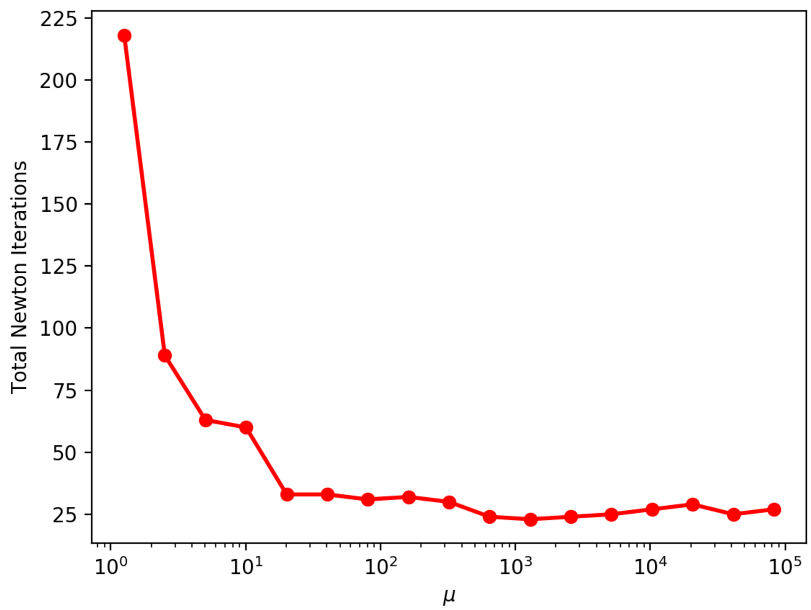
1.2.2 Gradient Descent, Newton t vs iterations

The problem imposed in *1.2.1* can be solved here. If we look at the plot, we can see that the gradient descent iterations increase significantly after a certain t value. The increase might be explained due to the increased number of outer and inner contour iterations. Our goal is to minimize the central path condition so that we can get the perfect value of t for which the iterations are less. Also Gradient Descent itself depends on the function condition number to decide the number of iterations to solve whereas the Newton doesn't. The newton exhibits a static iteration count in this case where there ain't any significant changes to the number of iterations. This might even be a worse t value for Newton Algorithm as we didn't compare it exclusively with other values of t higher than 3.



2.2.1 μ vs iterations

Here we can get an interesting insight about the μ value and number of iterations. The Barrier method can suprisingly work for upto 10^4 values of μ in our case. The iterations are higher for μ values upto some 10 which might be explained due to the increased number of outer iterations at this value. But as the value of μ increases the number of iterations stabilizes and doesn't show any significant change. I don't probably think there'll be a significant change by expanding the value of μ.



2.2.2

The plot does indeed meet our expectations and could have been better given it's just a linear function optimization and it's newton. For linear function optimization, gradient descent would have given a better solution with less iterations in my opinion and it doesn't need to calculate the outer and inner iterations for every constraint.

2.2.3

The solution to the problem is 30610.00006064 for x value at [[48.00000121], [30.99999886], [38.99999764], [43.00000243, [