Assignment 2: Computational Methods of Optimization

Godwin Khalko
Roll Number: 24006

October 2024

Problem 1: Conjugate Gradient Descent

(1) **Suppose A is an $n \times n$ symmetric, PD matrix. Does this equation $Ax = b$ have a unique solution? State a convex quadratic minimization problem whose optimal solution is x^* such that $Ax^* = b$.**

Solution:
Yes, the equation $\mathbf{Ax = b}$ has a unique solution since the A is a n x n matrix. The unique solution to this problem would be $x = A^{-1}b$. This is because the inverse of A exists as it is a **PD** matrix.

A convex quadratic minimization problem whose optimal solution is x^* such that $Ax^* = b$ could be

$$\min_x \frac{1}{2} x^T A x - b^T x$$

This is because the gradient of the above function would be $\nabla f(x) = x^T A - b$. To minimise our optimization problem, we need to make $\nabla f(x^*) = 0 \implies Ax^* - b = 0 \implies Ax^* = b$

(2) Query the oracle for a PD matrix A and a vector b. Implement conjugate gradient descent on the optimization problem in the previous part. Report the optimum x^* and the number of iterations it took to reach the optimum.

Solution:
The minimiser reached at : [2. 5. 4. 9. 2.]
The number of iterations: 5

(3) Suppose A is an $m \times n$ matrix with $m > n$. Does $Ax = b$ have a unique solution? One strategy to solve such equations is to minimize the error term $||Ax - b||$. Write down a quadratic minimization problem to minimize this error term. When does it have a unique solution?

Solution:
In the scenario where **A** is an m×n matrix with m¿n (meaning there are more rows than columns, i.e., an over-determined system), the equation **Ax=b** may not have a unique solution or may not have any solution at all. The objective is to minimize the error term ||Ax − b||. For this we can minimize the norm:

$$\min_x ||Ax - b||^2$$

This is equivalent to minimizing the quadratic objective function:

$$f(x) = x^T A^T Ax - 2b^T Ax + b^T b$$

A unique solution to the least squares problem exists if and only if $A^T A$ is invertible (i.e., non-singular). For $A^T A$ to be invertible, the columns of A must be linearly independent, which implies that A has full rank (i.e., rank n).

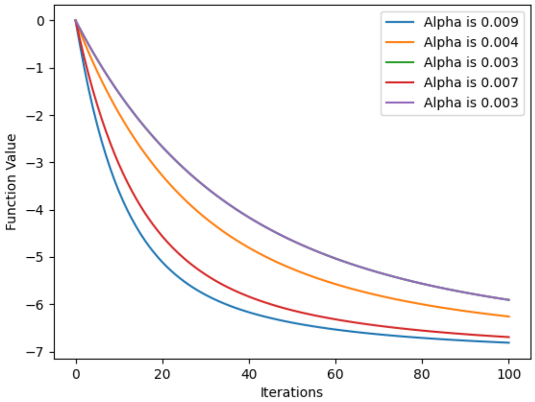
(4) Query the oracle for an $m \times n$ matrix A with $m > n$ and a $m \times 1$ vector b. Find an x^* that minimizes the error term $||Ax - b||$. Report the optimal x^* and the number of iterations it took to reach the optimum.

Solution:
The minimiser reached at : [2. 5. 8. 7. 1.99e^{15}]
The number of iterations: 1

Problem 2: Newton's Method

(1) Run gradient descent (implemented in Assignment 1) for 100 iterations with five choices of step-sizes with x_0 as the initial point. Plot the function value at each iteration for each choice of step-size. Report x at the final iteration.

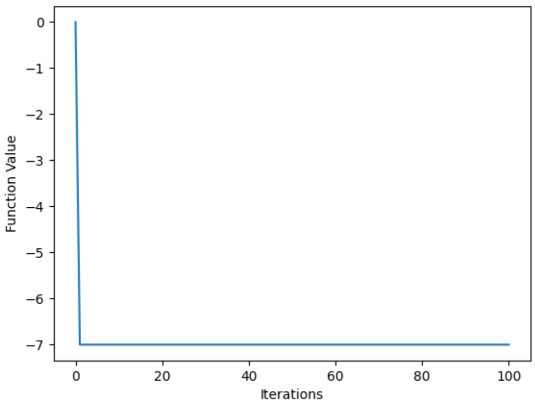
Solution:



Final point for alpha = 0.009 is [0.84 0.97 0.59 0.59 0.99]
Final point for alpha = 0.004 is [0.55 0.80 0.33 0.33 0.91]
Final point for alpha = 0.003 is [0.45 0.70 0.26 0.26 0.84]
Final point for alpha = 0.007 is [0.76 0.94 0.50 0.50 0.98]
Final point for alpha = 0.003 is [0.45 0.70 0.26 0.26 0.84]

(2) Execute Newton's method for 100 iterations starting from x_0, and plot the function values at each iteration. Compare the results with those from gradient descent. Report x at the final iteration.

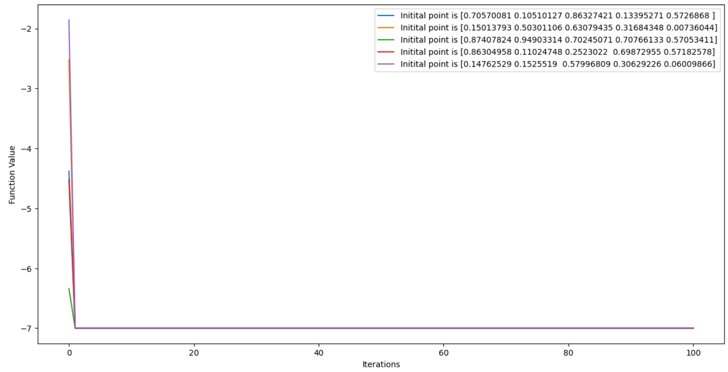
Solution:

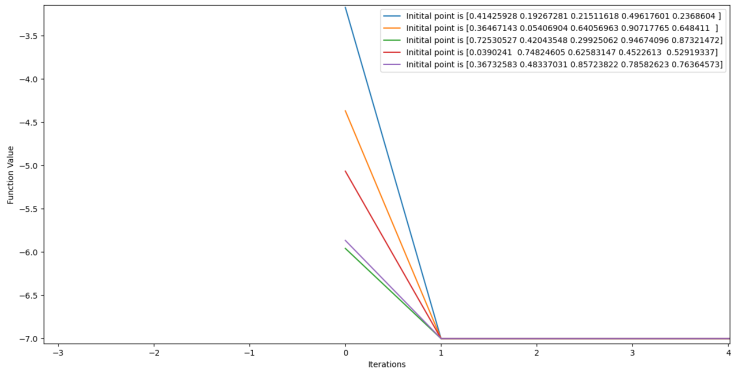


final Points of the Newton is : [1. 1. 1. 1. 1.]

(3) Execute Newton's method for 100 iterations starting from five different initial points and plot the function values at each iteration. What do you observe?

Solution:





Some observations that can be made are that:

- No matter which initial point, we converge to the minima in one step
- Hessian of the function must be PD at every point of the domain.

(4) Based on the above observation, guess the nature of the function in the oracle, and prove that the observation holds for such functions.

Solution:
The function seems to be a convex quadratic function with the hessian of the function being PD. Since the function is convex quadratic, Consider a quadratic function:

$$\frac{1}{2}x^TQx + b^Tx + c$$

Where Q is a PD matrix. The gradient of the function,

$$\nabla f(x) = Qx - b$$

$$H_f(x) = Q$$

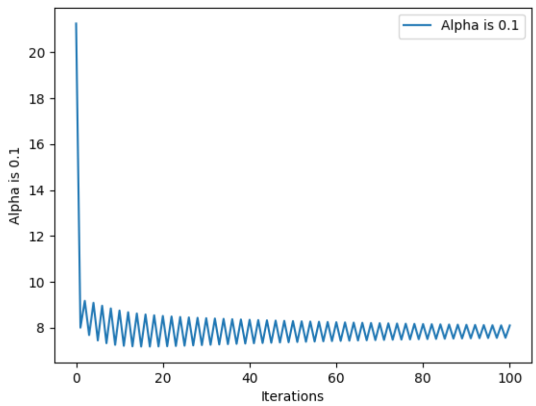
Equating $\nabla f(x)$ to 0, we get $x^* = -Q^{-1}b$ and since Q is PD, it is invertible and we can reach the minima in one step.

Problem 3: Newton's Method continued

(1) Run gradient descent with a step-size of 0.1 for 100 iterations and plot the function values. Report the best function value obtained over all the iterations.

Solution:

Best function value observed over all the 100 iterations 4.48



(2) Did you observe oscillations in the above plot? Try to guess possible reasons for this phenomenon, and suggest ways to overcome them.

Solution:

The reason for the oscillation in the above plot is because due to the big alpha value, the algorithm overshoots from the minima point a few iterations, before converging at the minima.

(3) Execute Newton for 100 iterations and report the function values for only the first 10 iterations. Did the algorithm converge? If not, why?

Solution:

function value at iteration 0 is : 14.272347500570127
function value at iteration 1 is : inf
function value at iteration 2 is : inf

function value at iteration 3 is : nan
function value at iteration 4 is : nan
function value at iteration 5 is : nan
function value at iteration 6 is : nan
function value at iteration 7 is : nan
function value at iteration 8 is : nan
function value at iteration 9 is : nan

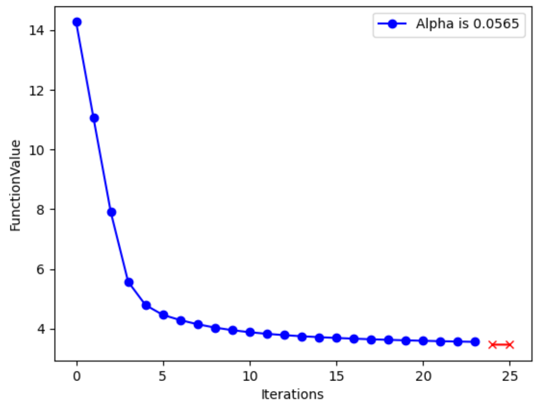
No, the function didn't converge. For Newton method to work properly, we need the Hessian of the function at a point x^k to be Positive Definite, since we'll be taking the inverse of the matrix to compute the iteration points. We can only guarantee that the points that are at a distance of $\frac{a}{M}$ from the minima x^*. Since we're not close to x^*, the Hessian at the iteration points could be non-invertible resulting in 'nan' values.

(4) Report the least observed cost, the corresponding function-value and x at the last iteration for that value of K. Also, plot the function-values at each iteration,

Solution:
Least observed cost is 73
X point value is

$$[5.31e^{-18}, 1.14e^{-17}, 5.09e^{-06}, 5.09e^{-06}, -1.614e^{-15}]$$

Corresponding function value is 3.46
The value of K was 23



Problem 4: Quasi-Newton Methods

(1) Derive an update expression for approximating the Hessian by a scalar multiple of the identity matrix in a Quasi-Newton method.

Solution:
Let B^{k+1} be the approximation of the H^{-1} at the k+1th iteration and the B^k be the approximation of the H^{-1} at the kth iteration. Let the k+1th and the kth iteration points be $x^{k+1}$and x^k.
Let the gradient at k+1th and the kth point be $g^{k+1}$and g^k. Let approximate the B^{k+1} via scalar multiple of I. So, we denote

$$\delta^k = x^{k+1} - x^k$$
$$\gamma^k = g^{k+1} - g^k$$

Since, B^{k+1} can be calculated by the addition of B^k and scalar multiple of I

$$B^{k+1} = B^k + \lambda I$$

Multiplying on both the sides with γ^k

$$\begin{aligned}
B^{k+1}\gamma^k &= B^k\gamma^k + \lambda I\gamma^k \\
\delta^k &= B^k\gamma^k + \lambda\gamma^k \\
\gamma^k\delta^k &= \gamma^k B^k\gamma^k + \lambda\gamma^k\gamma^k \\
\lambda &= \frac{\gamma^k\delta^k - \gamma^k B^k\gamma^k}{\gamma^k\gamma^k}
\end{aligned}$$

So, with this value of the λ, we can use Identity matrix for quasi-Newton conditions.

(2) Use the oracle from Problem 2 to compare Quasi-Newton solutions with gradient descent. Report x^* obtained by each method and plot the values of f over 100 iterations.

Solution:
Final X for α : 0.05 is [1. 1. 0.99 0.99 1.]
Final X for α : 0.001 is [0.18 0.33 0.1 0.1 0.45]
Final X for α : 0.051 is [1. 1. 0.99 0.99 1.]
Final X for α : 0.025 is [0.99 1. 0.92 0.92 1.]
Final X for α : 0.003 is [0.45 0.7 0.26 0.26 0.84]

Final X at Quasi Newton: [1. 1. 1. 1. 1.]