

Instructions: (Please read carefully and follow them!)

Try to solve all problems on your own. If you have difficulties, ask the instructor or TAs.

In this session, we will continue with the implementation of gradient descent algorithm to solve problems of the form $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$. Recall that gradient descent works iteratively where in each iteration a suitable descent direction and an appropriate step length are chosen to update the current iterate. We will discuss a new idea to find the step length parameter.

The implementation of the optimization algorithms in this lab will involve extensive use of the `numpy` Python package. It would be useful for you to get to know some of the functionalities of `numpy` package. For details on `numpy` Python package, please consult <https://numpy.org/doc/stable/index.html>

For plotting purposes, please use `matplotlib.pyplot` package. You can find examples in the site <https://matplotlib.org/examples/>.

Please follow the instructions given below to prepare your solution notebooks:

- Please use different notebooks for solving different Exercise problems.
- The notebook name for Exercise 1 should be `YOURROLLNUMBER_IE684_Lab2_Ex1.ipynb`.
- Similarly, the notebook name for Exercise 2 should be `YOURROLLNUMBER_IE684_Lab2_Ex2.ipynb`, etc.

There are only 3 exercises in this lab. Try to solve all problems on your own. If you have difficulties, ask the Instructors or TAs.

Only the questions marked **[R]** need to be answered in the notebook. You can either print the answers using `print` command in your code or you can write the text in a separate text tab. To add text in your notebook, click **+Text**. Some questions require you to provide proper explanations; for such questions, write proper explanations in a text tab. Some questions require the answers to be written in LaTeX notation. Please see the demo video to know how to write LaTeX in Google notebooks. Some questions require plotting certain graphs. Please make sure that the plots are present in the submitted notebooks. Please include all answers in your `.pynb` files.

After completing this lab's exercises, click File → Download `.ipynb` and save your files to your local laptop/desktop. Create a folder with name `YOURROLLNUMBER_IE684_Lab2` and copy your `.ipynb` files to the folder. Then zip the folder to create `YOURROLLNUMBER_IE684_Lab2.zip`. Then upload only the `.zip` file to Moodle. There will be extra marks for students who follow the proper naming conventions in their submissions.

Please check the **submission deadline announced in moodle**.

Exercise 1: Backtracking Line Search

Recall that we implemented the gradient descent algorithm to solve $\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$. The main ingredients in the gradient descent iterations are the descent direction \mathbf{p}^k which is set to $-\nabla f(\mathbf{x}^k)$, and the step length η^k which is found by solving an optimization problem (or sometimes taken as a constant value over all iterations). We used the following procedure in the previous lab:

Input: Starting point \mathbf{x}^0 , Stopping tolerance τ
Initialize $k = 0$
 $\mathbf{p}^k = -\nabla f(\mathbf{x}^k)$.
while $\|\mathbf{p}^k\|_2 > \tau$ **do**
 $\eta^k = \arg \min_{\eta \geq 0} f(\mathbf{x}^k + \eta \mathbf{p}^k) = \arg \min_{\eta \geq 0} f(\mathbf{x}^k - \eta \nabla f(\mathbf{x}^k))$
 $\mathbf{x}^{k+1} = \mathbf{x}^k + \eta^k \mathbf{p}^k = \mathbf{x}^k - \eta^k \nabla f(\mathbf{x}^k)$
 $k = k + 1$
end
Output: \mathbf{x}^k .

Algorithm 1: Gradient Descent Procedure with line search to find step length

We saw that for particular quadratic functions, a closed form analytical solution for the minimizer of the optimization problem $\min_{\eta \geq 0} f(\mathbf{x}^k + \eta \mathbf{p}^k)$ exists. However finding a closed form expression as a solution to this optimization problem to find a suitable step length might not always be possible. To tackle general situations, we will try to devise a different procedure in this lab.

To find the step length, we will use the following property: Suppose a non-zero $\mathbf{p} \in \mathbb{R}^n$ is a descent direction at point \mathbf{x} , and let $\gamma \in (0, 1)$. Then there exists $\varepsilon > 0$ such that

$$f(\mathbf{x} + \alpha \mathbf{p}) \leq f(\mathbf{x}) + \gamma \alpha \nabla f(\mathbf{x})^\top \mathbf{p}, \quad \forall \alpha \in (0, \varepsilon].$$

This condition is called a **sufficient decrease condition**.

Using the idea of sufficient decrease, the step length η^k can be found using a backtracking procedure illustrated below to find appropriate value of ε .

Input: $\mathbf{x}^k, \mathbf{p}^k, \alpha^0, \rho \in (0, 1), \gamma \in (0, 1)$
Initialize $\alpha = \alpha^0$
 $\mathbf{p}^k = -\nabla f(\mathbf{x}^k)$.
while $f(\mathbf{x}^k + \alpha \mathbf{p}^k) > f(\mathbf{x}^k) + \gamma \alpha \nabla f(\mathbf{x}^k)^\top \mathbf{p}^k$ **do**
 $\alpha = \rho \alpha$
end
Output: α .

Algorithm 2: Backtracking line search

1. Note that in the notebook file shared with you, the gradient descent algorithm is implemented to solve $\min_{\mathbf{x}} f(\mathbf{x}) = (x_1 - 8)^2 + (x_2 + 12)^2$ with a constant step length $\eta = 0.1$, stopping tolerance $\tau = 10^{-5}$ and with the starting point $\mathbf{x}^0 = (1, 1)$.
2. Implement the methods to find step length using exact line search (closed form expression), and using backtracking line search for the function $f(\mathbf{x}) = f(x_1, x_2) = (x_1 - 8)^2 + (x_2 + 12)^2$.
3. [R] Note down the minimizer and minimum function value of $f(\mathbf{x}) = f(x_1, x_2) = (x_1 - 8)^2 + (x_2 + 12)^2$.
4. [R] Consider stopping tolerance $\tau = 10^{-12}$ and starting point $\mathbf{x}^0 = (25, 25)$. Compare the number of iterations taken by the gradient descent procedure which uses exact step length computation against the gradient descent

procedure which uses the backtracking line search procedure (with $\alpha^0 = 1, \rho = 0.5, \gamma = 0.5$). Comment on your observations.

5. [R] With starting point $\mathbf{x}^0 = (25, 25)$ and $\tau = 10^{-10}$, we will now study the behavior of the backtracking line search algorithm for different choices of α^0 . Take $\gamma = \rho = 0.5$. Try $\alpha^0 \in \{1, 0.9, 0.75, 0.6, 0.5, 0.4, 0.25, 0.1, 0.01\}$. For each α^0 , record the final minimizer, final objective function value and number of iterations taken by the gradient descent algorithm with backtracking line search to terminate. Prepare a plot where the number of iterations is plotted against α^0 values. Comment on the observations. Comment about the minimizers and objective function values obtained for different choices of the α^0 values. Check and comment if for any α^0 value, gradient descent with backtracking line search takes lesser number of iterations when compared to the gradient descent procedure with exact line search.
6. [R] With starting point $\mathbf{x}^0 = (25, 25)$ and $\tau = 10^{-10}$, we will now study the behavior of the backtracking line search algorithm for different choices of ρ . Take $\alpha = 1, \gamma = 0.5$. Try $\rho \in \{0.9, 0.75, 0.6, 0.5, 0.4, 0.25, 0.1, 0.01\}$. For each ρ , record the final minimizer, final objective function value and number of iterations taken by the gradient descent algorithm with backtracking line search to terminate. Prepare a plot where the number of iterations is plotted against ρ values. Comment on the observations. Comment about the minimizers and objective function values obtained for different choices of the ρ values. Check and comment if for any ρ value, gradient descent with backtracking line search takes lesser number of iterations when compared to the gradient descent procedure with exact line search.

Exercise 2:

Consider the function

$$h(\mathbf{x}) = \sum_{i=1}^N \frac{1}{8^i} (x_i - 2^i)^2.$$

1. Consider $N = 3$. Implement the `evalf`, `evalg` and other related functions for implementing gradient descent (with exact line search and with backtracking line search) to solve $\min_{\mathbf{x}} h(\mathbf{x})$.
2. [R] Note down the minimizer and minimum function value of $h(\mathbf{x})$.
3. [R] Consider stopping tolerance $\tau = 10^{-10}$ and starting point $\mathbf{x}^0 = (\frac{1}{64}, \frac{1}{8}, 1)$. Compare the number of iterations taken by the gradient descent procedure which uses exact step length computation against the gradient descent procedure which uses the backtracking line search procedure (with $\alpha^0 = 1, \rho = 0.5, \gamma = 0.5$). Comment on your observations.
4. [R] Check if similar observations hold in the $N = 4$ case as well. Choose a starting point $\mathbf{x}^0 = (\frac{1}{512}, \frac{1}{64}, \frac{1}{8}, 1)$ and for the backtracking line search procedure, use $\alpha^0 = 1, \rho = 0.5, \gamma = 0.5$. Comment on your observations.
5. [R] Can you also comment on the possible observations for $N > 4$ case as well, without actually running the program?

Exercise 3:

Consider the function

$$q(\mathbf{x}) = 512(x_2 - x_1^2)^2 + (4 - x_1)^2.$$

1. [R] Find the minimizer of the function $q(\mathbf{x})$.
 2. [R] Can you use the exact line search idea to find a closed form expression for solving $\min_{\eta} q(\mathbf{x} + \eta \mathbf{p})$ where $\mathbf{p} \neq \mathbf{0}$ is a descent direction at \mathbf{x} ? If yes, find the closed form expression for optimal value of η . If you cannot find closed form solution, explain the reasons.
 3. If you could use the exact line search procedure to solve $\min_{\eta} q(\mathbf{x} + \eta \mathbf{p})$ for finding the step length, implement the procedure. If not, you can ignore the implementation.
 4. Write appropriate code to implement the gradient descent procedure with backtracking line search to solve $\min_{\mathbf{x}} q(\mathbf{x})$.
 5. [R] With starting point $\mathbf{x}^0 = (100, 100)$ and $\tau = 10^{-10}$, we will now study the behavior of the backtracking line search algorithm for different choices of α^0 . Take $\gamma = \rho = 0.5$. Try $\alpha^0 \in \{1, 0.9, 0.75, 0.6, 0.5, 0.4, 0.25, 0.1, 0.01\}$. For each α^0 , record the final minimizer, final objective function value and number of iterations taken by the gradient descent algorithm with backtracking line search to terminate. Prepare a plot where the number of iterations is plotted against α^0 values. Comment on the observations. Comment about the minimizers and objective function values obtained for different choices of the α^0 values. If you have implemented exact line search, check and comment if for any α^0 value, gradient descent with backtracking line search takes lesser number of iterations when compared to the gradient descent procedure with exact line search.
 6. [R] With starting point $\mathbf{x}^0 = (100, 100)$ and $\tau = 10^{-10}$, we will now study the behavior of the backtracking line search algorithm for different choices of ρ . Take $\alpha = 1, \gamma = 0.5$. Try $\rho \in \{0.9, 0.75, 0.6, 0.5, 0.4, 0.25, 0.1, 0.01\}$. For each ρ , record the final minimizer, final objective function value and number of iterations taken by the gradient descent algorithm with backtracking line search to terminate. Prepare a plot where the number of iterations is plotted against ρ values. Comment on the observations. Comment about the minimizers and objective function values obtained for different choices of the ρ values. If you have implemented exact line search, check and comment if for any ρ value, gradient descent with backtracking line search takes lesser number of iterations when compared to the gradient descent procedure with exact line search.
-