EEE319 Optimisation
Lecture 8 Nonlinear Programming (2)

Prof. Xinheng Wang

xinheng.wang@xjtlu.edu.cn

Office: EE512

Nonlinear programming

- Last week
  - Nonlinear programming
    - Single variable nonlinear programming
    - Line search
      - Bisection
      - Golden section
      - Newton method
      - Secant method

- Today
  - Nonlinear programming
    - Multiple variables

Prerequisite of Today's Lecture

- Matrix and notations of matrix
- A matrix is usually denoted by a capital letter printed in a boldface font, for example **A**, **B** and **X**. Following is a matrix **A** of dimension $m \times n$.

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \\ a_{m1} & a_{m1} & \dots & a_{mn} \end{bmatrix}$$

- Matrix **A** is also denoted as

$$\mathbf{A} = [a_{i,j}], i = 1,2, \dots, m; j = 1, 2, \dots, n.$$

It is denoted as a matrix with *m* rows and *n* columns, whose typical element is $a_{i,j}$.

Prerequisite of Today's Lecture

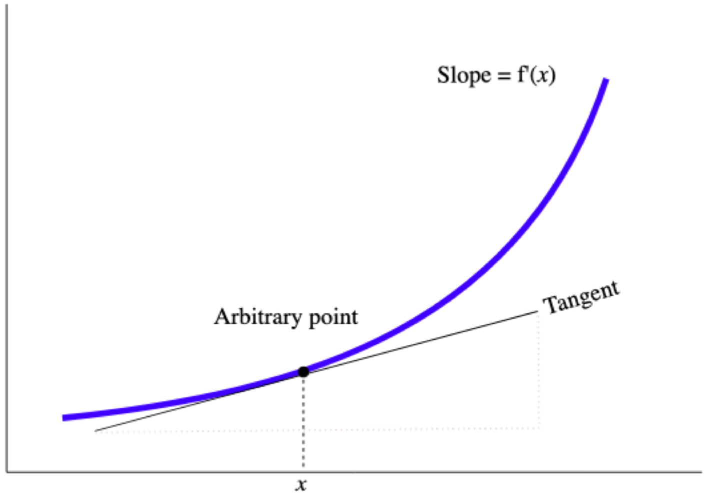
- A superscript ^{T} denotes the matrix transpose operation; for example $\mathbf{A}^T$ denotes the transpose of $\mathbf{A}$.
- if $\mathbf{A}$ has an inverse it will be denoted by $\mathbf{A}^{-1}$.
- The determinant of $\mathbf{A}$ will be denoted by either $|\mathbf{A}|$ or $\det(\mathbf{A})$.

Prerequisite of Today’s Lecture

- Vector and notations of vector
- Vector is a matrix with only one row or one column. A matrix having only one row is called a row vector. A matrix having only one column is called a column vector.
- Vector is denoted by boldfaced lowercase letters, i.e., **a**, **b**, and **x**.

Prerequisite of Today's Lecture

- Derivative vs partial derivative vs gradient
  - **Derivative** demonstrates the instantaneous rate of change. We can think of the derivative as of a tangent line's slope at some point on a graph.
  - It is normally noted as df/dx, derivative of function f with regard to x. The ***dx*** part means: a small change in the variable ***x***. And the ***df*** part means: the small change in the output of the function ***f***. All together it answers the question: what is the function's change (output) if some change in ***x*** (input) has occurred?



Prerequisite of Today's Lecture

- Derivative vs partial derivative vs gradient
  - Partial Derivative is used for a function with multiple variables. It demonstrates the change of one variable, and keeping the other variables as constants.
  - For example $f(x, y) = x^4 y^3$
  - Notation of partial derivative
    - Partial derivative of function *f* with respect to *x*
$$\frac{\partial f}{\partial x} = 4x^3 y^3$$
    - Partial derivative of function *f* with respect to *y*
$$\frac{\partial f}{\partial y} = 3x^4 y^2$$
    - A ∂, del symbol is normally used.

Prerequisite of Today’s Lecture

- Derivative vs partial derivative vs gradient
  - The gradient **holds all the partial derivatives** of a multivariable function. Symbol ∇ nabla is normally used to denote the gradient.
  - Derivative is a number. Gradient is a vector (value and direction).
  - **The gradient points to the direction of greatest increase; keep following the gradient, and you will reach the local maximum.**
  - **Important Symbols**: use *d* for the derivative; ∂ for the partial derivative and ∇ for the gradient.
  - Gradient example: function $f = x + y^2 + z^3$

$$\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}\right) = (1, 2y, 3z^2)$$

If x=3, y=4, z=5, direction is (1, 8, 75)

Prerequisite of Today's Lecture

- Jacobian Matrix
  - Given a function *f* with multiple (*n*) variables, the Jacobian Matrix of this function is the function's gradient.

$$\mathrm{J} = \nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \frac{\partial f}{\partial x_3} & \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix}$$

  - If we have *m* functions, $f_1, f_2, \ldots, f_m$ with each function having *n* variables, denoted as a vector function **f**, the Jacobian matrix is:

$$\mathrm{J} = \nabla \mathbf{f} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$$

Jacobian Matrix in Multivariable Newton Method

- A few definitions
  - Given *m* functions, f_1, f_2, …, f_m with each function having *n* variables, the functions are denoted by a vector function **f**. Variables $x_1, x_2, \ldots, x_n$ are denoted by a vector $\mathbf{x}$.
  - Aim is to solve equation **f**(**x**)=**0** by using Newton method

where **0**=[0, 0, …, 0]

  - Newton method with multivariable:

$$\mathbf{x}_{i+1} = \mathbf{x}_i - \frac{\mathbf{f}}{\mathbf{J(f)}} = \mathbf{x}_i - \mathbf{J}^{-1}\mathbf{f}$$

Jacobian Matrix in Multivariable Newton Method

- Example
  - Solving 2-variable functions:

$$x + xy - 4 = 0$$
$$x + y - 3 = 0$$

  - Steps to follow
    - Step 1 Compute the Jacobian Matrix
    - Step 2 Initiate the iteration with i=0, $\mathbf{x}_0$,and also initial values of variables
    - Step 3 Compute $\mathbf{x}_1$
    - Step 4 Check the difference of $|\mathbf{x}_1 - \mathbf{x}_0|$. If it is lower than a predefined tolerance ϵ, $|\mathbf{x}_1 - \mathbf{x}_0| < \epsilon$, the algorithm has converged and the solution is $\mathbf{x}_1$. Otherwise, update the iteration and continue.

Jacobian Matrix in Multivariable Newton Method

- Example
  - Solving 2-variable functions:

$$x + xy - 4 = 0$$
$$x + y - 3 = 0$$

  - Set $f_1 = x + xy - 4$; $f_2 = x + y - 3$.
  - Compute partial derivative

$$\frac{\partial f_1}{\partial x} = 1 + y;\ \frac{\partial f_1}{\partial y} = x.$$

$$\frac{\partial f_2}{\partial x} = 1;\ \frac{\partial f_2}{\partial y} = 1.$$

  - Compute Jacobian Matrix

$$\mathbf{J} = \begin{bmatrix} 1 + y & x \\ 1 & 1 \end{bmatrix}$$

Jacobian Matrix in Multivariable Newton Method

- Example
  - Step 1 Start the iteration with *i*=0 and initial guess, i.e., $x_0 = 1.98; y_0 = 1.02$. We have

$$\mathbf{x}_0 = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \begin{bmatrix} 1.98 \\ 1.02 \end{bmatrix}$$

  - Step 2 Compute $\mathbf{x}_1$

$$\mathbf{x}_1 = \mathbf{x}_0 - \mathbf{J_0}^{-1}\mathbf{f_0}$$

  - Recap Jacobian Matrix

$$\mathbf{J} = \begin{bmatrix} 1+y & x \\ 1 & 1 \end{bmatrix}$$

Therefore, we have

$$\mathbf{J_0} = \begin{bmatrix} 1+y_0 & x_0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 2.02 & 1.98 \\ 1 & 1 \end{bmatrix}$$

$$\mathbf{f_0} = \begin{bmatrix} f_{10} \\ f_{20} \end{bmatrix} = \begin{bmatrix} x_0 + x_0 y_0 - 4 \\ x_0 + y_0 - 3 \end{bmatrix} = \begin{bmatrix} 1.98 + 1.98 * 1.02 - 4 \\ 1.98 + 1.02 - 3 \end{bmatrix} = \begin{bmatrix} -0.0004 \\ 0 \end{bmatrix}$$

Jacobian Matrix in Multivariable Newton Method

- Example
  - Recap

$$\mathbf{x}_1 = \mathbf{x}_0 - \mathbf{J_0}^{-1}\mathbf{f_0}$$

$$\mathbf{J_0} = \begin{bmatrix} 1 + y_0 & x_0 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 2.02 & 1.98 \\ 1 & 1 \end{bmatrix}$$

$$\mathbf{f_0} = \begin{bmatrix} f_{10} \\ f_{20} \end{bmatrix} = \begin{bmatrix} x_0 + x_0 y_0 - 4 \\ x_0 + y_0 - 3 \end{bmatrix} = \begin{bmatrix} 1.98 + 1.98 * 1.02 - 4 \\ 1.98 + 1.02 - 3 \end{bmatrix} = \begin{bmatrix} -0.0004 \\ 0 \end{bmatrix}$$

  - We have

$$\mathbf{x}_1 = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} - \mathbf{J_0}^{-1}\mathbf{f_0} = \begin{bmatrix} 1.98 \\ 1.02 \end{bmatrix} - \begin{bmatrix} 2.02 & 1.98 \\ 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} -0.0004 \\ 0 \end{bmatrix}$$

Jacobian Matrix in Multivariable Newton Method

- Example
  - We have

$$\mathbf{x}_1 = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} - \mathbf{J_0}^{-1}\mathbf{f_0} = \begin{bmatrix} 1.98 \\ 1.02 \end{bmatrix} - \begin{bmatrix} 2.02 & 1.98 \\ 1 & 1 \end{bmatrix}^{-1} \begin{bmatrix} -0.0004 \\ 0 \end{bmatrix}$$

  - Recall on the calculation of inverse matrix

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

In other word, **swap** the positions of *a* and *d*, put **negatives** in front of *b* and *c*, and **divide** everything by the determinant (*ad*-*bc*)

$$\begin{bmatrix} 2.02 & 1.98 \\ 1 & 1 \end{bmatrix}^{-1} = \frac{1}{2.02 * 1 - 1.98 * 1} \begin{bmatrix} 1 & -1.98 \\ -1 & 2.02 \end{bmatrix} = \begin{bmatrix} 25 & -24.5 \\ -25 & 50.5 \end{bmatrix}$$

$$\mathbf{x}_1 = \begin{bmatrix} 1.98 \\ 1.02 \end{bmatrix} - \begin{bmatrix} 25 & -24.5 \\ -25 & 50.5 \end{bmatrix} \begin{bmatrix} -0.0004 \\ 0 \end{bmatrix} = \begin{bmatrix} 1.98 \\ 1.02 \end{bmatrix} - \begin{bmatrix} 25 * (-0.0004) + (-24.5) * 0 \\ (-25) * -0.0004 + 0 \end{bmatrix} = \begin{bmatrix} 1.98 \\ 1.02 \end{bmatrix} - \begin{bmatrix} -0.01 \\ 0.01 \end{bmatrix}$$
$$= \begin{bmatrix} 1.99 \\ 1.01 \end{bmatrix}$$

Jacobian Matrix in Multivariable Newton Method

- Example
  - Step 3 Compute the difference between $\mathbf{x}_1$ and $\mathbf{x}_0$. Normally ϵ is taken as a very small value, i.e., 10^{-4}.
  - Difference of variable *x* is
$$|x_1 - x_0| = |1.99 - 1.98| = 0.01 > \epsilon$$
  - Difference of variable *y* is
$$|y_1 - y_0| = |1.01 - 1.02| = 0.01 > \epsilon$$
  - Step 4 Continue to next iteration
  - …

Jacobian Matrix in Multivariable Newton Method

- Example
  - Iterations

Iteration	x	y
0	1.9800	1.0200
1	1.9900	1.0100
2	1.9950	1.0050
3	1.9975	1.0025
4	1.9987	1.0013
5	1.9994	1.0006
6	1.9997	1.0003
7	1.9998	1.0002
8	1.9999	1.0001
9	2.0000	1.0000

THANK YOU

VISIT US

WWW.XJTLU.EDU.CN

FOLLOW US

@XJTLU

Xi'an Jiaotong-Liverpool University
西交利物浦大學