# Training and Workshop on

# Python Programming and Numerical Methods

## *An Introduction to Numerical Linear Algebra and Differential Equations*

**January 29-31, 2025**

**2F CCE Building, Visayas State University,**
**Visca, Baybay City, Leyte**

# Optimization

Arrianne Crystal Velasco

Institute of Mathematics
University of the Philippines Diliman

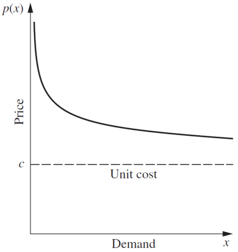
January 30, 2025

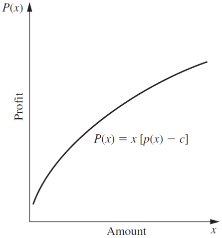
# UNCONSTRAINED OPTIMIZATION

**Example 1. Product-mix problem with price elasticity**[1]

Nonlinearities arise from

- price elasticity - the amount of product that can be sold has an inverse relationship to the price charged.



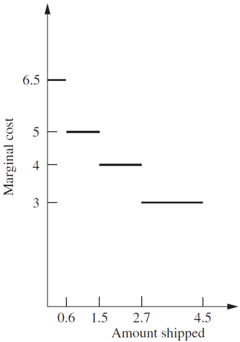



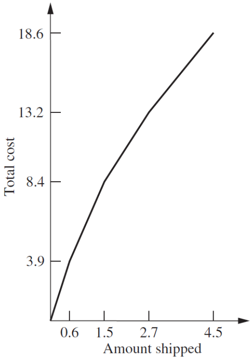

Profit $P(x) = xp(x) - cx$, where $x$ is the number of units sold, $p(x)$ price required to sell $x$ units, $c$ unit cost.

[1] FS Hillier and GJ Lieberman, Introduction to Operations Research 10th Ed., McGraw-Hill Education

**Example 2. Transportation Problem with Volume discounts on shipping costs**[2]

Nonlinearities arise from volume discounts: cost of shipping is less if the amount of goods to be shipped is larger.




[2] FS Hillier and GJ Lieberman, Introduction to Operations Research 10th Ed., McGraw-Hill Education

# UNIVARIATE UNCONSTRAINED OPTIMIZATION

# Univariate Unconstrained Optimization

**Problem.**

$$\max \text{ or } \min \ f(x)$$

over all possible values of $x$.

We can transform maximization problems to minimization by the following:

$$\max f(x) \Longleftrightarrow \min -f(x).$$

Hence, we consider minimization problems only.

# Univariate Unconstrained Optimization

*Recall*: Consider a function of a single variable $f(x)$, possessing up to second-order derivative.

**Necessary and sufficient condition for optimality**

If $f(x)$ is a convex function, then we have a minimum value at $x^*$ when

$$f'(x^*) = 0.$$

The optimization problem now is a root-finding problem for nonlinear equations.

**Problem:** finding one or more roots of the equation

$$g(x) = 0.$$

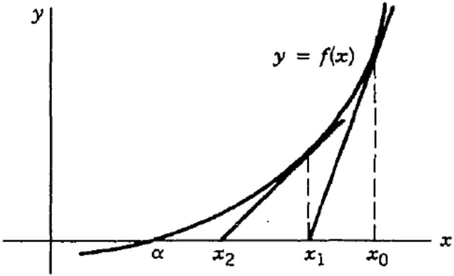
**Methods:**

1. Newton's method (for one variable)
2. Bisection method

**Newton's method for solving nonlinear equations:**

Solve $g(x) = 0$.

$$x_{n+1} = x_n - \frac{g(x_n)}{g'(x_n)}.$$




(Atkinson, 1989)

**Newton's method for optimization:**

Solve $f'(x) = 0$.

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)}.$$

# Newton's method for optimization

Initialization. Select tolerance `tol` and an initial trial solution $x'$ by inspection. Set $i = 1$.

Iteration $i$:

1. Calculate $f'(x_i)$ and $f''(x_i)$.
2. Set $x_{i+1} = x_i - \dfrac{f'(x_i)}{f''(x_i)}$.

*Stopping criteria*: If $|x_{i+1} - x_i| \leq \texttt{tol}$, stop. $x_{i+1}$ is approximately the optimal solution. Otherwise, reset $i = i + 1$ and perform another iteration.

# NEWTON'S METHOD FOR OPTIMIZATION

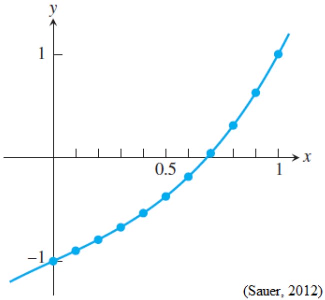
**Example:** Apply Newton's method to minimize (with `tol`$=1e-5$)

$$f(x) = -12x + 3x^4 + 2x^6.$$

# BISECTION METHOD FOR NONLINEAR EQUATIONS

The *bisection method* begins with an initial bracket and successively reduces its length until the solution has been isolated as accurately as desired.

For simplicity, we assume that the root in the interval is unique.

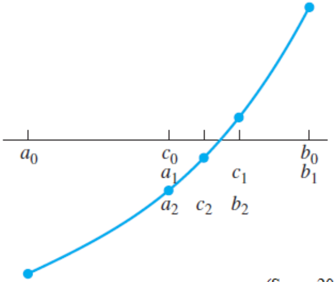



(Sauer, 2012)

# Bisection method for nonlinear equations

## Steps for solving $g(x) = 0$ using bisection method.

- Determine the midpoint $c$ of $[a, b]$, $c = \dfrac{a+b}{2}$.
- If $f(c) = 0$, then $r = c$. DONE.
- If $f(c) \neq 0$, then $f(c)$ has the same sign as either $f(a)$ or $f(b)$. If $sign(f(b))sign(f(c)) < 0$, then $a := c$; otherwise $b := c$.




(Sauer, 2012)

# Bisection method for optimization

Let $f(x)$ be a convex function, $x^*$ an optimal solution, and denote the following

$$
\begin{aligned}
x' &: \quad \text{current trial solution} \\
\underline{x} &: \quad \text{current lower bound on } x^* \\
\overline{x} &: \quad \text{current upper bound on } x^* \\
\varepsilon &: \quad \text{error tolerance for } x^*.
\end{aligned}
$$

# Bisection method for optimization

Initialization: Select `tol`. Find an initial $\underline{x}$ and $\overline{x}$. Then let the initial trial solution

$$x' = \frac{\underline{x} + \overline{x}}{2}.$$

Iteration:

1. Evaluate $f'(x')$.
2. If $f'(x') \geq 0$, reset $\underline{x} = x'$.
3. If $f'(x') \leq 0$, reset $\overline{x} = x'$.
4. Reset a new $x' = \dfrac{\underline{x} + \overline{x}}{2}$.

Stopping criteria: If $\overline{x} - \underline{x} \leq 2\texttt{tol}$, stop. Else, $i = i + 1$.

# Bisection method for optimization

**Example:** Apply bisection method to minimize (with `tol`$= 1e-5$)

$$f(x) = -12x + 3x^4 + 2x^6.$$

# Multivariate Unconstrained Optimization

# Multivariate Unconstrained Optimization

Problem:

$$\text{minimize } f(\mathbf{x})$$

over all values of $\mathbf{x} = (x_1, \ldots, x_n)$.

A *necessary* condition for $\mathbf{x} = \mathbf{x}^*$ to be optimal if $f$ is differentiable is

$$\frac{\partial f}{\partial x_j} = 0 \text{ at } \mathbf{x} = \mathbf{x}^* \text{ for } j = 1, \ldots, n.$$

If $f$ is a convex function, then the condition above is *sufficient* for $\mathbf{x} = \mathbf{x}^*$ to be optimal.

# Gradient-based methods

1. Steepest descent method
2. Newton's method for 2 or more variables

Given a function $f$ of $n$ variables $x_1, \ldots, x_n$, the **gradient** of $f$, denoted $\nabla f(x)$ is defined as

$$\nabla f(x) = \left\langle \frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \right\rangle.$$

Recall that this vector gives the direction of the maximum rate of change of $f$ at $x$, while the vector $-\nabla f(x)$ gives the direction of the minimum rate of change of $f$ at $x$.

# STEEPEST DESCENT METHOD

Initialization: Select ε and an initial trial solution $x'$ then go to the stopping criteria.

1. Express $f(\mathbf{x}' + t\nabla f(\mathbf{x}'))$ as a function of $t$ by setting

$$x_j = x_j' + t\left(\frac{\partial f}{\partial x_j}\right)_{\mathbf{x}=\mathbf{x}'} \quad \text{for } j = 1, 2, \ldots, n,$$

   and substituting these expressions into $f(\mathbf{x})$.
2. Use a search procedure for one-variable unconstrained optimization to find $t = t^*$ that minimizes $f(\mathbf{x}' + t\nabla f(\mathbf{x}'))$ over $t \geq 0$.

3. Reset $\mathbf{x}' = \mathbf{x}' + t^* \nabla f(\mathbf{x}')$. Go to stopping criteria.

   *Stopping criteria*: Evaluate $\nabla f(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}'$. Check if

$$\left| \frac{\partial f}{\partial x_j} \right| \leq \varepsilon \quad \text{for all } j = 1, 2, \ldots, n.$$

   If it is, stop and the optimal solution $\mathbf{x}^*$ is approximately the current $\mathbf{x}'$. Else, perform another iteration.

# Steepest descent method

**Example:** Use the steepest descent method to find the optimal solution of

$$\min f(\mathbf{x}) = -2x_1x_2 - 2x_2 + x_1^2 + 2x_2^2.$$

# NEWTON'S METHOD

If the objective function is concave, $\mathbf{x}$ and $\nabla f(\mathbf{x})$ are column vectors, the solution $\mathbf{x}'$ that maximizes the approximating quadratic function is of the form

$$\mathbf{x}' = \mathbf{x} - [\nabla^2 f(\mathbf{x})]^{-1} \nabla f(\mathbf{x}),$$

where $\nabla^2 f(\mathbf{x})$ is the $n \times n$ Hessian matrix and $[\nabla^2 f(\mathbf{x})]^{-1}$ is the inverse of this Hessian matrix.

# NEWTON'S METHOD

Let $J_f = \nabla^2 f(\mathbf{x})$. Alternatively, we have the following steps for Newton's method:

- Start with an initial guess $\mathbf{x}^0$.
- Solve the equation
$$J_f s = -\nabla f(\mathbf{x})$$
  for $s$ (Newton step).
- Update:
$$\mathbf{x}' = \mathbf{x} + s$$

# Hands-on Activity

- Apply the bisection and Newton's Method to find the minimum of
$$f(x) = x^6 + 3x^4 - 12x^3 + x^2 - x - 7.$$
- Write a code for solving optimization problems using Newton's method. Test your code using the previous example
$$\max f(x_1, x_2) = 2x_1x_2 + 2x_2 - x_1^2 - 2x_2^2.$$
- Apply steepest descent and Newton's Method to find the maximum of
$$f(x_1, x_2) = 1 - 2x_1 - x_1^2 + 4x_2 - 2x_2^2.$$