Optimization

1. Introduction

Optimization is the act of obtaining the best result under given circumstances.

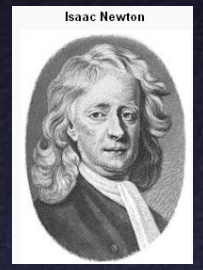
Optimization can be defined as the process of finding the conditions that give the maximum or minimum of a function.

The optimum seeking methods are also known as *mathematical programming techniques* and are generally studied as a part of operations research.

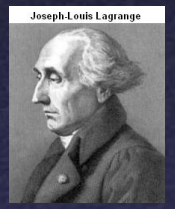
Operations research is a branch of mathematics concerned with the application of scientific methods and techniques to decision making problems and with establishing the best or optimal solutions.

1. Introduction

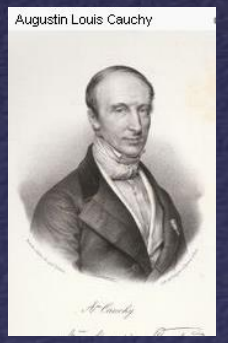
Historical development



- Isaac Newton (1642-1727)
  (The development of differential calculus methods of optimization)



- Joseph-Louis Lagrange (1736-1813)
  (Calculus of variations, minimization of functionals, method of optimization for constrained problems)



- Augustin-Louis Cauchy (1789-1857)
  (Solution by direct substitution, steepest descent method for unconstrained optimization)

1. Introduction

Historical development

- Leonhard Euler (1707-1783)
  (Calculus of variations, minimization of functionals)

- Gottfried Leibnitz (1646-1716)
  (Differential calculus methods of optimization)

İsim: Gottfired Wilhelm von Leibniz

1. Introduction

Historical development

- George Bernard Dantzig (1914-2005)

  (Linear programming and Simplex method (1947))

- Richard Bellman (1920-1984)

  (Principle of optimality in dynamic programming problems)

- Harold William Kuhn (1925-)

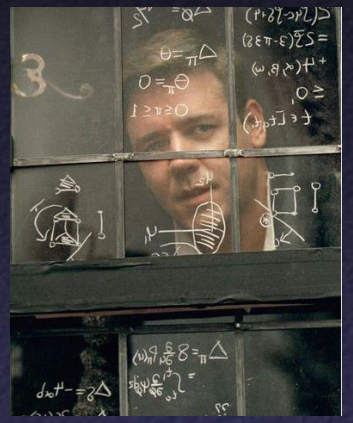
  (Necessary and sufficient conditions for the optimal solution of programming problems, game theory)

1. Introduction

Historical development

- Albert William Tucker (1905-1995)
  (Necessary and sufficient conditions
  for the optimal solution of programming
  problems, nonlinear programming, game
  theory: his PhD student
  was John Nash)

- Von Neumann (1903-1957)
  (game theory)



1. Introduction

- Objective function
- Variables
- Constraints

Find values of the variables
that minimize or maximize the objective function
while satisfying the constraints

1. Introduction

- **Mathematical optimization problem:**

$$\begin{aligned} &\text{minimize} && f_0(x) \\ &\text{subject to} && g_i(x) \le b_i, \quad i = 1, \ldots, m \end{aligned}$$

- $f_0 : \boldsymbol{R}^n \rightarrow \boldsymbol{R}$: objective function
- $x=(x_1, \ldots, x_n)$: design variables (unknowns of the problem, they must be linearly independent)
- $g_i : \boldsymbol{R}^n \rightarrow \boldsymbol{R}$: $(i=1, \ldots, m)$: inequality constraints
- The problem is a constrained optimization problem

1. Introduction

- If a point x^* corresponds to the minimum value of the function $f(x)$, the same point also corresponds to the maximum value of the negative of the function, $-f(x)$. Thus optimization can be taken to mean minimization since the maximum of a function can be found by seeking the minimum of the negative of the same function.

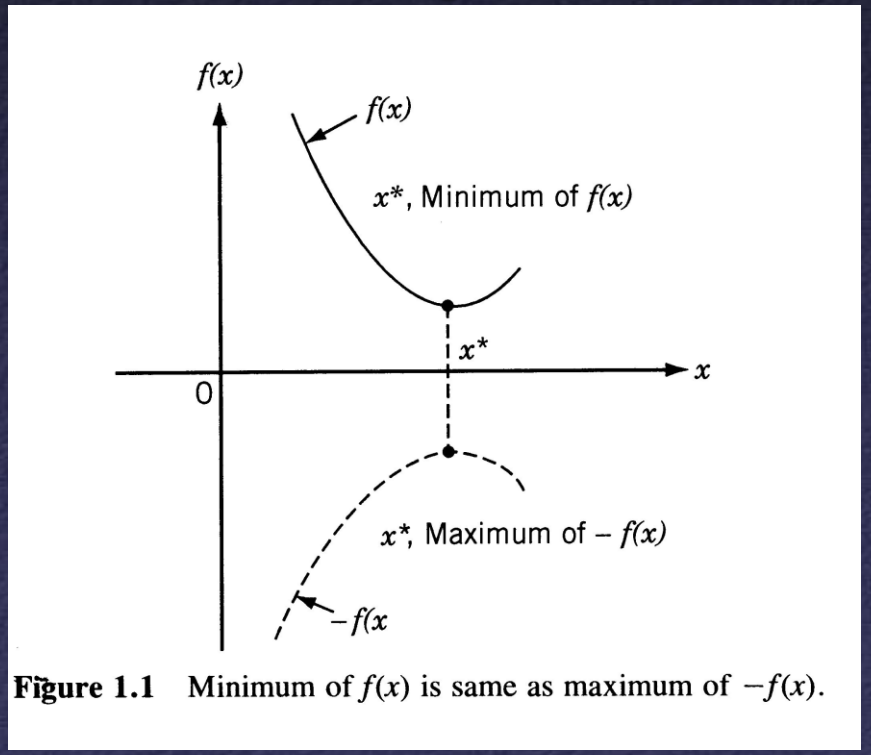


Figure 1.1 Minimum of $f(x)$ is same as maximum of $-f(x)$.

Examples

device sizing in electronic circuits

- variables: device widths and lengths
- constraints: manufacturing limits, timing requirements, maximum area
- objective: power consumption

data fitting

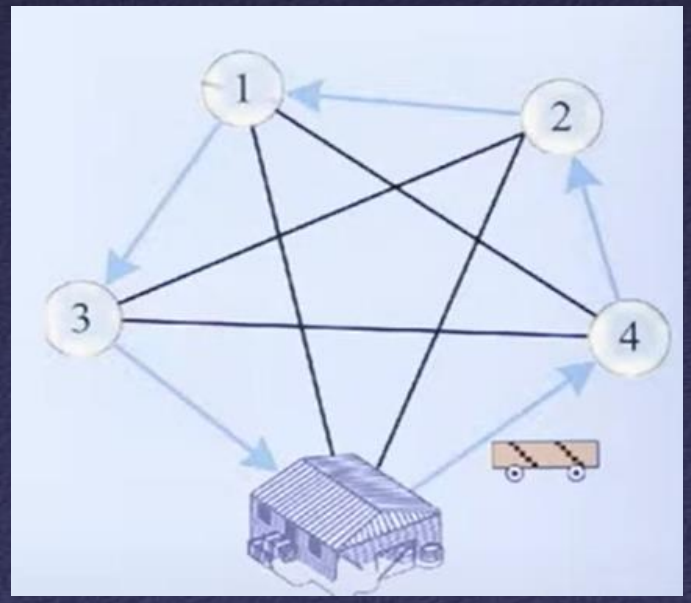
- variables: model parameters
- constraints: prior information, parameter limits
- objective: measure of misfit or prediction error, plus regularization term

Example

Transportation Problem - LP Formulation

Minimum Costs

Supply locations (Origins) → **Demand locations (Destinations)**

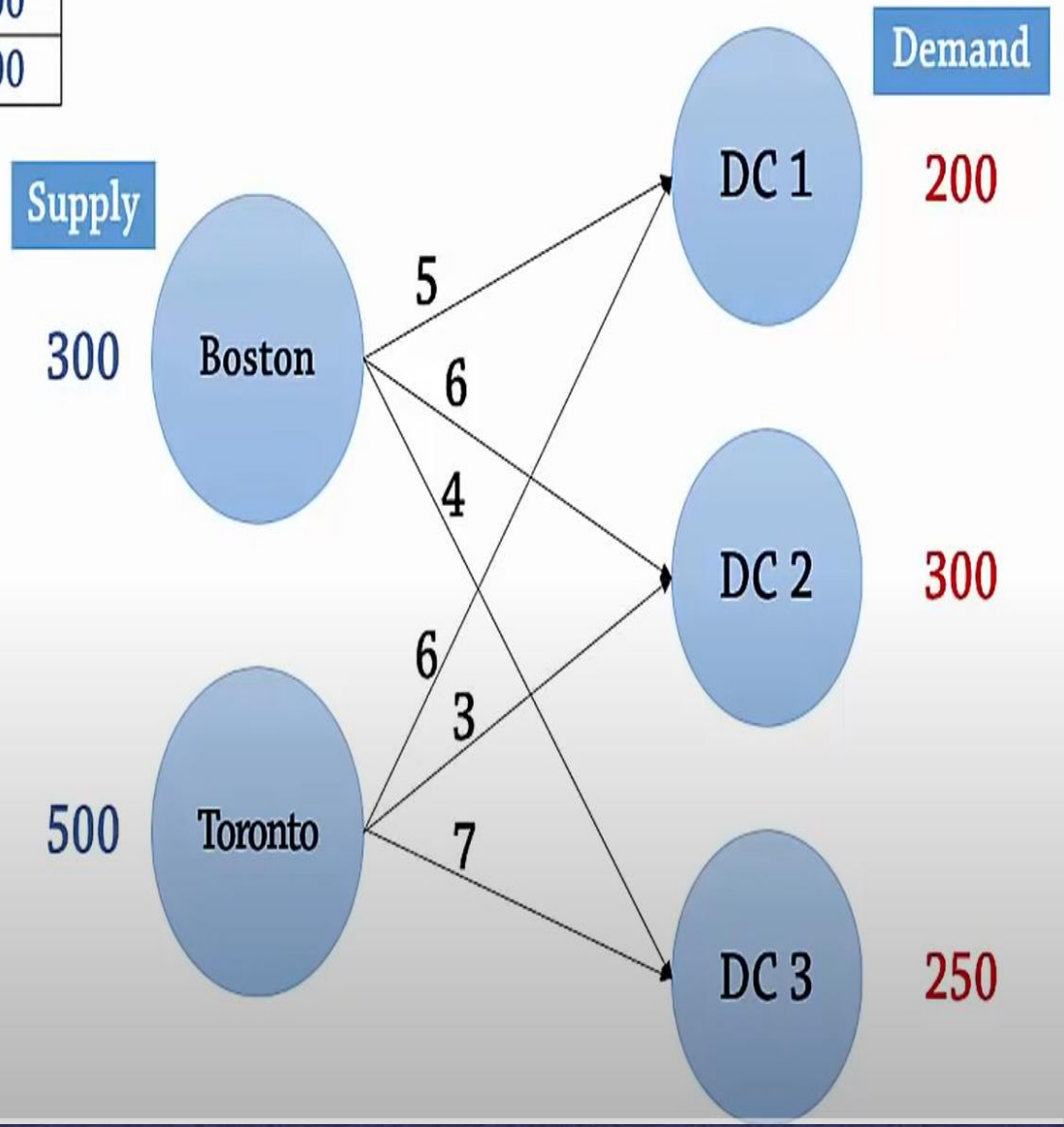


Example

Transportation Problem

From \ To	DC1	DC2	DC3	*Supply*
Boston	5	6	4	300
Toronto	6	3	7	500
Demand	200	300	250	

From \ To	DC1	DC2	DC3	Supply
Boston	5	6	4	300
Toronto	6	3	7	500
Demand	200	300	250	



Decision Variables:

X_{B1} = # units shipped from *Boston* to DC *1*

X_{B2} = # units shipped from *Boston* to DC 2

X_{B3} = # units shipped from *Boston* to DC 3

X_{T1} = # units shipped from *Toronto* to DC *1*

X_{T2} = # units shipped from *Toronto* to DC 2

X_{T3} = # units shipped from *Toronto* to DC 3

X_{ij} = # units shipped from Plant i to DC j

$i = B(\text{Boston}), T(\text{Toronto})$
$j = 1(\text{DC1}), 2(\text{DC2}), 3(\text{DC3})$

Objective Function

$$\textbf{Min}\ \ 5X_{B1} + 6X_{B2} + 4X_{B3} + 6X_{T1} + 3X_{T2} + 7X_{T3}$$

Subject to:

$$X_{B1} + X_{B2} + X_{B3} \leq 300 \quad \textit{(Boston's Supply)}$$
$$X_{T1} + X_{T2} + X_{T3} \leq 500 \quad \textit{(Toronto's Supply)}$$

$$X_{B1} + X_{T1} = 200 \quad \textit{(DC1's Demand)}$$
$$X_{B2} + X_{T2} = 300 \quad \textit{(DC2's Demand)}$$
$$X_{B3} + X_{T3} = 250 \quad \textit{(DC3's Demand)}$$

$$\textbf{Min}\ \ 5X_{B1} + 6X_{B2} + 4X_{B3} + 6X_{T1} + 3X_{T2} + 7X_{T3}$$

Subject to:

$$X_{B1} + X_{B2} + X_{B3} \leq 300 \quad \textit{(Boston's Supply)}$$

$$X_{T1} + X_{T2} + X_{T3} \leq 500 \quad \textit{(Toronto's Supply)}$$

$$X_{B1} + X_{T1} = 200 \quad \textit{(DC1's Demand)}$$

$$X_{B2} + X_{T2} = 300 \quad \textit{(DC2's Demand)}$$

$$X_{B3} + X_{T3} = 250 \quad \textit{(DC3's Demand)}$$

$$X_{B1}, X_{B2}, X_{B3}, X_{T1}, X_{T2}, X_{T3} \geq 0 \quad \textit{or} \quad X_{ij} \geq 0$$

Different Kinds of Optimization

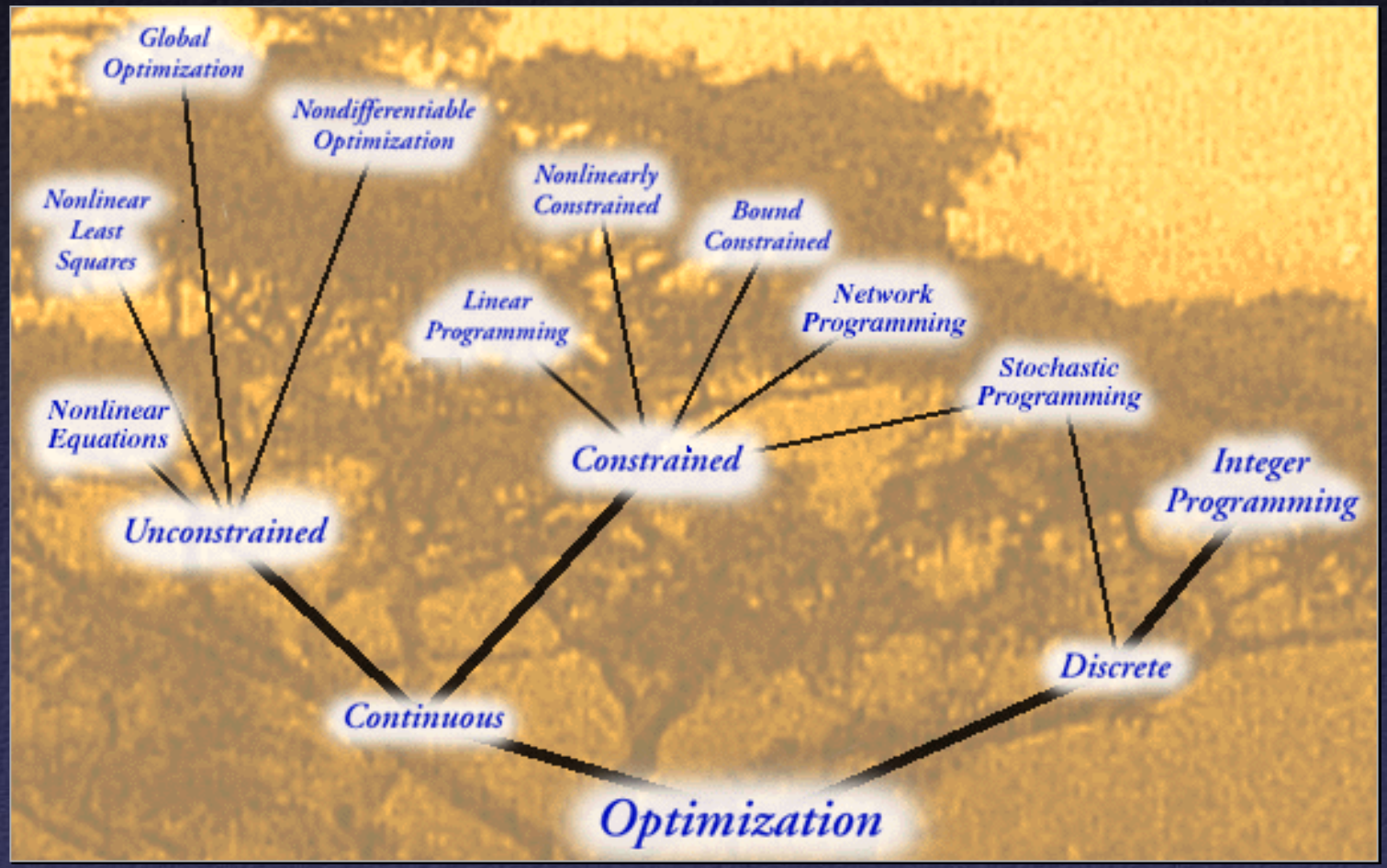


Figure from: Optimization Technology Center
http://www-fp.mcs.anl.gov/otc/Guide/OptWeb/

Solving Optimization Problems

General optimization problem

- Very difficult to solve
- Methods involve some compromise, e.g., very long computation time, or not always finding the solution (which may not matter in practice)

Exceptions: certain problem classes can be solved efficiently and reliably

- Linear programing problems
- Least-squares problems
- Convex optimization problems

Linear programing

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & a_i^T x \leq b_i, \quad i = 1, \ldots, m \end{array}$$

solving linear programs

- no analytical formula for solution
- reliable and efficient algorithms and software
- computation time proportional to $n^2 m$ if $m \geq n$; less with structure
- a mature technology

using linear programming

- not as easy to recognize as least-squares problems
- a few standard tricks used to convert problems into linear programs (*e.g.*, problems involving ℓ_1- or ℓ_∞-norms, piecewise-linear functions)

Least-squares problems

$$\text{minimize} \quad \|Ax - b\|_2^2$$

solving least-squares problems

- analytical solution: $x^\star = (A^T A)^{-1} A^T b$
- reliable and efficient algorithms and software
- computation time proportional to $n^2 k$ ($A \in \mathbf{R}^{k \times n}$); less if structured
- a mature technology

using least-squares

- least-squares problems are easy to recognize
- a few standard techniques increase flexibility ($e.g.$, including weights, adding regularization terms)

Convex optimization

Convex optimization problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m \end{array}$$

- objective and constraint functions are convex:

$$f_i(\alpha x + \beta y) \leq \alpha f_i(x) + \beta f_i(y)$$

  if $\alpha + \beta = 1$, $\alpha \geq 0$, $\beta \geq 0$

- includes least-squares problems and linear programs as special cases

solving convex optimization problems

- no analytical solution
- reliable and efficient algorithms
- computation time (roughly) proportional to $\max\{n^3, n^2m, F\}$, where F is cost of evaluating f_i's and their first and second derivatives
- almost a technology

using convex optimization

- often difficult to recognize
- many tricks for transforming problems into convex form
- surprisingly many problems can be solved via convex optimization

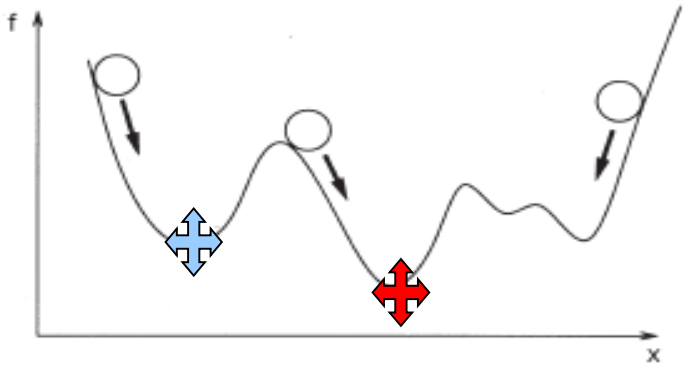
2. Classicial optimization techniques

Single variable optimization

- A function of one variable $f(x)$ has a relative or local minimum at $x = x^*$ if

$f(x^*) \leq f(x^*+h)$ for all sufficiently small positive and negative values of h

- A point x^* is called a relative or local maximum if $f(x^*) \geq f(x^*+h)$ for all values of h sufficiently close to zero.



Global minima

Local minima

2. Classicial optimization techniques

Single variable optimization

- A function $f(x)$ is said to have a global or absolute minimum at x^* if $f(x^*) \leq f(x)$ for all x, and not just for all x close to x^*, in the domain over which $f(x)$ is defined.

- Similarly, a point x^* will be a global maximum of $f(x)$ *if* $f(x^*) \geq f(x)$ for all x in the domain.

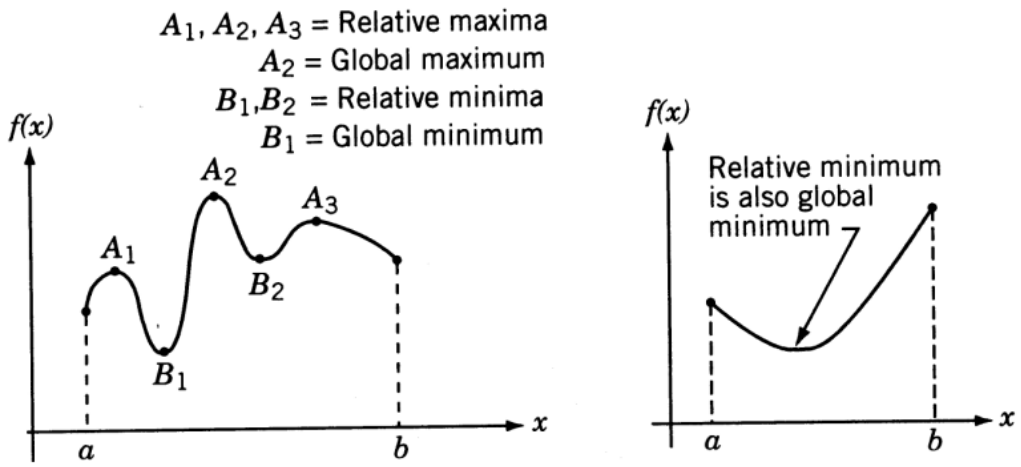
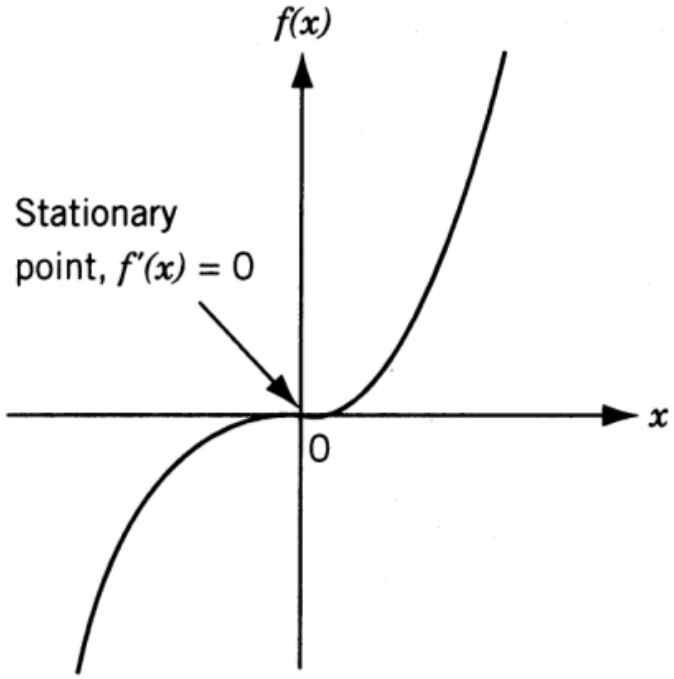


Figure 2.1 Relative and global minima.

Necessary condition

- If a function $f(x)$ is defined in the interval $a \leq x \leq b$ and has a relative minimum at $x = x^*$, where $a < x^* < b$, and if the derivative $df(x) / dx = f'(x)$ exists as a finite number at $x = x^*$, then $f'(x^*)=0$

- The theorem does not say that the function necessarily will have a minimum or maximum at every point where the derivative is zero. *e.g.* $f'(x)=0$ at $x= 0$ for the function shown in figure. However, this point is neither a minimum nor a maximum. In general, a point x^* at which $f'(x^*)=0$ is called a ***stationary point***.



Necessary condition

- The theorem does not say what happens if a minimum or a maximum occurs at a point x^* where the derivative fails to exist. For example, in the figure

$$\lim_{h\to 0}\frac{f(x^*+h)-f(x^*)}{h} = m^+ \text{ (positive) or } m^- \text{ (negative)}$$

depending on whether h approaches zero through positive or negative values, respectively. Unless the numbers m^+ or m^- are equal, the derivative $f'(x^*)$ does not exist.

If $f'(x^*)$ does not exist, the theorem is not applicable.

FIGURE 2.2 SAYFA 67

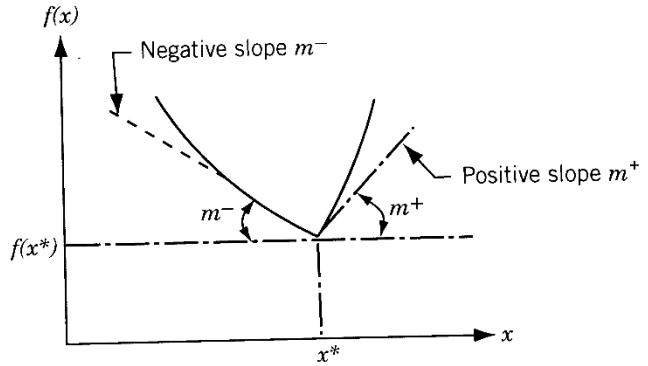


Figure 2.2 Derivative undefined at x^*.

Sufficient condition

- Let $f'(x^*)=f''(x^*)=\ldots=f^{(n-1)}(x^*)=0$, but $f^{(n)}(x^*) \neq 0$. Then $f(x^*)$ is
  - A **minimum** value of $f(x)$ if $f^{(n)}(x^*) > 0$ and n is **even**
  - A **maximum** value of $f(x)$ if $f^{(n)}(x^*) < 0$ and n is **even**
  - Neither a minimum nor a maximum if n is **odd**

Example

Determine the maximum and minimum values of the function:

$$f(x) = 12x^5 - 45x^4 + 40x^3 + 5$$

Solution: Since $f'(x)=60(x^4-3x^3+2x^2)=60x^2(x-1)(x-2)$,

$f'(x)=0$ at $x=0, x=1,$ and $x=2.$

The second derivative is: $f''(x)=60(4x^3-9x^2+4x)$

At $x=1$, $f''(x)=-60$ and hence $x=1$ is a relative maximum.

Therefore,

$$f_{max} = f(x=1) = 12$$

At $x=2$, $f''(x)=240$ and hence $x=2$ is a relative minimum.

Therefore,

$$f_{min} = f(x=2) = -11$$

At $x=0$, $f''(x)=0$ and hence we must investigate the next derivative.

$$f'''(x) = 60(12x^2 - 18x + 4) = 240 \text{ at } x = 0$$

Since $f'''(x) \neq 0$ at $x=0$, $x=0$ is neither a maximum nor a minimum, and it is an inflection point.

2. Classicial optimization techniques

Multivariable optimization with no constraints

- **Necessary condition**

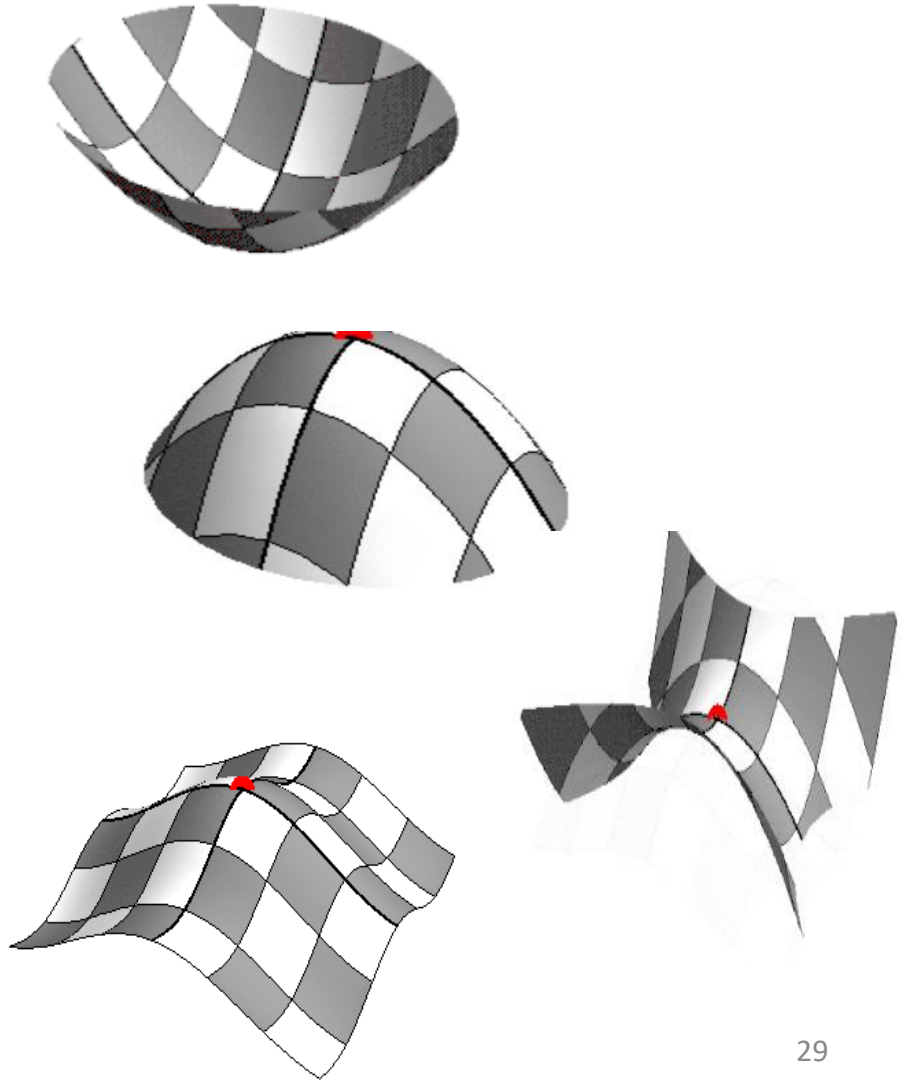
  If $f(\mathbf{X})$ has an extreme point (maximum or minimum) at $\mathbf{X}=\mathbf{X}^*$ and if the first partial derivatives of $f(\mathbf{X})$ exist at $\mathbf{X}^*$, then

$$\frac{\partial f}{\partial x_1}(\mathbf{X}^*) = \frac{\partial f}{\partial x_2}(\mathbf{X}^*) = \cdots = \frac{\partial f}{\partial x_n}(\mathbf{X}^*) = 0$$

- **Sufficient condition**

  A sufficient condition for a stationary point **X*** to be an extreme point is that the matrix of second partial derivatives (Hessian matrix) of $f(\mathbf{X}^*)$ evaluated at **X*** is

  - **Positive definite** when **X*** is a **relative minimum point**
  - **Negative definite** when **X*** is a **relative maximum point**



Where, the Hessian matrix is defined:

$$\mathbf{H}_f = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1\,\partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_1\,\partial x_n} \\ \dfrac{\partial^2 f}{\partial x_2\,\partial x_1} & \dfrac{\partial^2 f}{\partial x_2^2} & \cdots & \dfrac{\partial^2 f}{\partial x_2\,\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2 f}{\partial x_n\,\partial x_1} & \dfrac{\partial^2 f}{\partial x_n\,\partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_n^2} \end{bmatrix}.$$

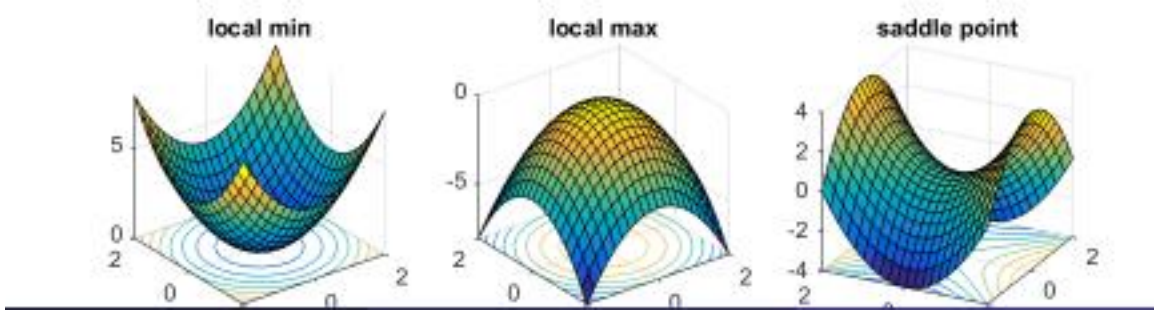
Note: Given a multivariable function *f*. We denote :

The **gradient** ∇f *: is the* **vector of its first partial derivatives**

$$\nabla f = \begin{pmatrix} \dfrac{\partial f}{\partial x_1} \\ \dfrac{\partial f}{\partial x_2} \\ \vdots \\ \dfrac{\partial f}{\partial x_n} \end{pmatrix}$$

And the **Hessian** $\nabla^2 f$ *: is the* **matrix of its second partial derivatives**

$$\nabla^2 f = \begin{pmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_1 \partial x_n} \\ \dfrac{\partial^2 f}{\partial x_2 \partial x_1} & \dfrac{\partial^2 f}{\partial x_2^2} & & \dfrac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & & \ddots & \vdots \\ \dfrac{\partial^2 f}{\partial x_n \partial x_1} & \dfrac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$



- We classify a stationary point of a function $f : \mathbb{R}^n \to \mathbb{R}$ as a global minimizer if the Hessian matrix of f is positive semidefinite **everywhere**,
- and as a global maximizer if the Hessian matrix is negative semidefinite everywhere.
- If the Hessian matrix is positive definite, or negative definite, the minimizer and maximizer (respectively) is strict.

Positive definite matrix (Review)

Definitions:

1) An *nxn* symmetric real matrix ***A*** is said to be **positive-definite if** $\boldsymbol{x^T Ax > 0}$ for all non-zero x$\in R^n$

2) An *nxn* symmetric real matrix ***A*** is said to be **positive-semidefinite if** $\boldsymbol{x^T Ax \geq 0}$ for all non-zero x$\in R^n$

3) An *nxn* symmetric real matrix ***A*** is said to be nega**tive-definite if** $\boldsymbol{x^T Ax < 0}$ for all non-zero x$\in R^n$

4) An *nxn* symmetric real matrix ***A*** is said to be nega**tive-semidefinite if** $\boldsymbol{x^T Ax \leq 0}$ for all non-zero x$\in R^n$

Example 1: $A = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$

$$x^T Ax = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$= \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 2x_1 \\ x_2 \end{bmatrix}$$

$$= 2x_1^2 + x_2^2$$

whenever $x_1 \neq 0$ or $x_2 \neq 0$ (hence x$\neq 0$), the matrix A is positive definite.

Example 2: $A = \begin{bmatrix} 9 & -15 \\ -15 & 25 \end{bmatrix}$

$$x^T Ax = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 9 & -15 \\ -15 & 25 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$= \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 9x_1 - 15x_2 \\ -15x_1 + 25x_2 \end{bmatrix}$$

$$= 9x_1^2 - 15x_1x_2 - 15x_1x_2 + 25x_2^2$$

$$= (3x_1 - 5x_2)^2$$

Then, $\boldsymbol{x^T Ax \geq 0}$ if x $\neq$0 and A is positive semi-definite. However, it is not positive definite because there exist non-zero vectors, for example the vector

$$x = \begin{bmatrix} 5 \\ 3 \end{bmatrix}$$

Positive definite matrix (Review)

Definitions:

1) An *nxn* symmetric real matrix ***A*** is said to be **positive-definite if** $\boldsymbol{x^T Ax > 0}$ for all non-zero x$\in R^n$

2) An *nxn* symmetric real matrix ***A*** is said to be **positive-semidefinite if** $\boldsymbol{x^T Ax \geq 0}$ for all non-zero x$\in R^n$

3) An *nxn* symmetric real matrix ***A*** is said to be nega**tive-definite if** $\boldsymbol{x^T Ax < 0}$ for all non-zero x$\in R^n$

4) An *nxn* symmetric real matrix ***A*** is said to be nega**tive-definite if** $\boldsymbol{x^T Ax \leq 0}$ for all non-zero x$\in R^n$

Theory:

Let A be *nxn* symmetric real matrix ***A.*** All eigenvalues of are real.

1) **A** is positive definite if and only if all of its eigenvalues are positive
2) A is positive semi-definite if and only if all of its eigenvalues are non-negative.
3) A is negative definite if and only if all of its eigenvalues are negative
4) A is negative semi-definite if and only if all of its eigenvalues are non-positive.
5) A is indefinite if and only if it has both positive and negative eigenvalues.

Review of mathematics

Positive definiteness

- **Test 1:** A matrix **A** will be positive definite if all its eigenvalues are positive; that is, all the values of λ that satisfy the determinental equation

$$\left|\mathbf{A} - \lambda \mathrm{I}\right| = 0$$

should be positive. Similarly, the matrix **A** will be negative definite if its eigenvalues are negative.

Review of mathematics

Negative definiteness

- Equivalently, a matrix is **negative-definite** if all its **eigenvalues** are **negative**
- It is **positive-semidefinite** if all its **eigenvalues are all greater than or equal to zero**
- It is **negative-semidefinite** if all its **eigenvalues are all less than or equal to zero**

Example 3: $A = \begin{pmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}$

The matrix is positive-definite.

Example 4: $A = \begin{pmatrix} 6 & 5 & 12 \\ 5 & 19 & 0 \\ 12 & 3 & 7 \end{pmatrix}$

The matrix is Not positive-definite.

Review of mathematics

Positive definiteness

- **Test 2:** Another test that can be used to find the positive definiteness of a matrix **A** of order *n* involves evaluation of the determinants

$$A = |a_{11}|$$

$$A_2 = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$$

$$A_3 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix}$$

$$A_3 = \begin{vmatrix} a_{11} & a_{12} & a_{13} \cdots a_{1n} \\ a_{21} & a_{22} & a_{23} \cdots a_{2n} \\ a_{31} & a_{32} & a_{33} \cdots a_{3n} \\ \vdots & & \\ a_{n1} & a_{n2} & a_{n3} \cdots a_{nn} \end{vmatrix}$$

- The matrix **A** will be **positive definite** if and only if all the values A_1, A_2, A_3,…A_n are positive
- The matrix **A** will be **negative definite** if and only if the sign of A_j is $(-1)^j$ for j=1,2,…,n
- If some of the A_j are positive and the remaining A_j are zero, the matrix **A** will be **positive semidefinite**

Example 5:

$$A = \begin{pmatrix} 6 & 5 & 12 \\ 5 & 19 & 0 \\ 12 & 3 & 7 \end{pmatrix}$$

A???

Example 6:

$$f(x, y, z) = x^2 + y^2 + z^2 - xy + yz - xz.$$

Find the extreme point of f

Example 7:

$$f(x, y, z) = e^{x-y} + e^{y-x} + e^{x^2} + z^2.$$

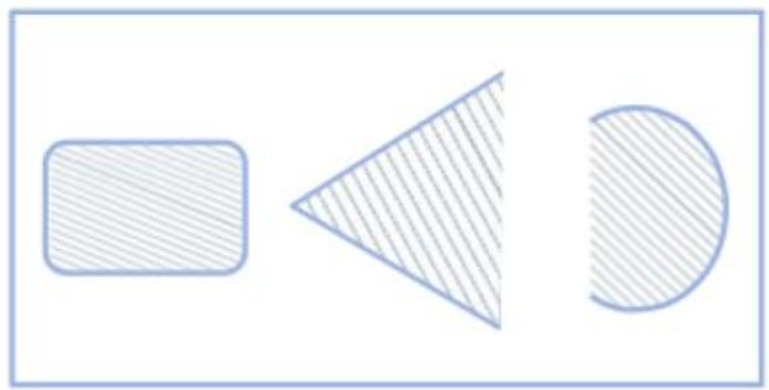
Find the extreme point of f

Convex set

- In **convex optimization**, the objective is a convex function defined over a convex set.
- In such problems, every local minimum is also a global minimum.
- Many models are designed so that their training objectives are convex.
- We say $\mathcal{S}$ is a **convex set** if, for any $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{S}$, we have

$$\lambda \boldsymbol{x} + (1 - \lambda)\boldsymbol{x}' \in \mathcal{S}, \forall \lambda \in [0, 1]$$

- If we draw a line from $\boldsymbol{x}$ to $\boldsymbol{x}'$, all points on the line lie inside the set.



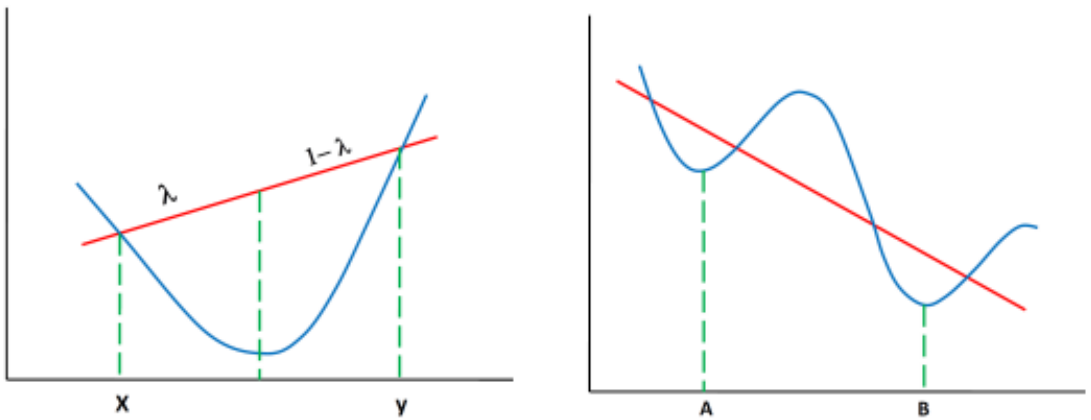
Convex

Not Convex

Convex functions

- $f(\boldsymbol{x})$ is called a convex function if it is defined on a convex set, and if, for any $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{S}$, and for any $0 \leq \lambda \leq 1$, we have:

$$f(\lambda \boldsymbol{x} + (1-\lambda)\boldsymbol{y}) \leq \lambda f(\boldsymbol{x}) + (1-\lambda) f(\boldsymbol{y})$$



- A function is **strictly convex** if the inequality is strict.
- A function is **concave** if $-f(\boldsymbol{x})$ is convex.
- A function can be neither convex nor concave.

Convex and concave functions

- Convex and concave functions in one variable

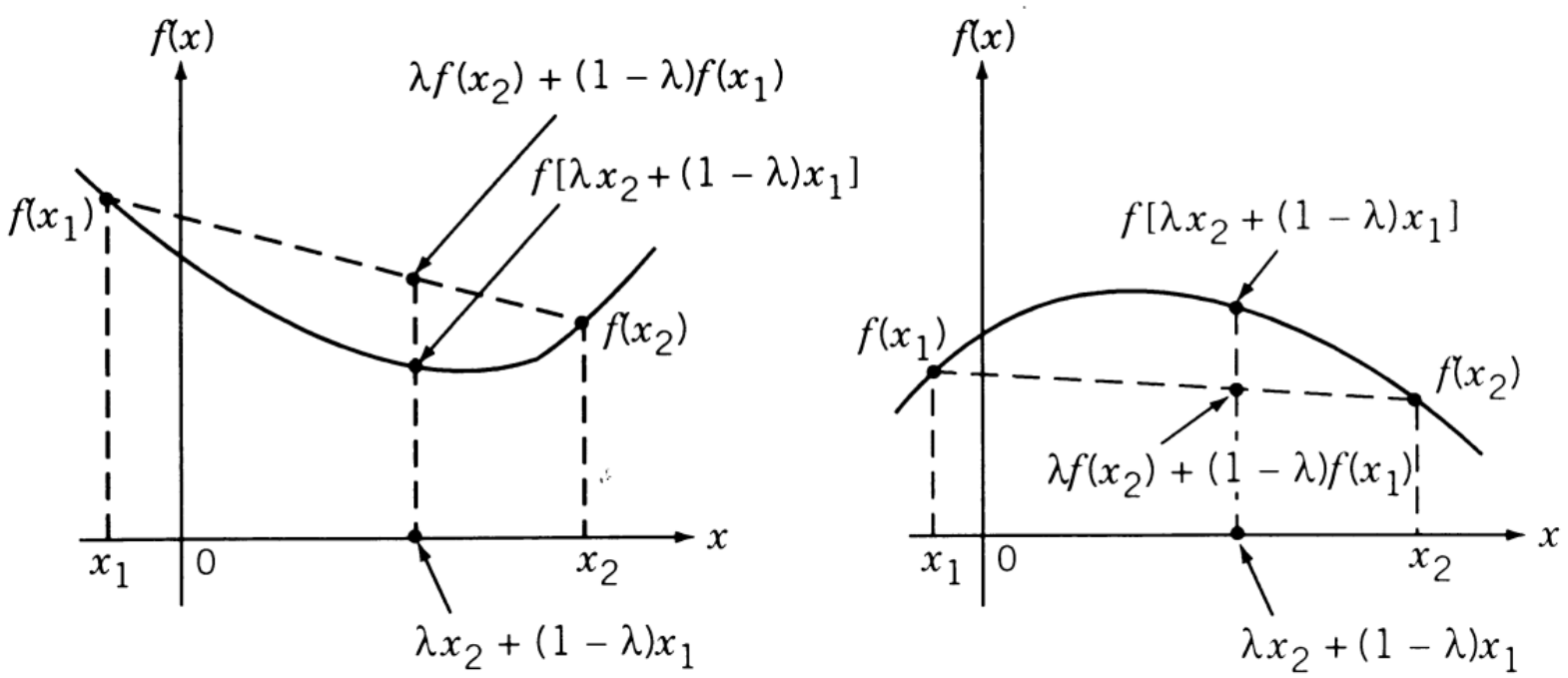


Figure A.1 Functions of one variable: (*a*) convex function in one variable; (*b*) concave function in one variable.

Concave function

- Convex and concave functions in two variables

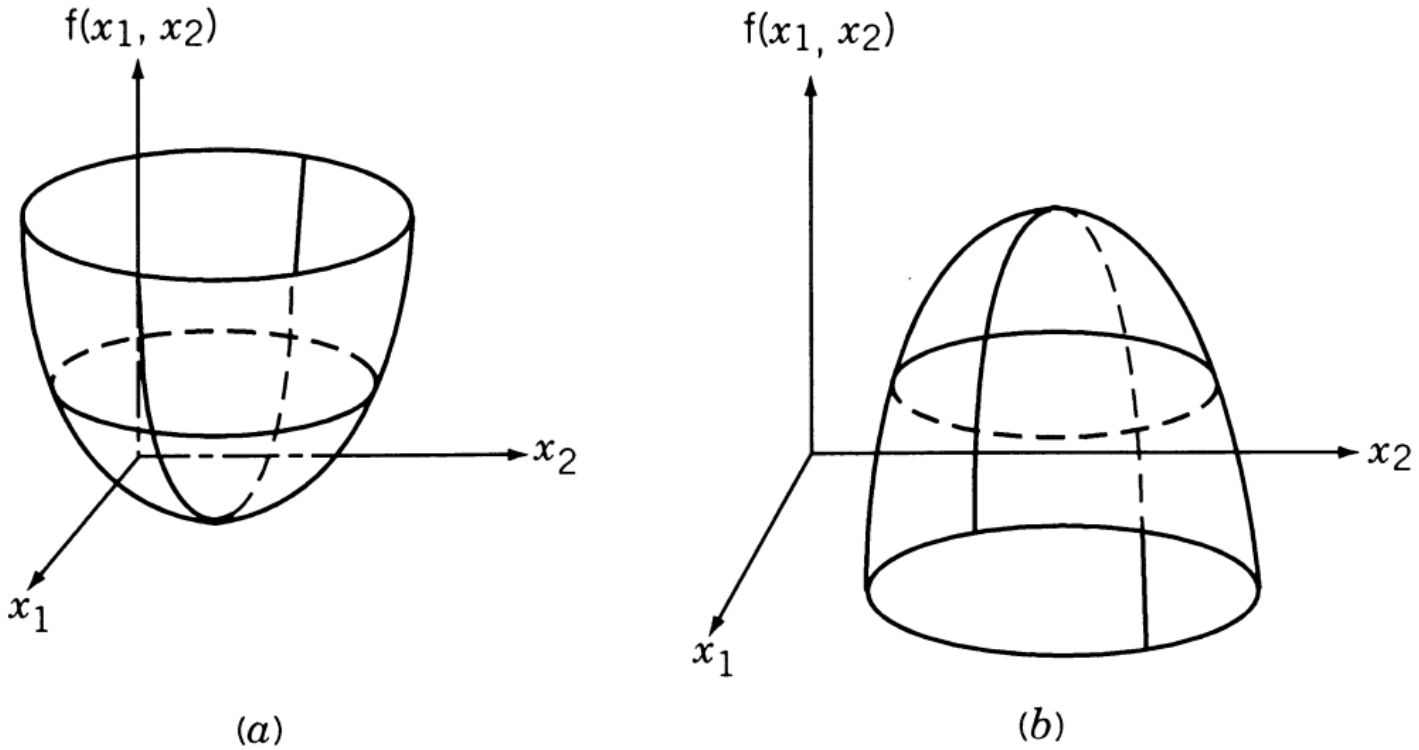


Figure A.2 Functions of two variables: (*a*) convex function in two variables; (*b*) concave function in two variables.

Examples of univariate convex functions

- e^{ax}
- $-\log(x)$
- x^a (defined on $\mathbb{R}_{++}$), $a \geq 1$ or $a \leq 0$
- $-x^a$ (defined on $\mathbb{R}_{++}$), $0 \leq a \leq 1$
- $|x|^a$, $a \geq 1$
- $x \log(x)$ (defined on $\mathbb{R}_{++}$)

Can you formally verify that these functions are convex?

First and second order characterizations of convex functions

Theorem 2. *Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is twice differentiable over an open domain. Then, the following are equivalent:*

(i) f *is convex.*

(ii) $f(y) \geq f(x) + \nabla f(x)^T (y - x)$, *for all* $x, y \in dom(f)$.

(iii) $\nabla^2 f(x) \succeq 0$, *for all* $x \in dom(f)$.

https://www.princeton.edu/~aaa/Public/Teaching/ORF523/S16/ORF523_S16_Lec7_gh.pdf

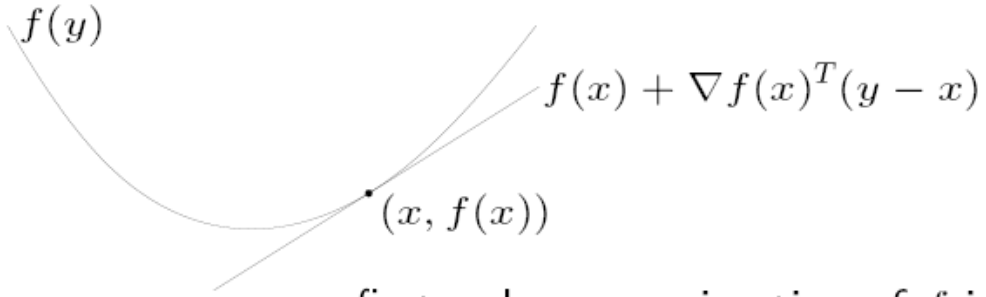
Theorem

Suppose $f : \mathbb{R}^n \to \mathbb{R}$ is twice differentiable over its domain. Then f is convex iff $\boldsymbol{H} = \nabla^2 f(\boldsymbol{x})$ is positive semi-definite for all $\boldsymbol{x} \in \mathbf{dom}(f)$. Furthermore, f is strictly convex if $\boldsymbol{H}$ is positive definite.

Convex functions

- A function *f*(x) is convex if for any two points x and **y**, we have

$$f(y) \geq f(x) + \nabla f^T(x)(y - x)$$



first-order approximation of f is global underestimator

- A function *f*(**X**) is convex if the Hessian matrix

$$\mathbf{H(X)} = \left[\partial^2 f(\mathbf{X}) / \partial x_i \partial x_j\right]$$

is positive semidefinite.

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

- Any local minimum of a convex function *f*(**X**) is a global minimum

- For example, consider the quadratic form

$$f(\boldsymbol{x}) = \boldsymbol{x}^\mathsf{T} \boldsymbol{A} \boldsymbol{x}$$

- This is convex if $\boldsymbol{A}$ is positive semi-definite.
- This is strictly convex if $\boldsymbol{A}$ is positive definite.
- It is neither convex nor concave if $\boldsymbol{A}$ has eigenvalues of mixed sign.
- Intuitively, a convex function is shaped like a bowl.

Corollary 1. *Consider an unconstrained optimization problem*

$$\min f(x)$$
$$s.t.\ x \in \mathbb{R}^n,$$

where f is convex and differentiable. Then, any point $\bar{x}$ that satisfies $\nabla f(\bar{x}) = 0$ is a global minimum.

Proof: From the first order characterization of convexity, we have

$$f(y) \geq f(x) + \nabla f^T(x)(y - x),\ \forall x, y$$

In particular,

$$f(y) \geq f(\bar{x}) + \nabla f^T(\bar{x})(y - x),\ \forall y.$$

Since $\nabla f(\bar{x}) = 0$, we get

$$f(y) \geq f(\bar{x}),\ \forall y.\ \square$$

Theorem 2. Consider an optimization problem

$$\min f(x)$$
$$s.t.\ x \in \Omega,$$

where $f : \mathbb{R}^n \to \mathbb{R}$ *is* strictly convex *on* Ω *and* Ω *is a convex set. Then the optimal solution (assuming it exists) must be unique.*

Proof: Suppose there were two optimal solutions $x, y \in \mathbb{R}^n$. This means that $x, y \in \Omega$ and

$$f(x) = f(y) \leq f(z), \forall z \in \Omega. \tag{6}$$

But consider $z = \frac{x+y}{2}$. By convexity of Ω, we have $z \in \Omega$. By strict convexity, we have

$$\begin{aligned} f(z) &= f\left(\frac{x+y}{2}\right) \\ &< \frac{1}{2}f(x) + \frac{1}{2}f(y) \\ &= \frac{1}{2}f(x) + \frac{1}{2}f(x) = f(x). \end{aligned}$$

But this contradicts (6). □

Example

Determine whether the following function is convex or concave.

$f(x_1,x_2) = 2x_1^3-6x_2^2$

Solution:

$$H(\mathbf{X}) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} \end{bmatrix} = \begin{bmatrix} 12x_1 & 0 \\ 0 & -12 \end{bmatrix}$$

Here

$$\frac{\partial^2 f}{\partial x_1^2} = 12x_1 \le 0 \text{ for } x_1 \le 0$$

$$\ge 0 \text{ for } x_1 \ge 0$$

$$\mathbf{H2} = -144x_1 \ge 0 \text{ for } x_1 \le 0$$

$$\le 0 \text{ for } x_1 \ge 0$$

Hence **H(X)** will be negative semidefinite and ***f*(X)** is concave for $x_1 \le 0$

Example

1) Determine whether the following function is convex or concave.
2) Find the global minimum/maximum of the given *f*

$$f(x_1, x_2, x_3) = 4x_1^2 + 3x_2^2 + 5x_3^2 + 6x_1x_2 + x_1x_3 - 3x_1 - 2x_2 + 15$$

Quadratic functions:

$$\text{Let } f(x) = x^T Ax + bx + c \text{ where } A \text{ is symmetric.}$$

Then *f* is convex if and only if A is positive semidefinite, independently of b, c.

Example $f(x_1, x_2, x_3) = 3x_1^2 + 3x_2^2 + 4x_3^2 + 4x_1x_2 + 2x_1x_3 + 2x_2x_3 - x_1 - 2x_2 - 3x_3$

1) Determine whether the following function is convex or concave.
2) Find the global minimum/maximum of the given *f*

Solution:

$$f(x_1, x_2, x_3) = x^T Ax + b^T x \qquad A = \begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 1 \\ 1 & 1 & 4 \end{pmatrix}; \qquad b = \begin{pmatrix} -1 \\ -2 \\ -3 \end{pmatrix};$$

then: $\nabla f(x_1, x_2, x_3) = 2Ax + b, \quad \nabla^2 f(x_1, x_2, x_3) = 2A.$

$$\nabla f(x_1, x_2, x_3) = (6x_1 + 4x_2 + 2x_3 - 1,\ 4x_1 + 6x_2 + 2x_3 - 2,\ 2x_1 + 2x_2 + 8x_3 - 3)$$

- $|A-\lambda I|=\begin{vmatrix} 3-\lambda & 2 & 1 \\ 2 & 3-\lambda & 1 \\ 1 & 1 & 4-\lambda \end{vmatrix}=(\lambda-1)(-\lambda^2+9\lambda-18)$

$|A-\lambda I|=0 \Leftrightarrow (\lambda-1)(-\lambda^2+9\lambda-18)=0 \Leftrightarrow \lambda=1;\ \lambda=3;\ \lambda=6$

Becase of all the eigen-values are positive, therefore A is possive definite.

$\Rightarrow$ *f*(**X**) is strictly convex function $\Rightarrow$ f has a unique global minimum,

$$\nabla f(x_1, x_2, x_3)=2Ax+b$$

$$\nabla f(x_1, x_2, x_3)=(6x_1+4x_2+2x_3-1,\ 4x_1+6x_2+2x_3-2,\ 2x_1+2x_2+8x_3-3)$$

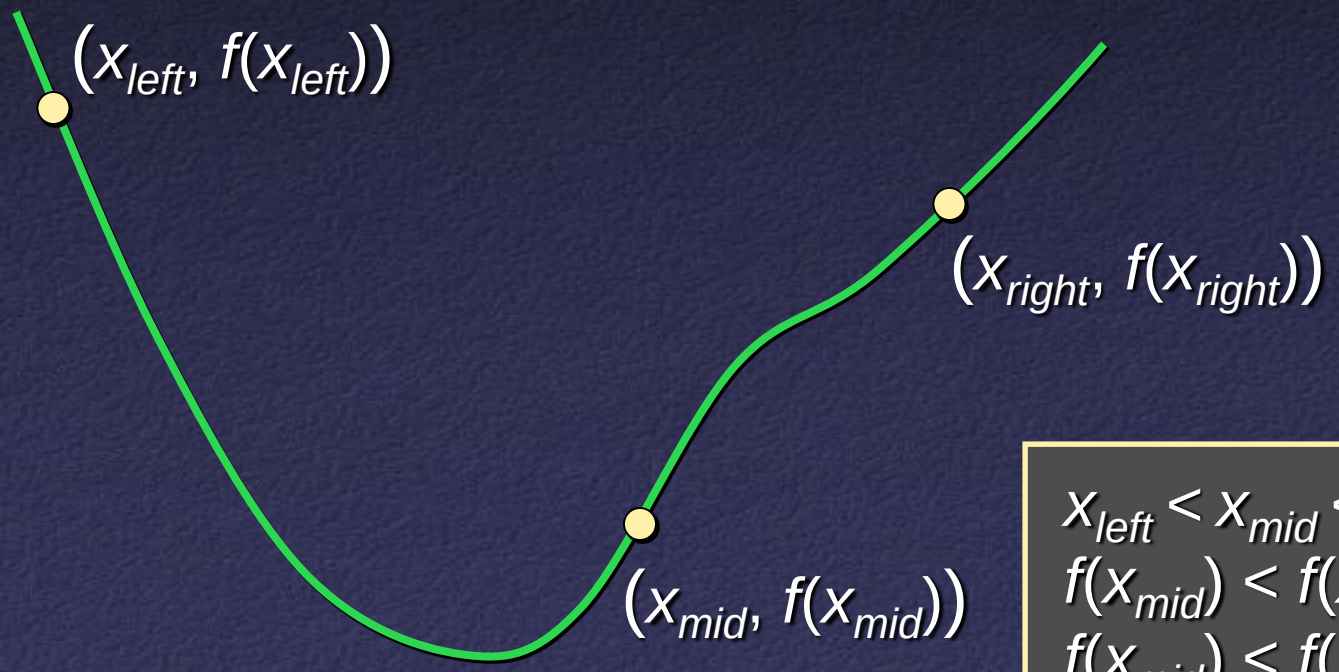
The global minimum point is the solution of equation system :

$$\nabla f(x_1, x_2, x_3)=0 \Leftrightarrow x_1=-\frac{1}{6},\ x_2=\frac{1}{3},\ x_3=\frac{1}{3}$$

Min *f* = $f(-\frac{1}{6}, \frac{1}{3}, \frac{1}{3})$

Optimization in 1-D

- Look for analogies to bracketing in root-finding
- What does it mean to *bracket* a minimum?



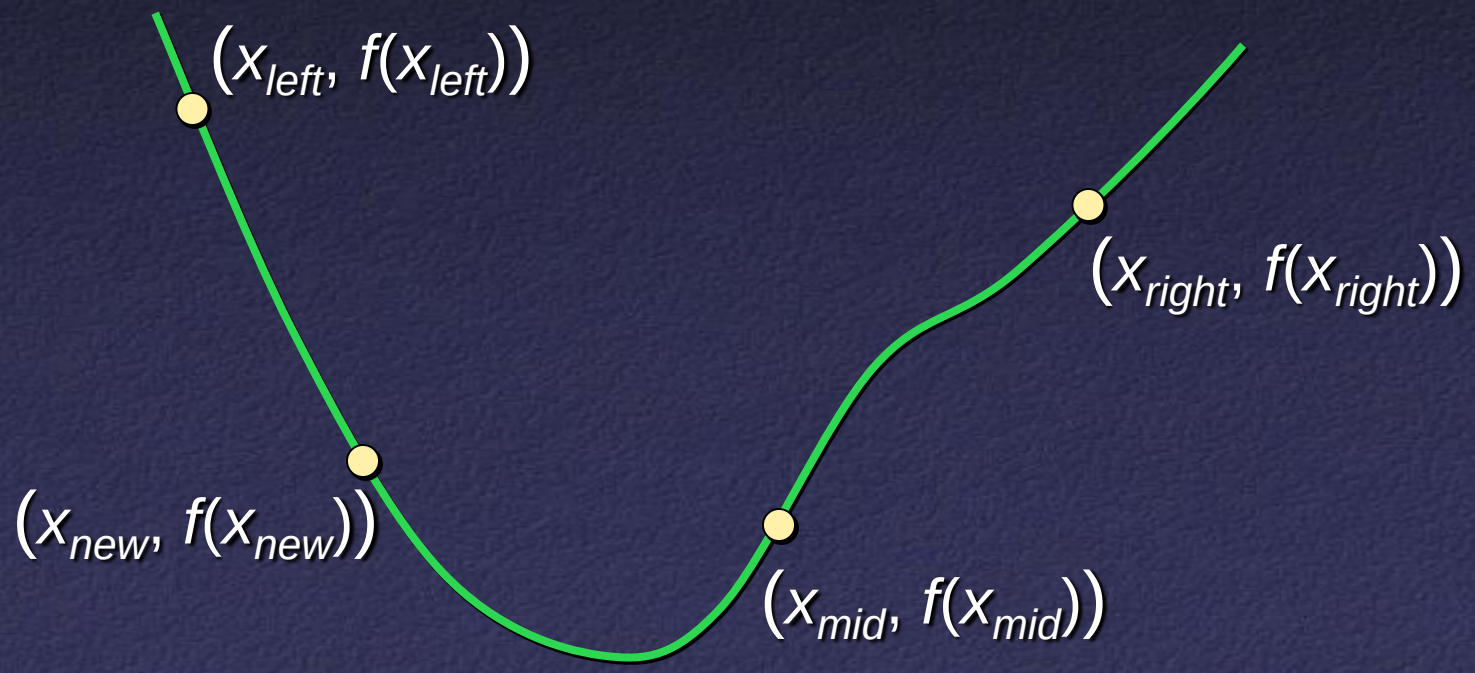
$$x_{left} < x_{mid} < x_{right}$$
$$f(x_{mid}) < f(x_{left})$$
$$f(x_{mid}) < f(x_{right})$$

Optimization in 1-D

- Once we have these properties, there is at least one local minimum between x_{left} and x_{right}
- Establishing bracket initially:
  - Given $x_{initial}$, *increment*
  - Evaluate $f(x_{initial})$, $f(x_{initial}+increment)$
  - If decreasing, step until find an increase
  - Else, step in opposite direction until find an increase
  - Grow increment at each step
- For maximization: substitute $-f$ for f

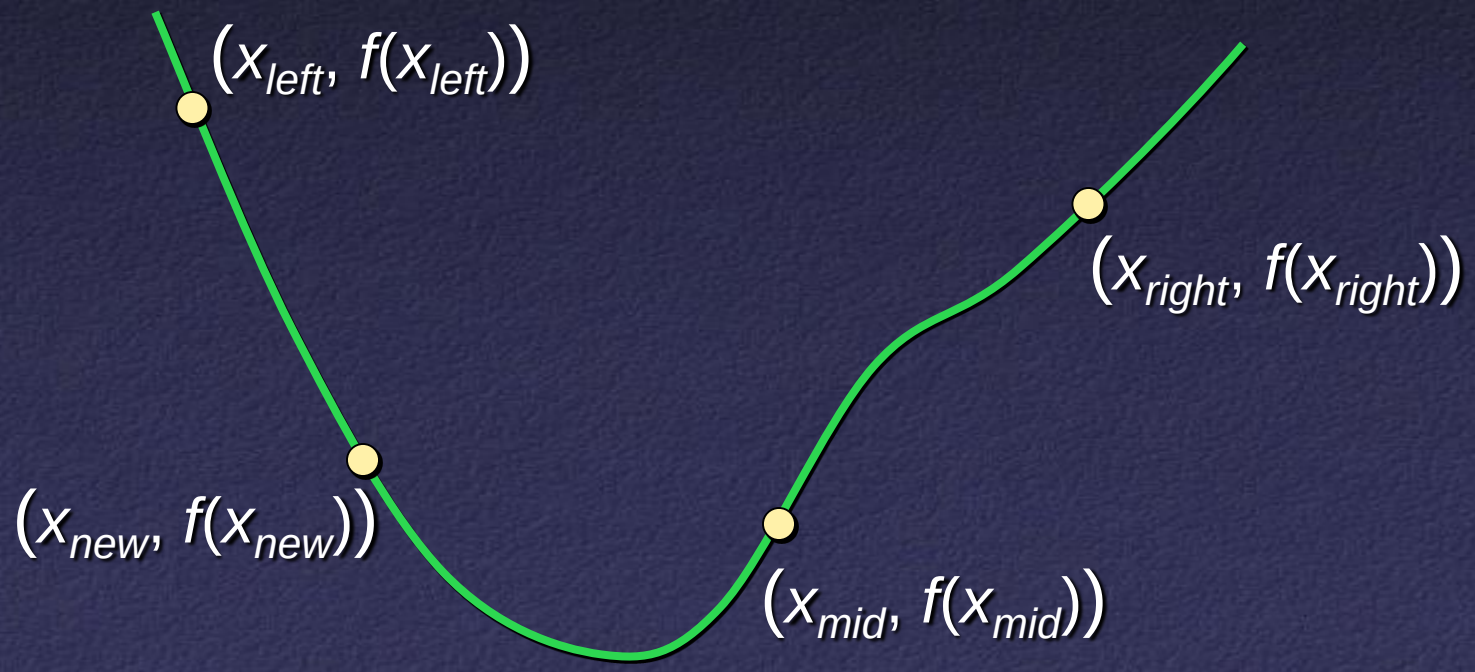
Optimization in 1-D

- Strategy: evaluate function at some x_{new}



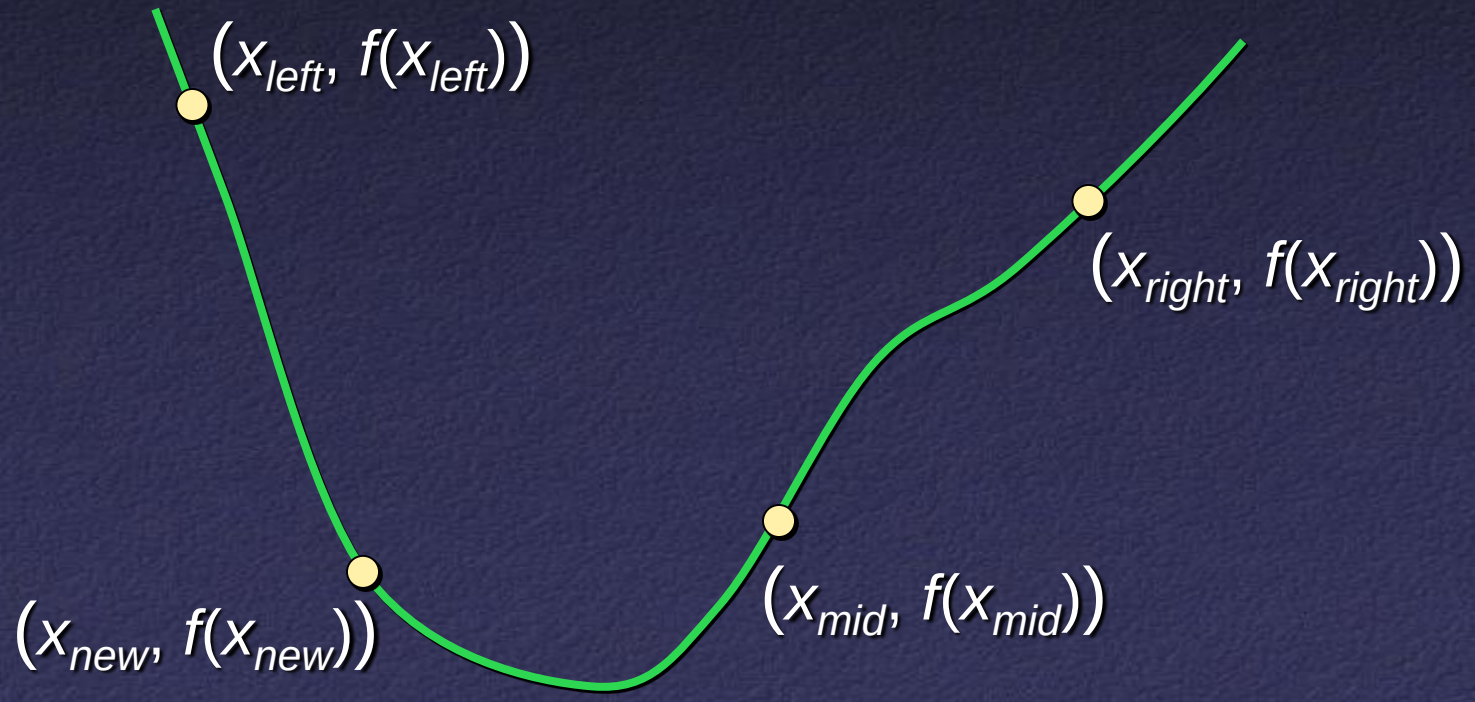
Optimization in 1-D

- Strategy: evaluate function at some x_{new}
  - Here, new "bracket" points are x_{new}, x_{mid}, x_{right}



Optimization in 1-D

- Strategy: evaluate function at some x_{new}
  - Here, new "bracket" points are x_{left}, x_{new}, x_{mid}



Optimization in 1-D

- Unlike with root-finding, can't always guarantee that interval will be reduced by a factor of 2
- Let's find the optimal place for x_{mid}, relative to left and right, that will guarantee same factor of reduction regardless of outcome

First-order methods

- We consider **iterative** optimization methods that leverage **first order** derivatives of the objective function.
- They compute which directions point "downhill", but ignore curvature information.
- All these algorithms require the user specify a starting point $\boldsymbol{\theta}_0$.
- At each iteration t, an update is performed

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \rho_t \boldsymbol{d}_t$$

  where ρ_t is the **step size** or **learning rate**, and $\boldsymbol{d}_t$ is a **descent direction**, e.g, the negative of the **gradient** given by $\boldsymbol{g}_t = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})|_{\boldsymbol{\theta}_t}$.
- The update steps are continued until a **stationary point** is reached, where the gradient is zero.

Descent direction

- A direction $\boldsymbol{d}$ is a **descent direction** if there is a small enough (but nonzero) amount ρ that we can move in direction $\boldsymbol{d}$ and be guaranteed to decrease the function value.
- We require there exists an $\rho_{max} > 0$ such that

$$\mathcal{L}(\boldsymbol{\theta} + \rho \boldsymbol{d}) < \mathcal{L}(\boldsymbol{\theta})$$

  for all $0 < \rho < \rho_{max}$.
- The gradient at the current iterate,

$$\boldsymbol{g}_t \triangleq \nabla \mathcal{L}(\boldsymbol{\theta})|_{\boldsymbol{\theta}_t} = \nabla \mathcal{L}(\boldsymbol{\theta}_t) = \boldsymbol{g}(\boldsymbol{\theta}_t)$$

  points in the direction of maximal increase in f, so the negative gradient is a descent direction.

Descent direction

- Any direction $\boldsymbol{d}$ is also a descent direction if the angle θ between $\boldsymbol{d}$ and $-\boldsymbol{g}_t$ is less than 90 degrees and satisfies

$$\boldsymbol{d}^{\mathsf{T}}\boldsymbol{g}_t = \|\boldsymbol{d}\|\|\boldsymbol{g}_t\|\cos(\theta) < 0$$

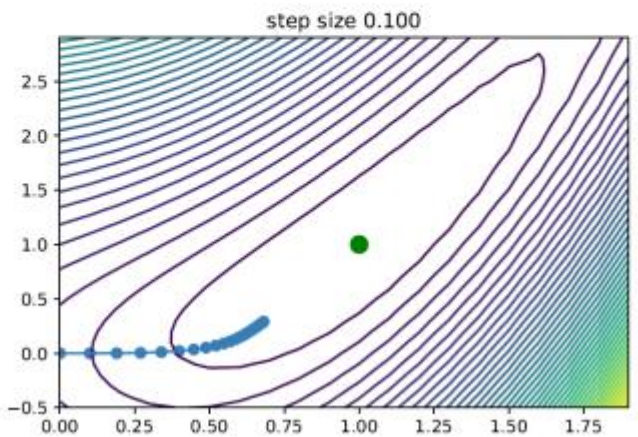
- The best choice would be to pick $\boldsymbol{d}_t = -\boldsymbol{g}_t$.
- This is the direction of **steepest descent**.

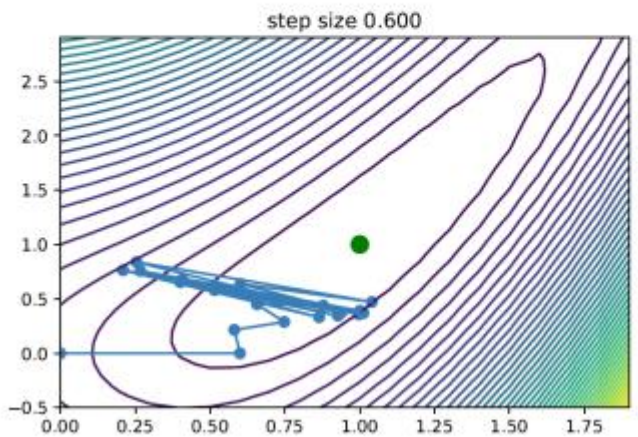
Step size (learning rate)

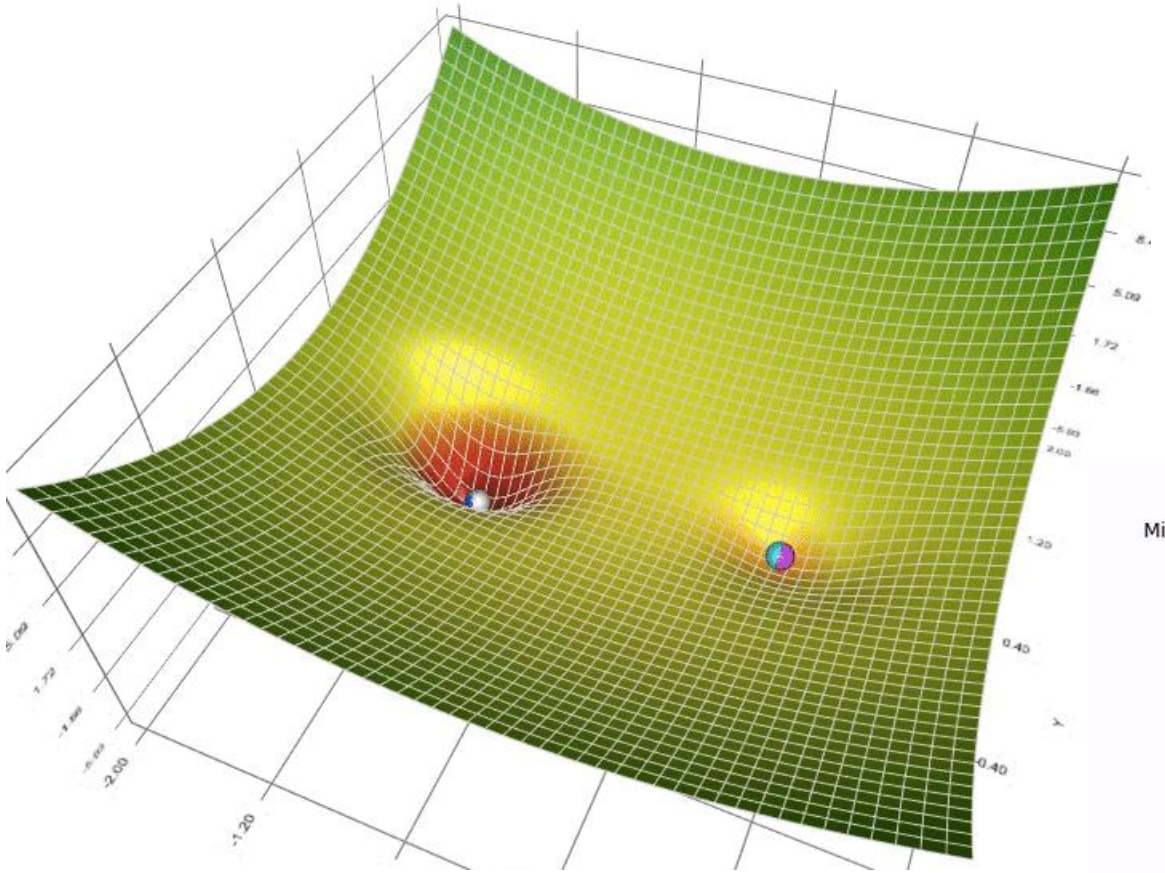
- The sequence of step sizes $\{\rho_t\}$ is called the **learning rate schedule**.
- The simplest method is to use constant step size, $\rho_t = \rho$.
- However, if it is too large, the method may fail to converge. If it is too small, the function will converge but very slowly.
- Example:

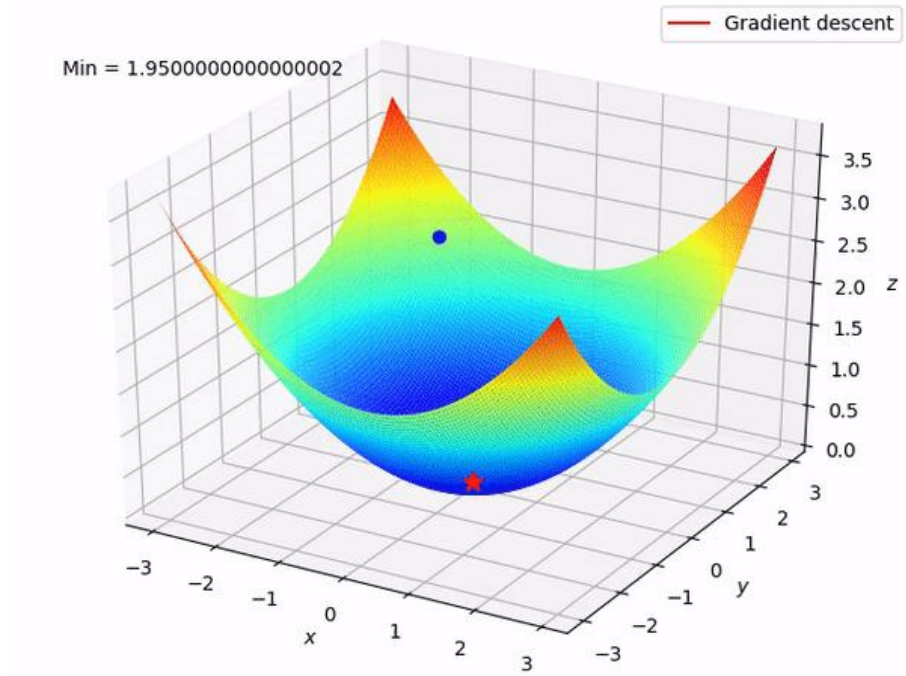
$$\mathcal{L}(\boldsymbol{\theta}) = 0.5(\theta_1^2 - \theta_2)^2 + 0.5(\theta_1 - 1)^2$$

- Pick our descent direction $\boldsymbol{d}_t = -\boldsymbol{g}_t$. Consider $\rho_t = 0.1$ vs $\rho_t = 0.6$:

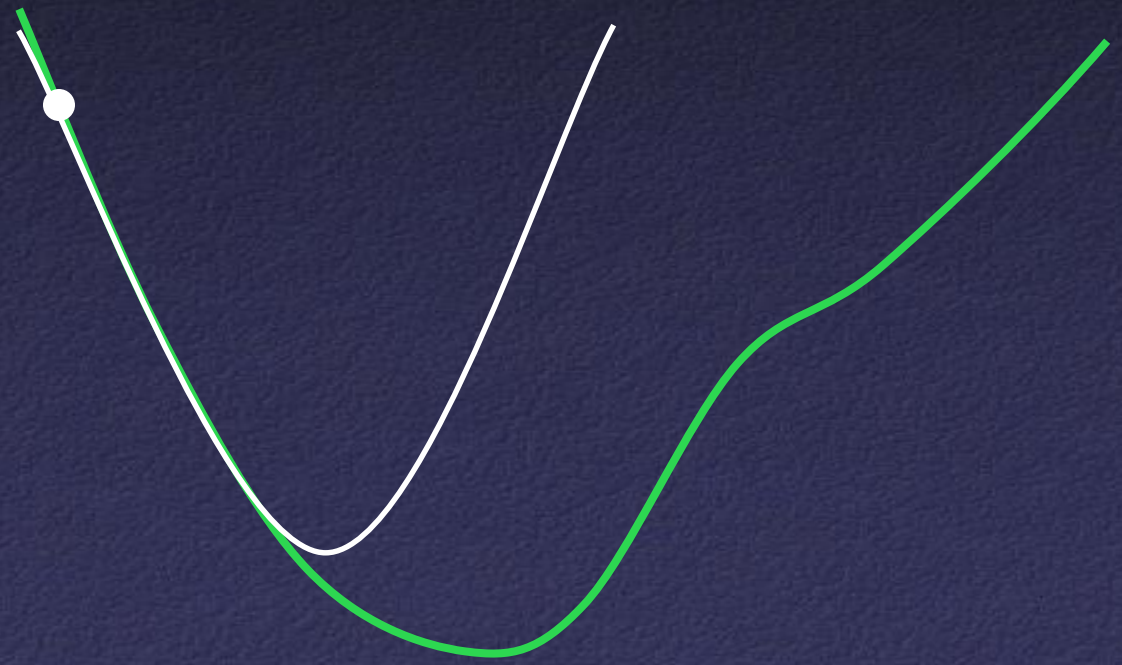







Gradient descent
Min = 1.9500000000000002
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
z
−3
−2
−1
0
1
2
3
x
y

Newton's Method



Newton's Method

Newton's Method

Newton's Method

Newton's Method

- At each step:

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

- Requires 1st and 2nd derivatives
- Quadratic convergence

Multi-Dimensional Optimization

- Important in many areas
  - Fitting a model to measured data
  - Finding best design in some parameter space
- Hard in general
  - Weird shapes: multiple extrema, saddles, curved or elongated valleys, etc.
  - Can’t bracket
- In general, easier than rootfinding
  - Can always walk “downhill”

Newton's Method in Multiple Dimensions

- Replace 1st derivative with gradient, 2nd derivative with Hessian

$$f(x, y)$$

$$\nabla f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix}$$

$$H = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix}$$

Newton's Method in Multiple Dimensions

- Replace 1st derivative with gradient,
  2nd derivative with Hessian
- So,

$$\vec{x}_{k+1} = \vec{x}_k - H^{-1}(\vec{x}_k)\nabla f(\vec{x}_k)$$

- Tends to be extremely fragile unless function very smooth and starting close to minimum

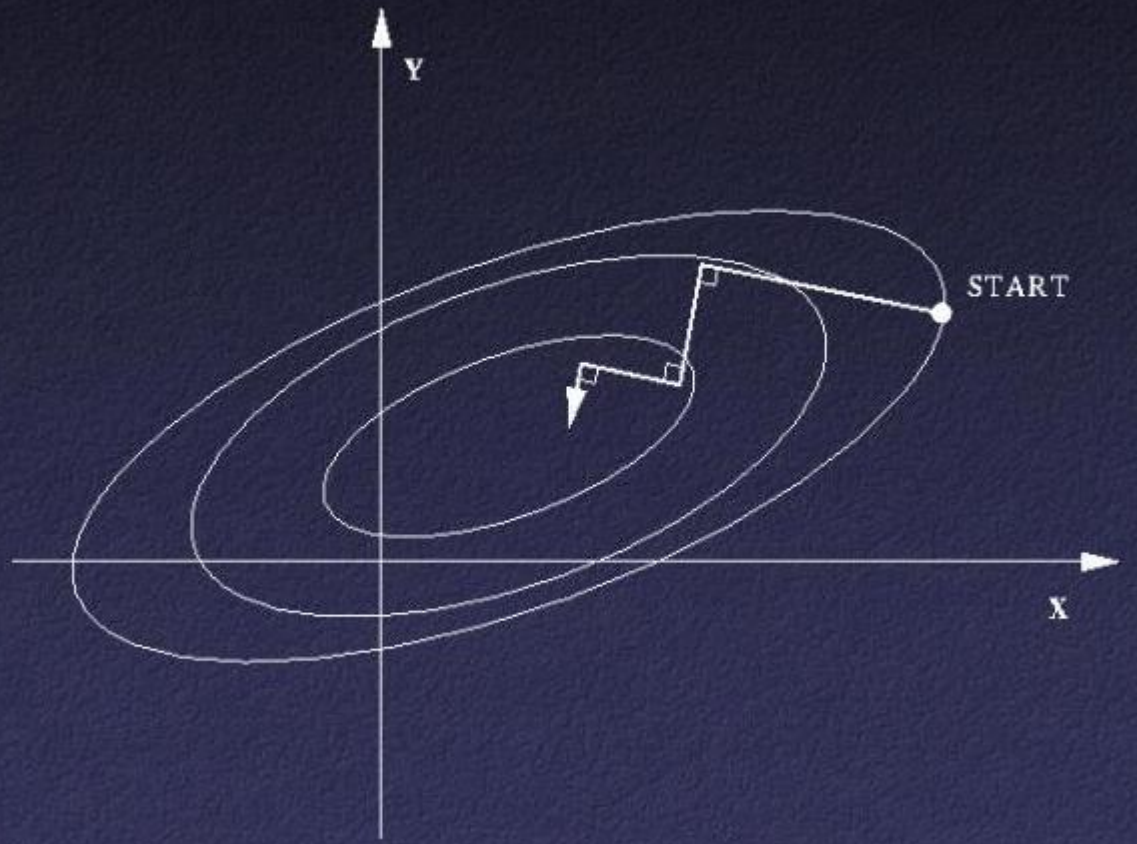
Important classification of methods

- Use function + gradient + Hessian (Newton)
- Use function + gradient (most descent methods)
- Use function values only (Nelder-Mead, called also “simplex”, or “amoeba” method)

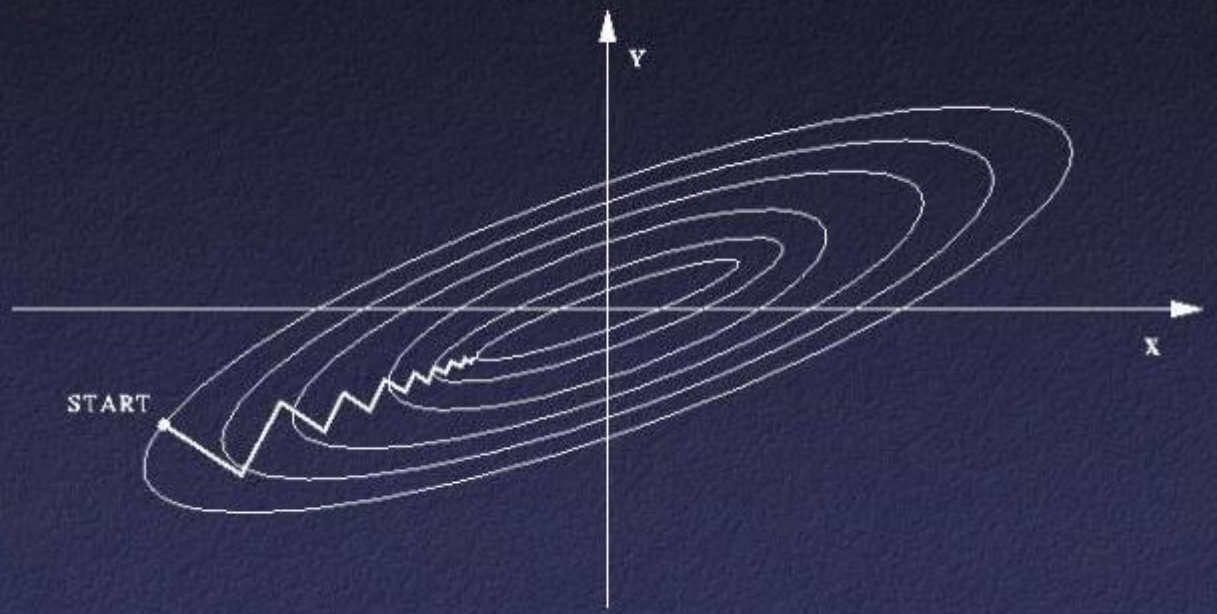
Steepest Descent Methods

- What if you can't / don't want to use 2^{nd} derivative?
- "Quasi-Newton" methods estimate Hessian
- Alternative: walk along (negative of) gradient…
  - Perform 1-D minimization along line passing through current point in the direction of the gradient
  - Once done, re-compute gradient, iterate

Problem With Steepest Descent



Problem With Steepest Descent



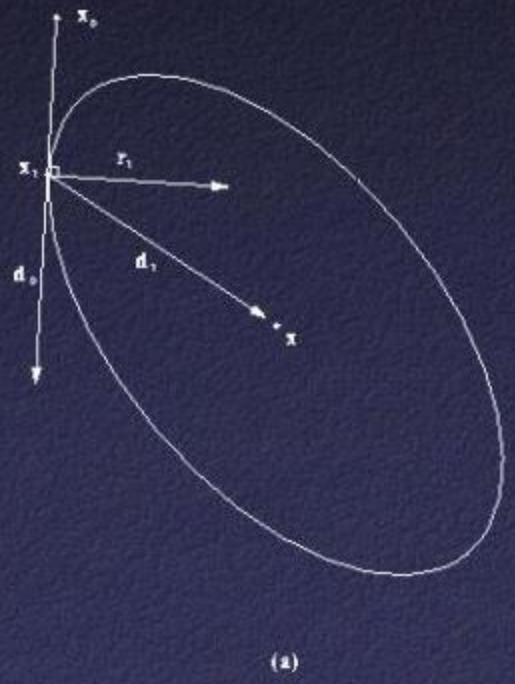
Conjugate Gradient Methods

- Idea: avoid “undoing” minimization that’s already been done
- Walk along direction

$$d_{k+1} = -g_{k+1} + \beta_k d_k$$

- Polak and Ribiere formula:

$$\beta_k = \frac{g_{k+1}^{\mathrm{T}} (g_{k+1} - g_k)}{g_k^{\mathrm{T}} g_k}$$



Conjugate Gradient Methods

- Conjugate gradient implicitly obtains information about Hessian
- For quadratic function in *n* dimensions, gets *exact* solution in *n* steps (ignoring roundoff error)
- Works well in practice…

Value-Only Methods in Multi-Dimensions

- If can't evaluate gradients, life is hard
- Can use approximate (numerically evaluated) gradients:

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f}{\partial e_1} \\ \frac{\partial f}{\partial e_2} \\ \frac{\partial f}{\partial e_3} \\ \vdots \end{pmatrix} \approx \begin{pmatrix} \frac{f(x+\delta \cdot e_1) - f(x)}{\delta} \\ \frac{f(x+\delta \cdot e_2) - f(x)}{\delta} \\ \frac{f(x+\delta \cdot e_3) - f(x)}{\delta} \\ \vdots \end{pmatrix}$$

Generic Optimization Strategies

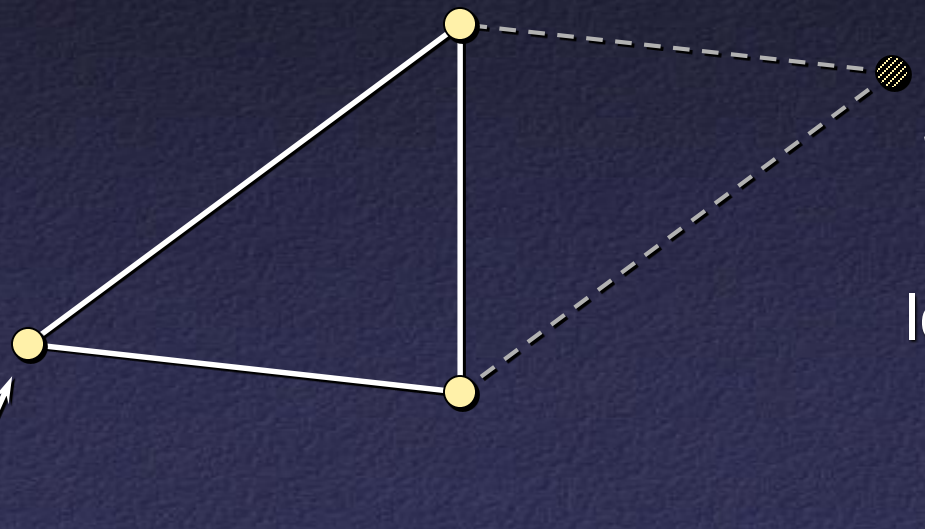
- Uniform sampling:
  - Cost rises exponentially with # of dimensions
- Simulated annealing:
  - Search in random directions
  - Start with large steps, gradually decrease
  - “Annealing schedule” – how fast to cool?

Downhill Simplex Method (Nelder-Mead)

- Keep track of n+1 points in n dimensions
  - Vertices of a *simplex* (triangle in 2D tetrahedron in 3D, etc.)
- At each iteration: simplex can move, expand, or contract
  - Sometimes known as *amoeba method*: simplex "oozes" along the function

Downhill Simplex Method (Nelder-Mead)

- Basic operation: reflection

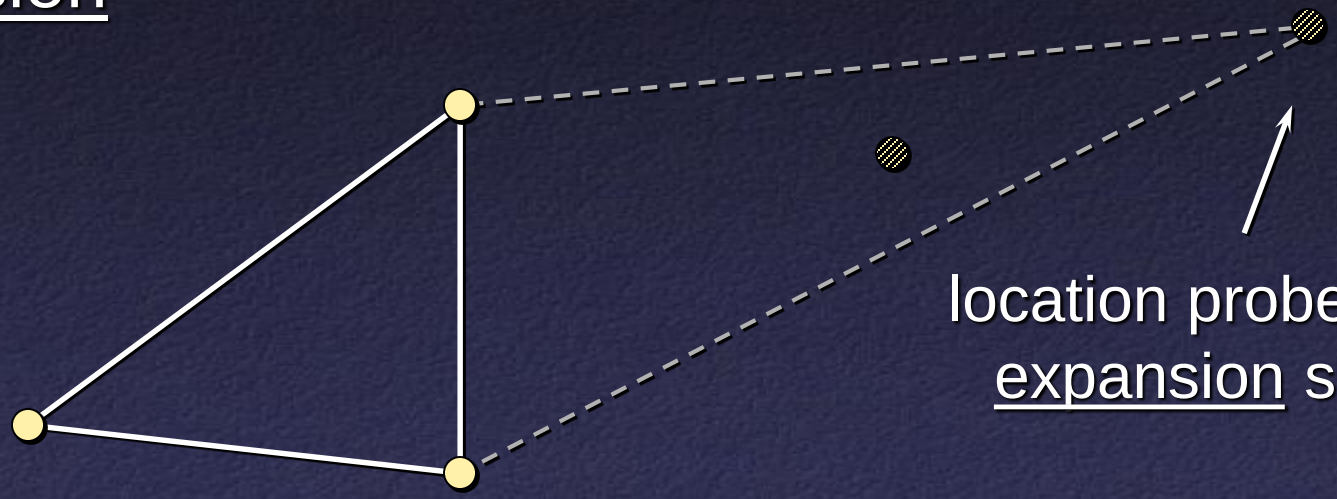


location probed by reflection step

worst point
(highest function value)

Downhill Simplex Method (Nelder-Mead)

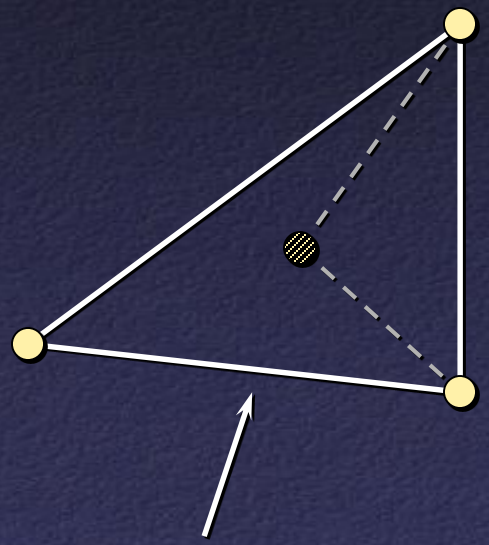
- If reflection resulted in best (lowest) value so far, try an expansion



- Else, if reflection helped at all, keep it

Downhill Simplex Method (Nelder-Mead)

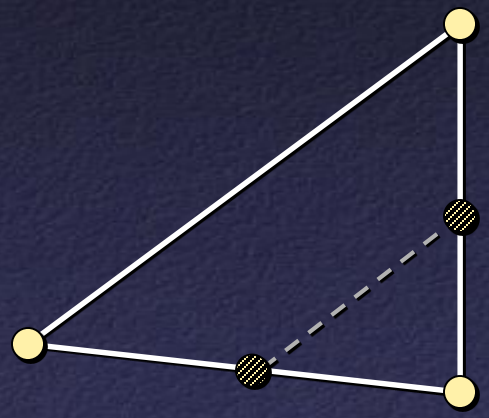
- If reflection didn't help (reflected point still worst) try a contraction



location probed by contration step

Downhill Simplex Method (Nelder-Mead)

- If all else fails shrink the simplex around the *best* point



Downhill Simplex Method (Nelder-Mead)

- Method fairly efficient at each iteration (typically 1-2 function evaluations)
- Can take *lots* of iterations
- Somewhat flakey – sometimes needs *restart* after simplex collapses on itself, etc.
- Benefits: simple to implement, doesn't need derivative, doesn't care about function smoothness, etc.