# Mathematical Analysis summary

P. Papageorgiou, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr

Version 3.0

Artificial Intelligence &
Information Analysis Lab

# Mathematical Analysis

- **1D/2D/3D Functions**
- Differentiation
- Integration
- Partial differentiation
- Double integrals
- Optimization
- Fourier transforms

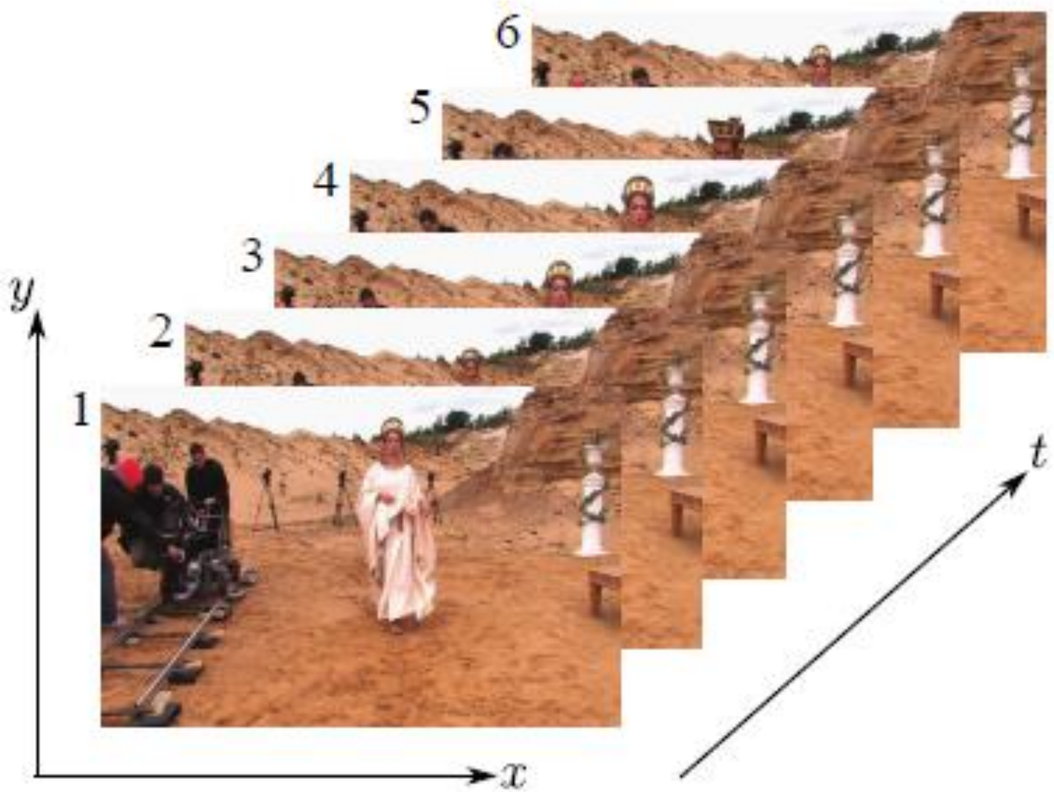
# 1D, 2D, 3D analog signals / functions

- 1D signals of the form $f(t): \mathbb{R} \rightarrow \mathbb{R}$
  - *Speech, music.*

- 2D signals of the form $f(x, y): \mathbb{R}^2 \rightarrow \mathbb{R}$
  - *Greyscale images.*

- 3D signals of the form $f(x, y, z): \mathbb{R}^3 \rightarrow \mathbb{R}$
  - *Video signals* $f(x, y, t): \mathbb{R}^3 \rightarrow \mathbb{R}$

# 3D data types: video signal $f(x, y, t)$



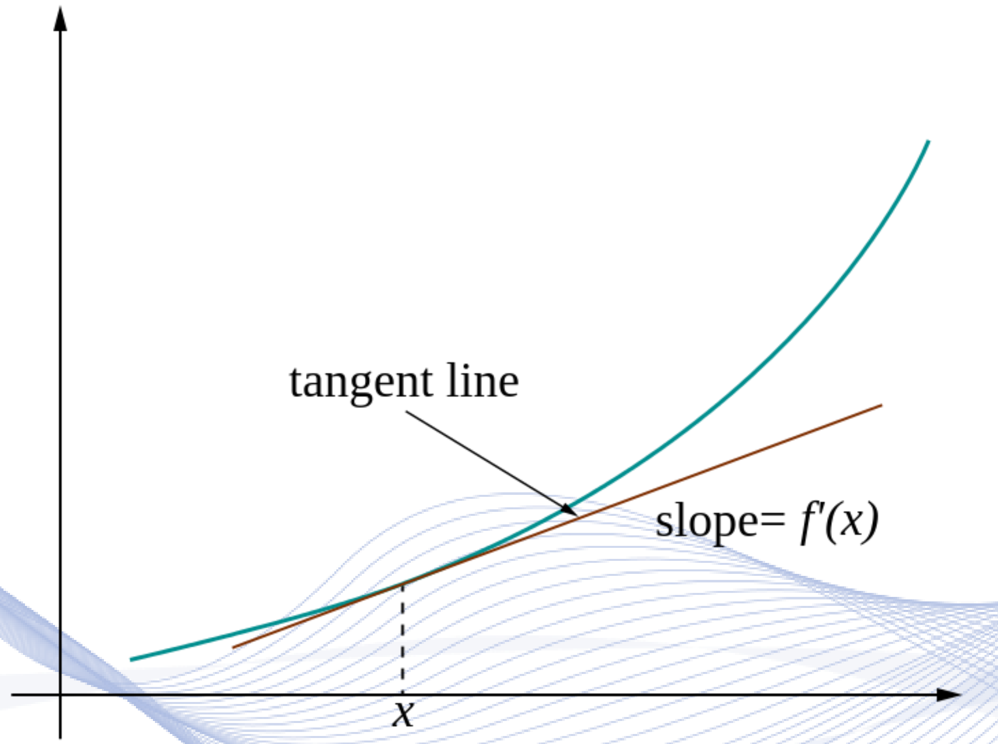
# 1D, 2D, 3D discrete signals

- 1D signals of the form $f(n): \mathbb{Z} \to \mathbb{R}$
  - *Digital speech, music*
- 2D signals of the form $f(i,j): \mathbb{Z}^2 \to \mathbb{R}$
  - ***Digital greyscale images***
- 3D signals of the
  - *Volumetric images* $f(i,j,k): \mathbb{Z}^3 \to \mathbb{R}$
  - ***Digital video signals*** $f(i,j,k): \mathbb{Z}^3 \to \mathbb{R}$

# Mathematical Analysis

- 1D/2D/3D Functions
- **Differentiation**
- Integration
- Partial differentiation
- Double integrals
- Optimization
- Fourier transforms

# 1D Differentiation




- The derivative of a function at a specific point is the rate of change of function output with respect to its input.

- For 1D continuous functions, it is the slope of the tangent line to the function graph at that point:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}.$$

# Numerical Differentiation

- Numerical Differentiation performs approximate differentiation, on discrete function values:

$$\hat{f}'(x) = \frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x}.$$

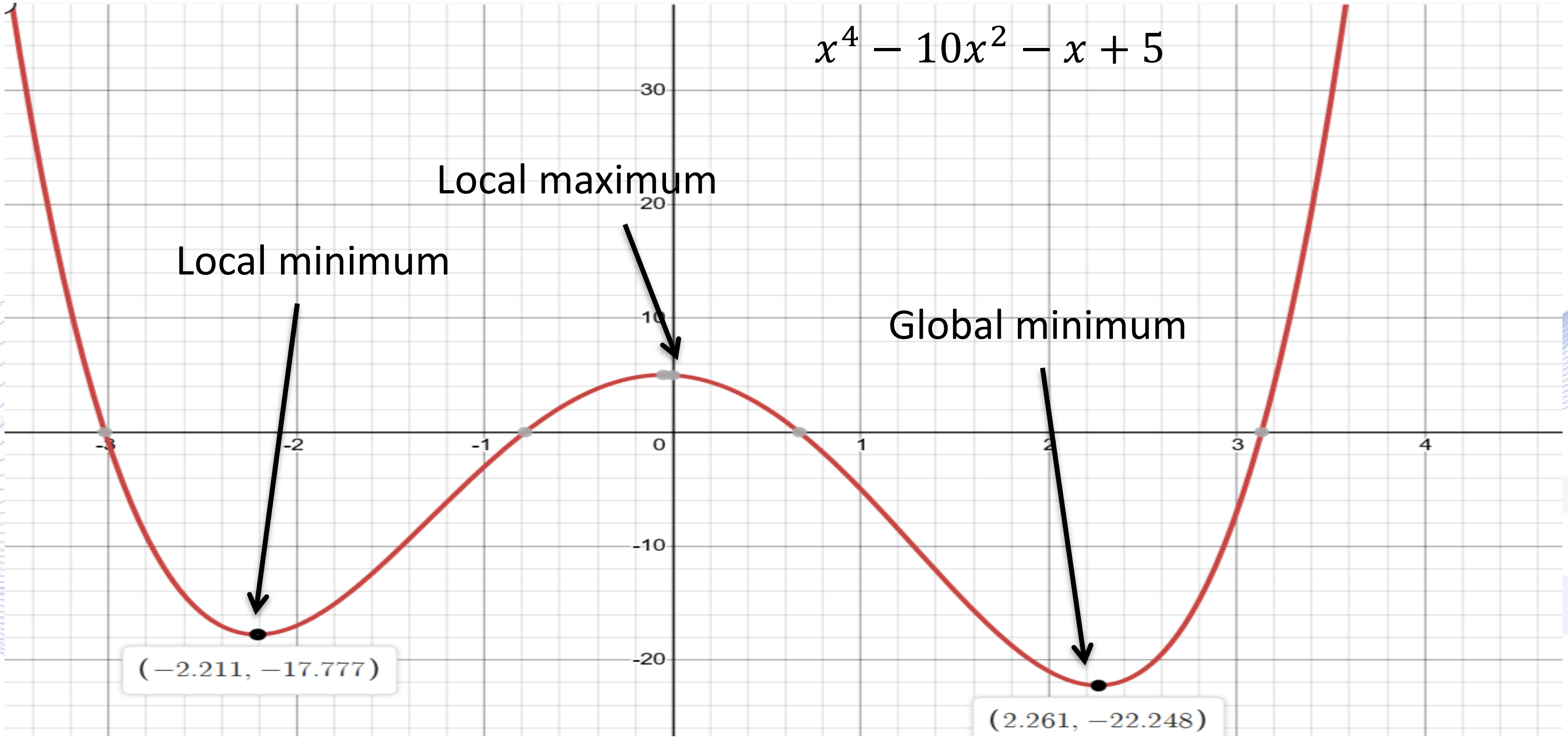
- For small sampling step $\Delta x$, it is equal to the slope of a nearby secant line through the points $(x - \Delta x, f(x - \Delta x))$ and $(x + \Delta x, f(x + \Delta x))$.

# Numerical Differentiation

- Numerical Differentiation is a high pass linear system:
- It amplifies high frequencies.
- It is sensitive to noise.

# Function Minima

- Local and global minima.
- Derivate is zero at function minima and maxima.



# Mathematical Analysis

- 1D/2D/3D Functions
- Differentiation
- **Integration**
- Partial differentiation
- Double integrals
- Optimization
- Fourier transforms

# Integration

- Integration is the reverse of differentiation.
- ***Indefinite integral*** for a function f(x):

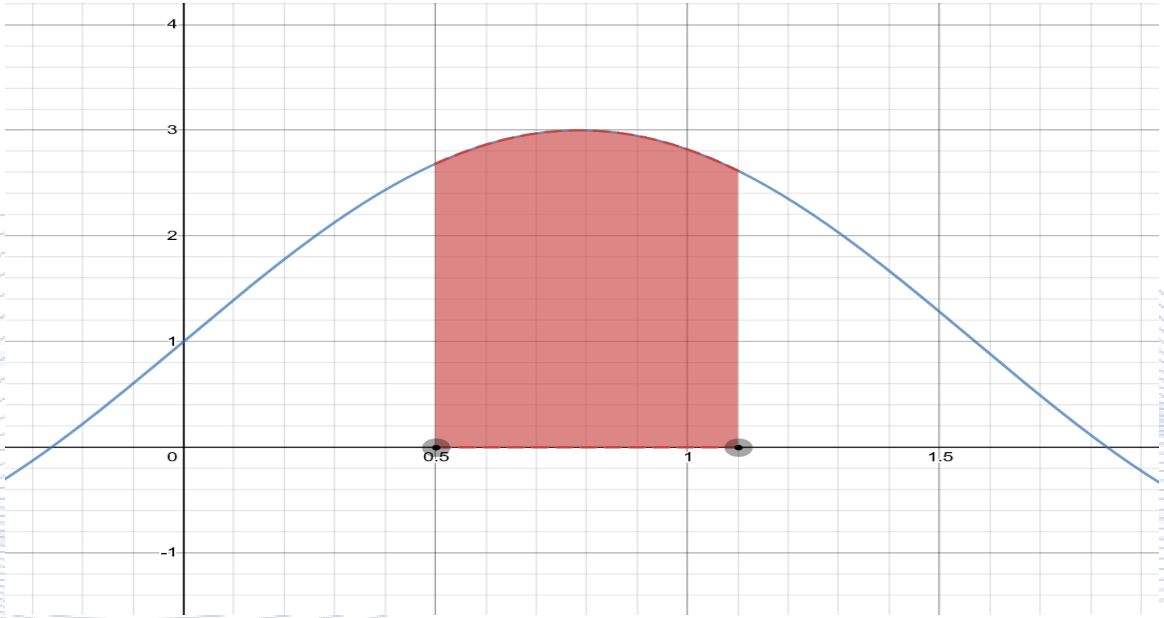
$$\int f(x)dx = F(x) + \mathrm{c}.$$

- Fundamental theorem of calculus.
- ***Definite integral*** in the interval $[a, b]$:

$$\int_a^b f(x)dx = F(b) - F(a).$$

# Integration

- Definite integrals is the area under a function curve for the interval $[a, b]$.




Artificial Intelligence & Information Analysis Lab

# Mathematical Analysis

- 1D/2D/3D Functions
- Differentiation
- Integration
- **Partial differentiation**
- Double integrals
- Optimization
- Fourier transforms

# Partial Differentiation

- A ***partial derivative*** of a multivariate function (of several variables) is its derivative with respect to one of those variables.
- Their vector defines multivariate ***function grad***.

- Total derivative of a multivariate function allows al variables to vary.

# Partial Differentiation

For functions of two variables $f(x, y)$:

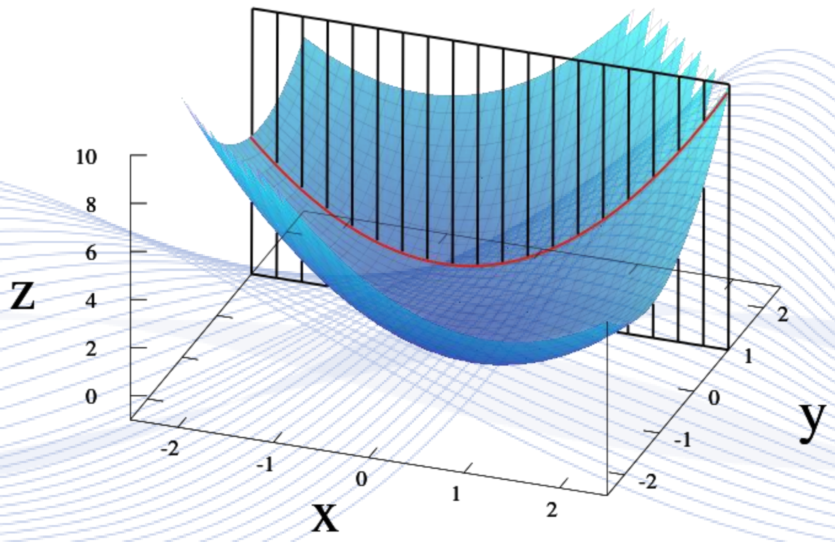
- the partial derivatives are $\partial f/\partial x, \partial f/\partial y$ .
- the ***grad*** of a function is defined by the vector:

$$\nabla f = [\partial f/\partial x, \partial f/\partial y]^T.$$

- Grad direction shows the steepest local ascent direction.
- Grad magnitude shows the rate of the ascent.

# Partial Differentiation

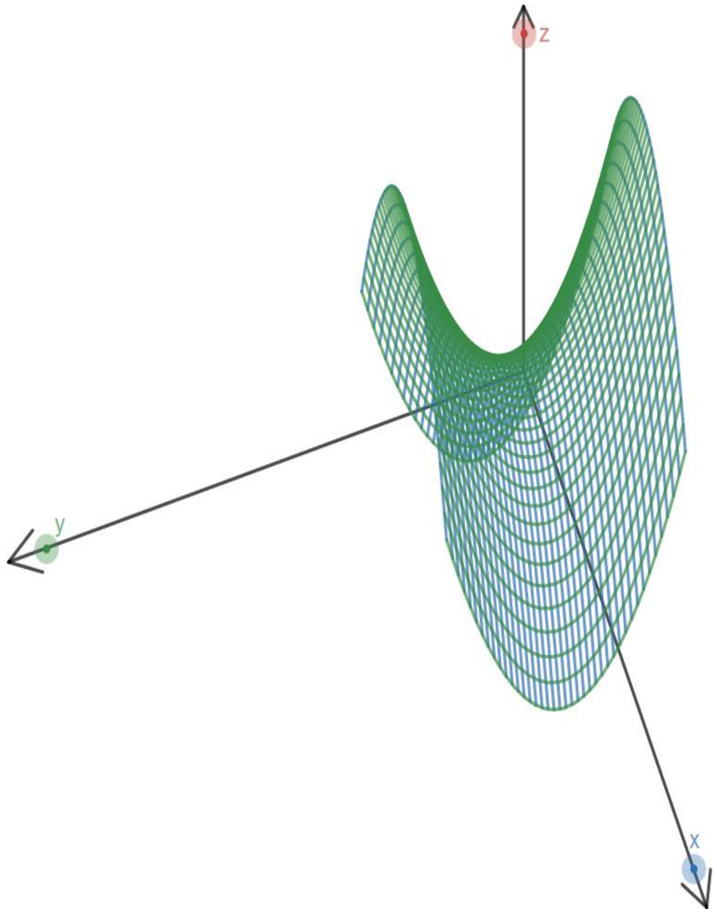
- At maxima/minima, saddle points: $\nabla f = 0$.



# Saddle Point

- ***Saddle point***: a point on the surface of a function where the slopes in orthogonal directions are all zero (critical point), but which is neither a minimum nor a maximum.

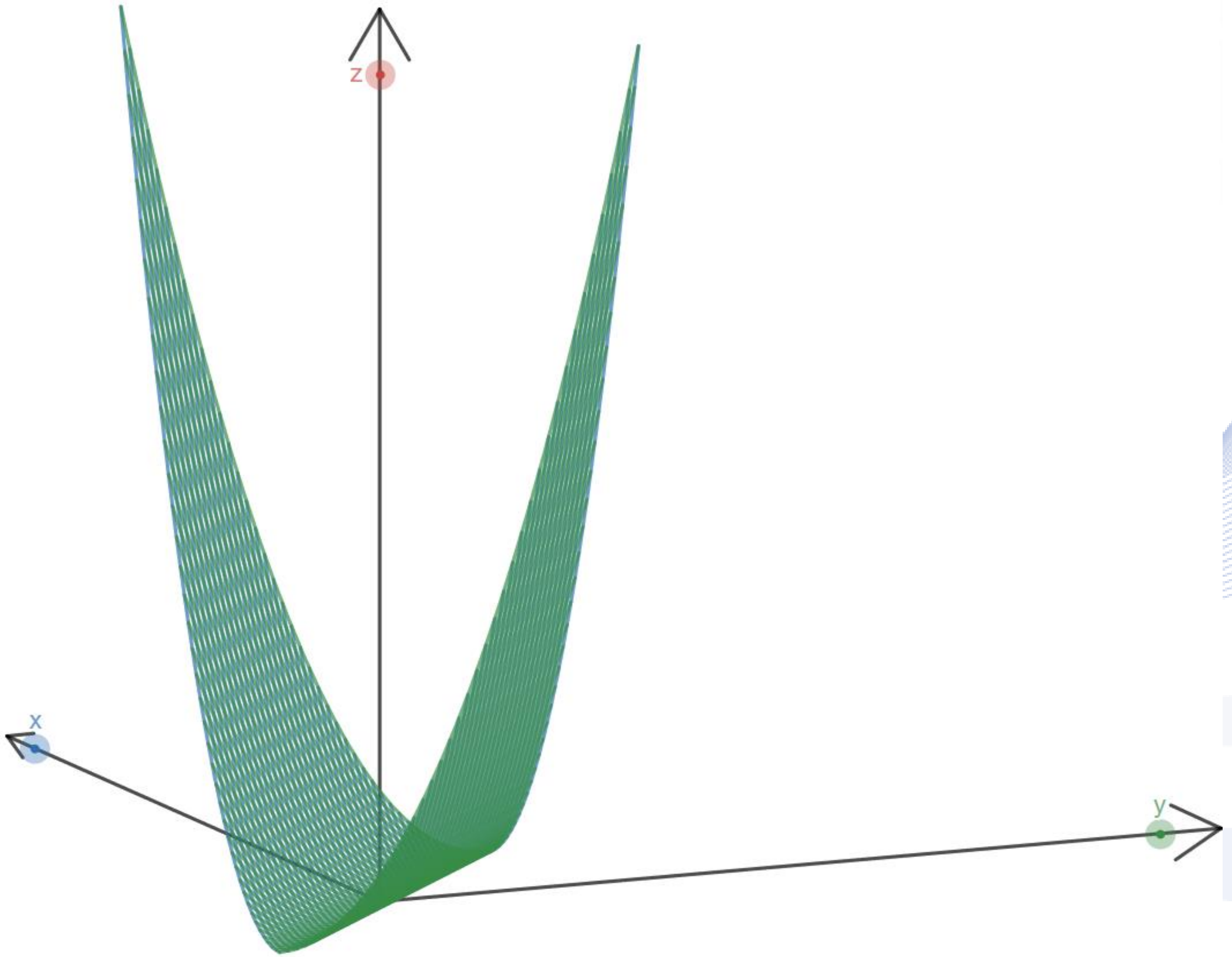
$f(x, y) = x^2 - y^2$



# Weak Minimum

$$f(x, y) = 0.5x^2 - xy + 0.5y^2$$

The function is almost flat along a certain direction.



# Hessian matrix

- For functions of many variables $f(\mathbf{x})$, $\mathbf{x} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$ the Hessian matrix is given by:

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

# Differentiation of linear functions

The differentiation of a linear function $\mathbf{y} \in \mathbb{R}^m$ , $\mathbf{x} \in \mathbb{R}^n, \mathbf{A} \in \mathbb{R}^{m \times n}$:

$$\mathbf{y} = \mathbf{A}\mathbf{x},$$

$$\begin{bmatrix} y_1 \\ . \\ . \\ . \\ y_m \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ . \\ . \\ . \\ x_n \end{bmatrix}$$

Is given by:

$$\nabla_{\mathbf{x}} \mathbf{y} = \mathbf{A}.$$

# Vector and Matrix Derivatives

- $\nabla_{\mathbf{x}}\left(\mathbf{x}^{\mathrm{T}}\mathbf{a}\right) = \nabla_{\mathbf{x}}\left(\mathbf{a}^{\mathrm{T}}\mathbf{x}\right) = \mathbf{a}$
- $\nabla_{\mathbf{X}}\left(\mathbf{a}^{\mathrm{T}}\mathbf{X}\mathbf{b}\right) = \mathbf{a}\mathbf{b}^{\mathrm{T}}$
- $\nabla_{\mathbf{x}}\left(\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}\right) = (\mathbf{A} + \mathbf{A}^{\mathrm{T}})\mathbf{x}$
- $\nabla_{\mathbf{X}}\left(\mathbf{b}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{A}\mathbf{X}\mathbf{c}\right) = \mathbf{A}^{\mathrm{T}}\mathbf{X}\mathbf{b}\mathbf{c}^{\mathrm{T}} + \mathbf{A}\mathbf{X}\mathbf{c}\mathbf{b}^{\mathrm{T}}$
- $\nabla_{\mathbf{X}}\left((\mathbf{X}\mathbf{b} + \mathbf{c})^{\mathrm{T}}\mathbf{A}(\mathbf{X}\mathbf{b} + \mathbf{c})\right) = (\mathbf{A} + \mathbf{A}^{\mathrm{T}})(\mathbf{X}\mathbf{b} + \mathbf{c})\mathbf{b}^{\mathrm{T}}$

# Overdetermined System of Linear Equations

Let an overdetermined system of linear equations be:

$$\mathbf{Mx}=\mathbf{y}.$$

- $\mathbf{M}$ is a $n \times m$ matrix,
- $\mathbf{y}$ is a known $n$-dimensional vector
- $\mathbf{x}$ is an unknown $m$-dimensional vector.
- If $n > m$, there are more constraints than unknowns.
- The system is overdetermined, with no solution (except for degenerate cases).

# Numerical partial differentiation

- Let $f$ be a function defined on a set $\mathcal{A} \subseteq \mathbb{R}^2$ and suppose that the points $(x, y), (x + r\Delta x, y), (x, r\Delta y + y)$ all lie in $\mathcal{A}$ for any $r \in [0,1]$. Then the two partial derivatives $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}$ can be approximated by:

$$\frac{\partial f}{\partial x}(x, y) \approx \frac{f(x+\Delta x, y) - f(x, y)}{\Delta x},$$

$$\frac{\partial f}{\partial y}(x, y) \approx \frac{f(x, y+\Delta y) - f(x, y)}{\Delta y}.$$

# Image Edge Detection

Local image differentiation techniques can produce edge detector operators.

- Image luminance gradient:

$$\nabla f(x,y) = [\frac{\partial f}{\partial x} \quad \frac{\partial f}{\partial y}]^T \triangleq [f_x \quad f_y]^T.$$

- Edge magnitude:

$$e(x,y) = \sqrt{f_x^2(x,y) + f_y^2(x,y)}.$$

- Edge direction angle:

$$\varphi(x,y) = \arctan(\frac{f_y}{f_x}).$$

# Image Edge detection

Gradient estimates can be obtained by using gradient operators of the form:

$$\hat{f}_x = \mathbf{w}_1^T \mathbf{x},$$
$$\hat{f}_y = \mathbf{w}_2^T \mathbf{x}.$$

- $\mathbf{x}$: local image pixel vector,
- $\mathbf{w}_1$ , $\mathbf{w}_2$: weight vectors (gradient masks).

# Image Edge Detection

Gradient masks examples:

| | | |
|---|---|---|
| -1 | 0 | 1 |
| -1 | 0 | 1 |
| -1 | 0 | 1 |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| -1 | -1 | -1 |

Prewitt edge detector masks

| | | |
|---|---|---|
| -1 | 0 | 1 |
| -2 | 0 | 2 |
| -1 | 0 | 1 |

| | | |
|---|---|---|
| 1 | 2 | 1 |
| 0 | 0 | 0 |
| -1 | -2 | -1 |

Sobel edge detector masks

**Edge templates** are masks that can be used to detect edges along different directions. Such masks of size 3X3 are:

| | | |
|---|---|---|
| -1 | 0 | 1 |
| -1 | 0 | 1 |
| -1 | 0 | 1 |

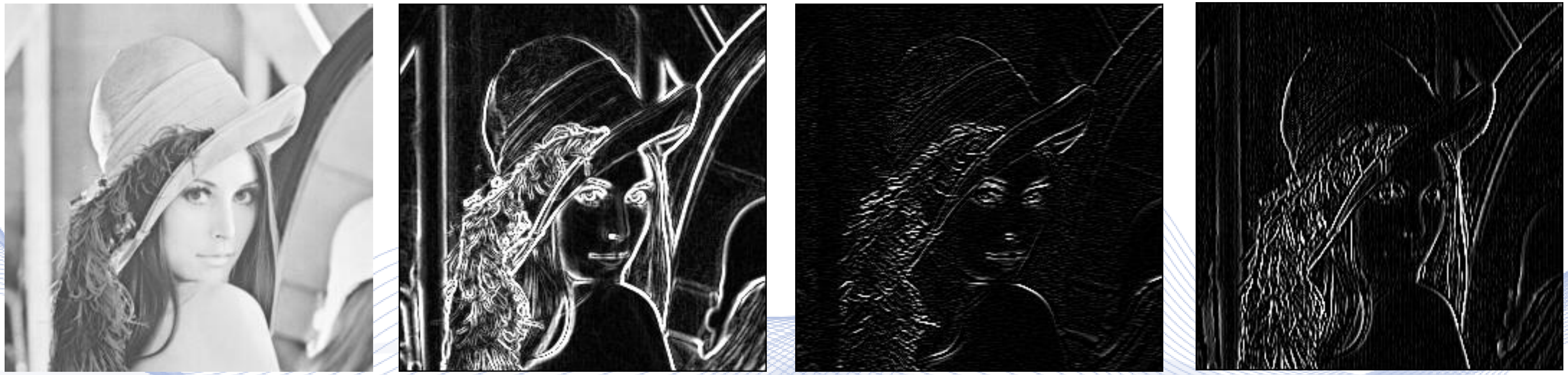
| | | |
|---|---|---|
| 1 | 1 | 1 |
| 0 | 0 | 0 |
| -1 | -1 | -1 |

| | | |
|---|---|---|
| 0 | 1 | 1 |
| -1 | 0 | 1 |
| -1 | -1 | 0 |

| | | |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 0 | -1 |
| 0 | -1 | -1 |

Kirsch edge detector masks

# Image Edge detection



a) Lenna image; b) Sobel edge detector output ; c) horizontal edges; d) vertical edges.

# Partial Differentiation in 3D imaging

***3D image gradient***:

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}\right]^T.$$

- Numerical estimation of its components at position $[x, y, z]^T$ assuming $\Delta x = \Delta y = \Delta z = 1$:

$$\widehat{f_x} = f(x+1, y, z) - f(x-1, y, z),$$

$$\widehat{f_y} = f(x, y+1, z) - f(x, y-1, z),$$

$$\widehat{f_z} = f(x, y, z+1) - f(x, y, z-1).$$

# Partial Differentiation in 3D imaging

- Normalized gradient components:

$$n_k = \frac{f_k}{\sqrt{f_x^2 + f_y^2 + f_z^2}}, \qquad k \in \{x, y, z\}.$$

- ***Laplacian operator***:

$$\nabla^2 f(x,y,z) = \frac{\partial^2 f(x,y,z)}{\partial x^2} + \frac{\partial^2 f(x,y,z)}{\partial y^2} + \frac{\partial^2 f(x,y,z)}{\partial z^2}.$$

# Partial Differentiation in 3D imaging

Numerical differentiation for:

- 3D volumetric images $f(n_1, n_2. n_3)$;
- Spatiotemporal signals (digital video) $f(n_1, n_2. n_t)$:

$$\widehat{f_x} = \frac{1}{4}\{f(n_1 + 1, n_2, n_t) - f(n_1, n_2, n_t) + f(n_1 + 1, n_2 + 1, n_t) - f(n_1, n_2 + 1, n_t) + f(n_1 + 1, n_2, n_t + 1) - f(n_1, n_2, n_t + 1) + f(n_1 + 1, n_2 + 1, n_t + 1) - f(n_1, n_2 + 1, n_t + 1)\}.$$

# Partial Differential Equations

- ***Laplacian equation***:

$$\nabla^2 f(x,y,z) = 0\,.$$

- ***Wave equation*** describes electromagnetic fields:

$$\nabla^2 f(x,y,z) = \frac{\partial^2 f(x,y,z)}{\partial x^2} + \frac{\partial^2 f(x,y,z)}{\partial y^2} + \frac{\partial^2 f(x,y,z)}{\partial z^2} =$$

$$= \frac{1}{c^2}\frac{\partial^2}{\partial t^2} f(x,y,z,t),$$

- $c$: speed of light.

# Partial Differential Equations

- ***Diffusion equation*** describes diffusion and transfer processes:

$$\nabla^2 f(x,y,z) = \frac{\partial^2 f(x,y,z)}{\partial x^2} + \frac{\partial^2 f(x,y,z)}{\partial y^2} + \frac{\partial^2 f(x,y,z)}{\partial z^2} =$$

$$= \frac{1}{c}\frac{\partial}{\partial t} f(x,y,z,t).$$

- $c$: diffusion coefficient.

# Partial Differential Equations

- Anisotropic diffusion within a 6-voxel neiborhood:

$$\frac{\partial}{\partial t}f(\mathbf{x},t) \cong$$

$$\cong \frac{1}{\Delta x^2}\left[c\left(x+\frac{\Delta x}{2},y,z,t\right)\left(f(x+\Delta x,y,z,t)-f(x,y,z,t)\right)-c\left(x-\frac{\Delta x}{2},y,z,t\right)\left(f(x,y,z,t)-f(x-\Delta x,y,z,t)\right)\right]+$$

$$+\frac{1}{\Delta \mathrm{y}^2}\left[c\left(x,y+\frac{\Delta \mathrm{y}}{2},z,t\right)\left(f(x,y+\Delta \mathrm{z},z,t)-f(x,y,z,t)\right)-c\left(x,y-\frac{\Delta \mathrm{y}}{2},z,t\right)\left(f(x,y,z,t)-f(x,y-\Delta y,z,t)\right)\right]+$$

$$+\frac{1}{\Delta \mathrm{z}^2}\left[c\left(x,y,z+\frac{\Delta \mathrm{z}}{2},t\right)\left(f(x,y,z+\Delta \mathrm{z},t)-f(x,y,z,t)\right)-c\left(x,y,z-\frac{\Delta \mathrm{z}}{2},t\right)\left(f(x,y,z,t)-f(x,y,z-\Delta \mathrm{z},t)\right)\right].$$

# Mathematical Analysis

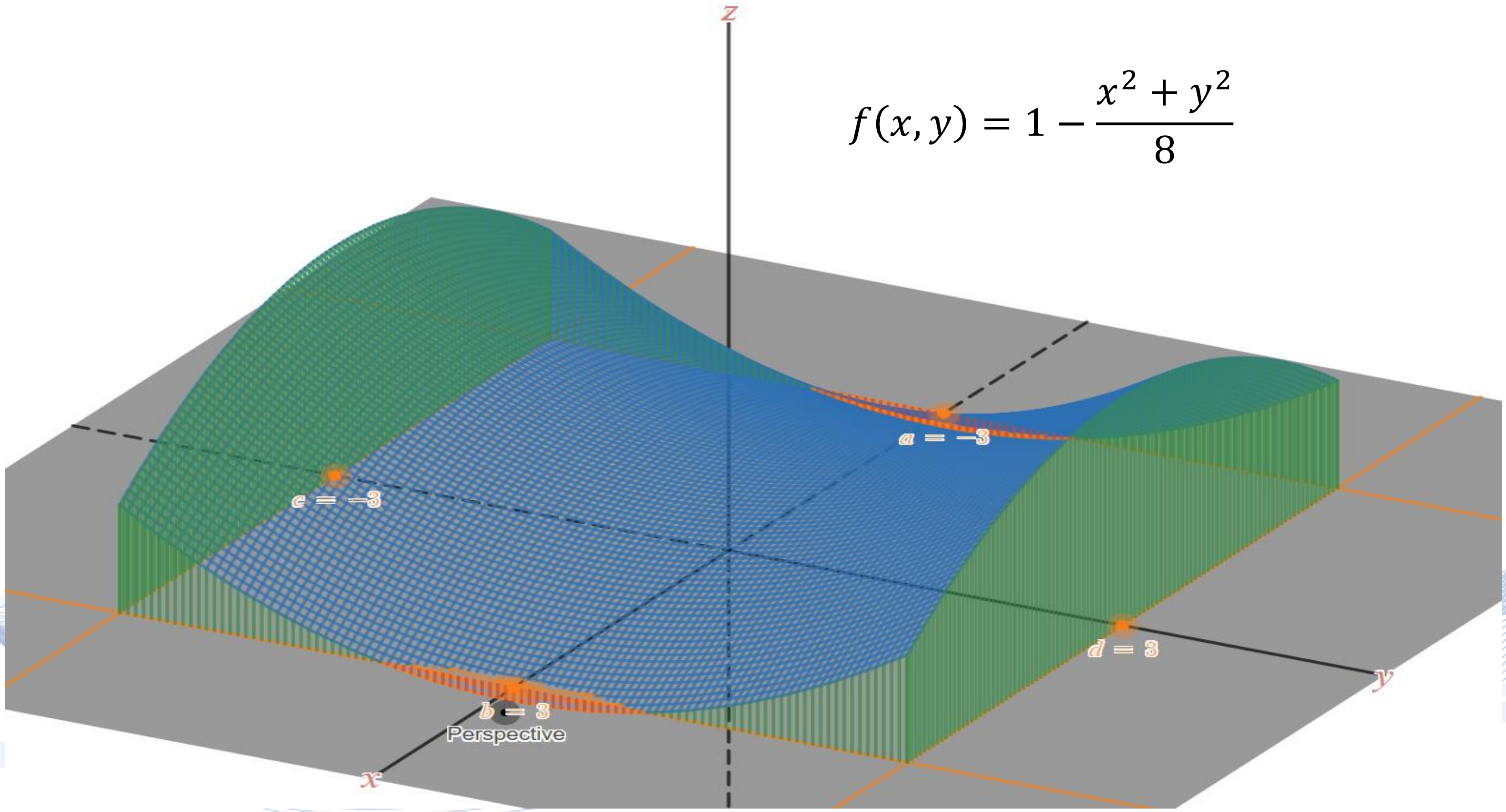
- 1D/2D/3D Functions
- Differentiation
- Integration
- Partial differentiation
- **Double integrals**
- Optimization
- Fourier transforms

# Double Integrals

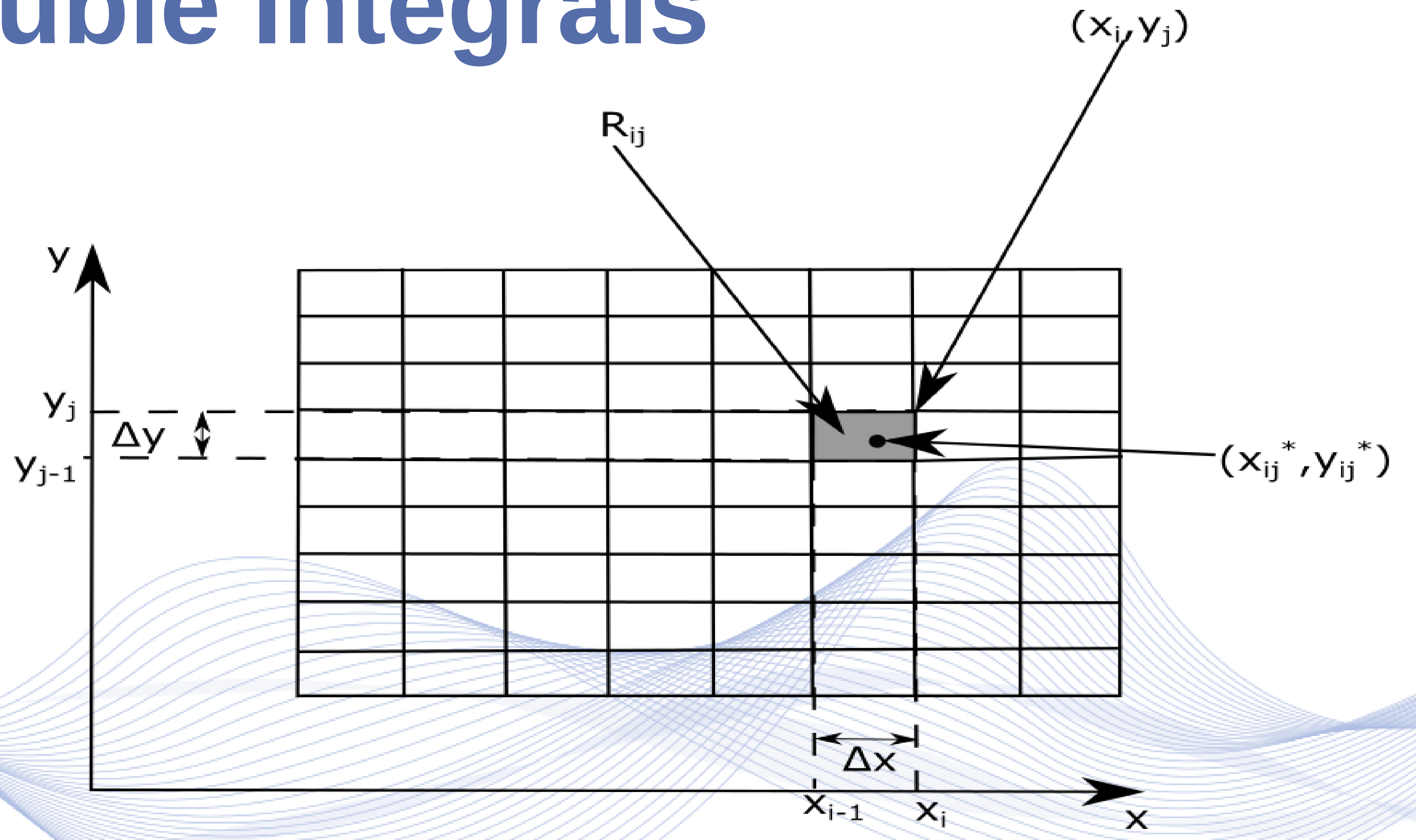
- Double integrals integrate a function of two variables over a region $\mathcal{A} \subseteq \mathbb{R}^2$.
- It represents the volume of the region between the surface defined by the function $z = f(x, y)$ and the plane $(x, y)$.

# Double Integrals

$$f(x,y) = 1 - \frac{x^2+y^2}{8}$$



# Double Integrals




Splitting the integration area into elementary rectangles.

# Double Integrals

- The volume $V$ of the function under a surface is approximately given by:

$$V \approx \sum_{i=1}^{n} \sum_{j=1}^{m} f\left(x_i^*, y_j^*\right).$$

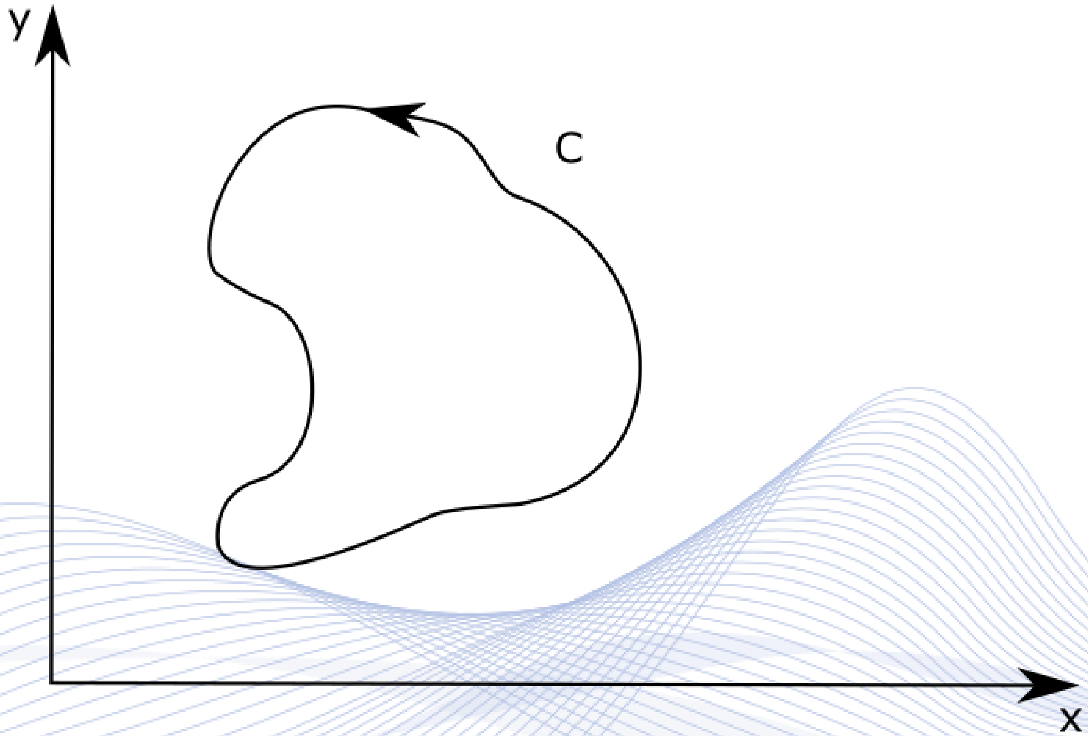
- At the limit $n, m \to \infty$ :

$$V = \iint_R f(x, y) = \lim_{n,m\to\infty} \sum_{i=1}^{n} \sum_{j=1}^{m} f\left(x_i^*, y_j^*\right) \Delta A.$$

# Contour Integration

- A ***curve*** is a continuous function from closed interval of the real line to the complex plane $f(z): [a, b] \to \mathbb{C}$.
- A ***smooth curve*** is a curve with non-vanishing, continuous derivative, such that each point is traversed only once with the exception of curve endpoints.
- A ***contour*** is a directed curve, which is made up of a finite sequence of directed smooth curves, whose start/endpoints match.

# Contour Integration




Contour integral over a real valued parameter.

# Contour Integration

***Unit circle*** in the complex plane $\mathbb{C}$:

$$|z| = 1.$$

- Parameterized unit circle equation:

$$z(\omega) = e^{i\omega}, \quad \omega \in [0,2\pi].$$

Since:

$$\frac{dz}{d\omega} = ie^{i\omega},$$

contour integration becomes:

$$\oint_C \frac{1}{z} dz = \int_0^{2\pi} \frac{1}{e^{i\omega}} ie^{i\omega} d\omega = i\int_0^{2\pi} 1 d\omega = [t]_0^{2\pi} = 2\pi i.$$

Artificial Intelligence & Information Analysis Lab

# Mathematical Analysis

- 1D/2D/3D Functions
- Differentiation
- Integration
- Partial differentiation
- Double integrals
- **Optimization**
- Fourier transforms

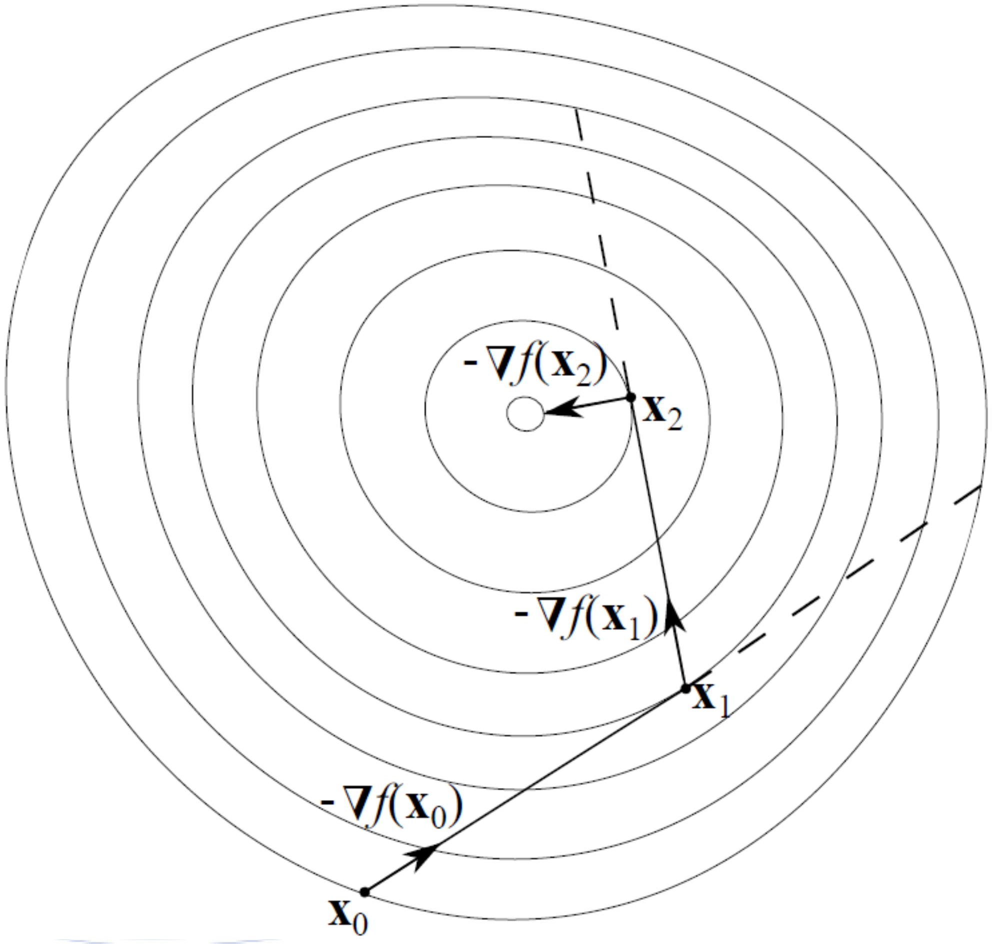
# Steepest Gradient Descent

- If function $f(\mathbf{x})$ is defined and differentiable in a neigh-borhood of a point $\mathbf{x}_t$, then $f(\mathbf{x})$ decreases fastest, going from $\mathbf{x}_t$ to $\mathbf{x}_{t+1}$ following the direction of the negative gradient of $f(\mathbf{x})$ at $\mathbf{x}_t$:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - a\nabla f(\mathbf{x}_t).$$

- $a$: step used to update the vector $\mathbf{x}_{t+1}$ at each iteration $t$.
- $f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t)$ and sequence $\mathbf{x}_t$ converges to a local minimum of $f(\mathbf{x}_t)$.

# Steepest Gradient Descent



# Mathematical Analysis

- 1D/2D/3D Functions
- Differentiation
- Integration
- Partial differentiation
- Double integrals
- Optimization
- **Fourier transforms**

# 1D Fourier transform

- *1D Fourier transform:*

$$F(\Omega) = \int_{-\infty}^{\infty} f(t) e^{-i\Omega t} dx.$$

- *1D inverse Fourier transform:*

$$f(t) = \int_{-\infty}^{\infty} F(\Omega)\, e^{i\Omega t} d\Omega.$$

# 1D Fourier transform

***1D convolution***:

$$g(t) = f(t) * h(t)$$

1D convolution support in the Fourier domain:

$$G(\Omega) = F(\Omega)H(\Omega).$$

# 2D Fourier transform

- *2D Fourier transform:*

$$F(\Omega_x, \Omega_y) \triangleq \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \exp(-i\Omega_x x - i\Omega_y y) dx dy .$$

- *2D inverse Fourier transform:*

$$f(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\Omega_x, \Omega_y) \exp(i\Omega_x x + i\Omega_y y) d\Omega_x d\Omega_y .$$

# Spatial image content


$$f(x, y) = \sin(20\pi x + 8\pi y)$$

$$(\Omega_x = 20\pi, \Omega_x = 8\pi)$$

# 3D Fourier transform

- ***3D Fourier transform***:

$$F(\Omega_x, \Omega_y, \Omega_z) \triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y, z) exp(-i\Omega_x x - i\Omega_y y - i\Omega_z z) dxdy\, dz\,.$$

- *3D inverse Fourier transform:*

$$f(x, y, z) = \frac{1}{(2\pi)^3} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(\Omega_x, \Omega_y, \Omega_z) exp(i\Omega_x x + i\Omega_y y + i\Omega_z z) d\Omega_x d\Omega_y d\Omega_z\,.$$

# 3D Fourier transform

***3D convolution***:

$$g(x,y,z) = f(x,y,z) *** h(x,y,z).$$

2D convolution support in the Fourier domain:

$$G(\Omega_x, \Omega_y, \Omega_z) = F(\Omega_x, \Omega_y, \Omega_z) H(\Omega_x, \Omega_y, \Omega_z) .$$

# 3D Fourier transform

***3D Discrete Fourier Transform***:

$$F(k_1,k_2,k_3)=\sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1}\sum_{n_3=0}^{N_3-1} f(n_1,n_2,n_3)\exp(-j\frac{2\pi n_1 k_1}{N_1},-j\frac{2\pi n_2 k_2}{N_2},-j\frac{2\pi n_3 k_3}{N_3}),$$

$$f(n_1,n_2,n_3)=\frac{1}{N_1N_2N_3}\sum_{k_1=0}^{N_1-1}\sum_{k_2=0}^{N_2-1}\sum_{k_2=0}^{N_2-1} F(k_1,k_2,k_3)\exp(j\frac{2\pi n_1 k_1}{N_1},j\frac{2\pi n_2 k_2}{N_2},j\frac{2\pi n_3 k_3}{N_3}).$$

# Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr