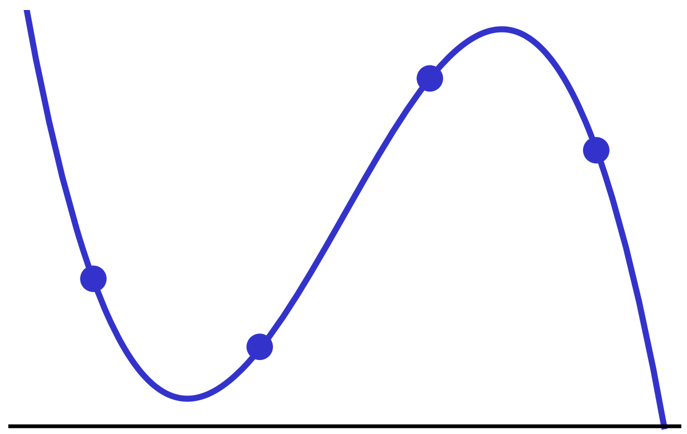
COL865: Special Topics in Computer Applications

Physics-Based Animation

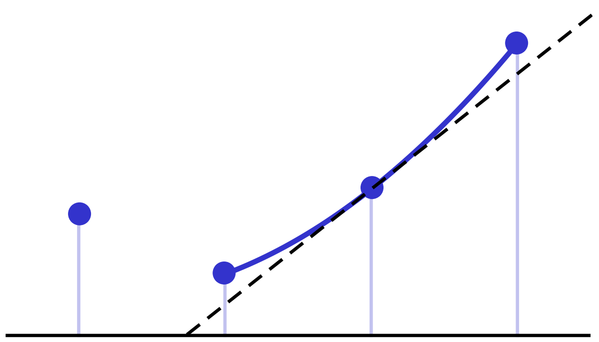
3 — Numerical analysis contd.

Last class

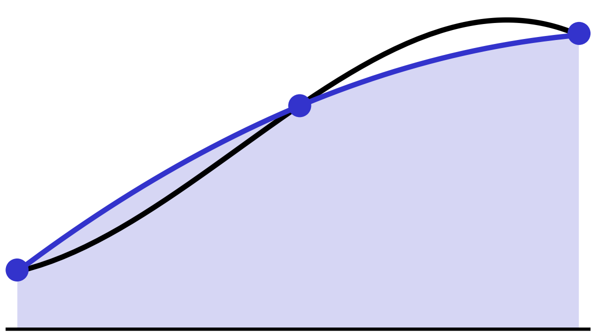
- **Interpolation** via linear combinations of basis functions
  - Polynomial and piecewise polynomial interpolation



- **Numerical differentiation** via finite differences
  - Accuracy analysis via Taylor series



- **Numerical integration** a.k.a. quadrature



FYI

All reading material will be posted on Moodle

All slides will be posted to the course web page after class

Starter graphics code will be posted soon
(This is optional, you can use your own code if you want)

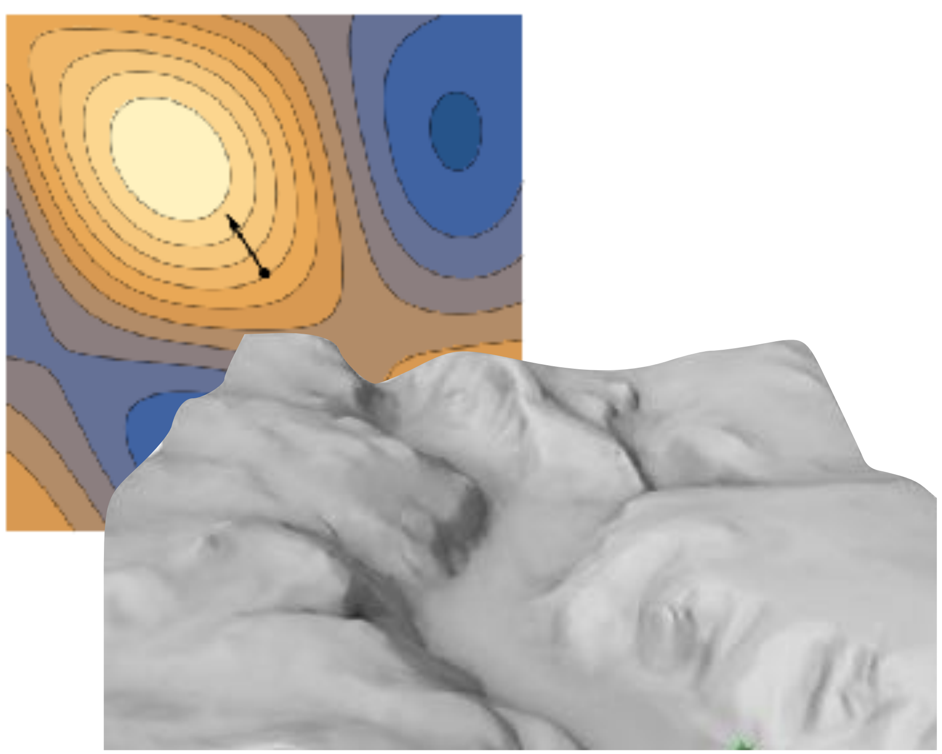
Functions in higher dimensions

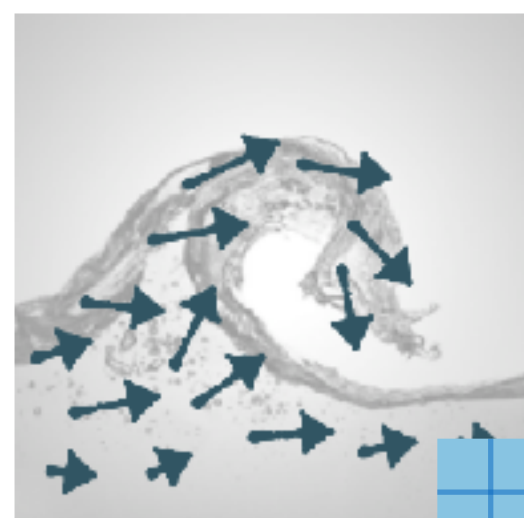
Functions in higher dimensions

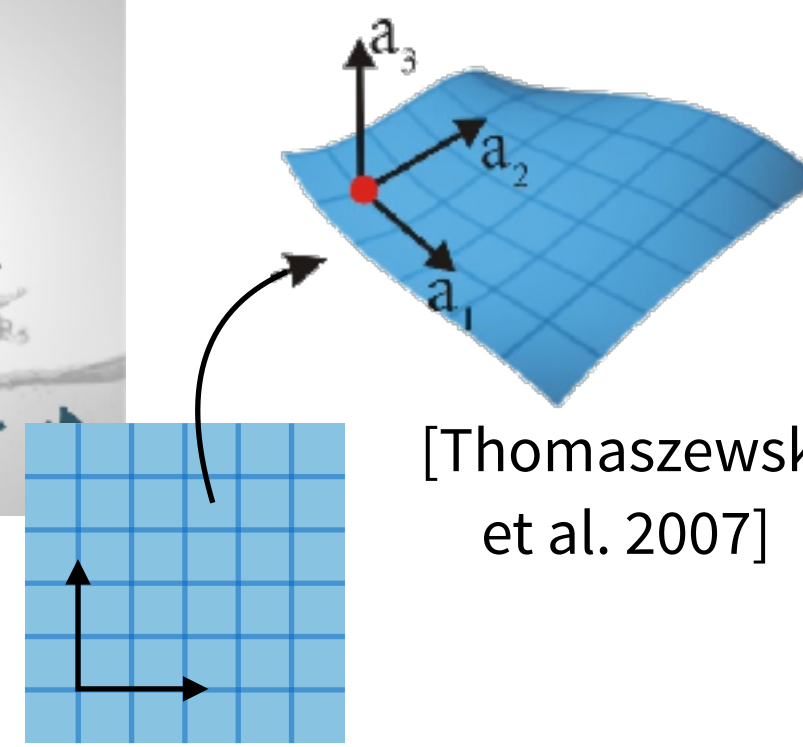
So far we've only considered $f : \mathbb{R} \to \mathbb{R}$

Scalar-valued functions $\mathbb{R}^n \to \mathbb{R}$, vector-valued functions $\mathbb{R}^n \to \mathbb{R}^m$

- Temperature of air $\mathbb{R}^3 \to \mathbb{R}$, height of terrain $\mathbb{R}^2 \to \mathbb{R}$
- Flow field $\mathbb{R}^3 \to \mathbb{R}^3$, deformation of cloth $\mathbb{R}^2 \to \mathbb{R}^3$







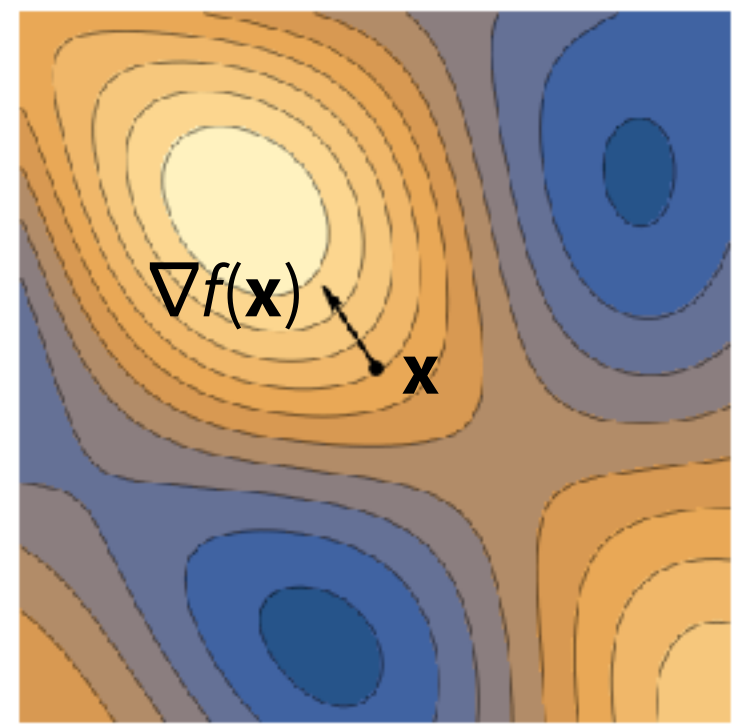
[Thomaszewski et al. 2007]

Derivatives in higher dimensions

A derivative is something such that $f(x+h) \approx f(x) + \mathrm{d}f \cdot h$

For a scalar-valued function $f : \mathbb{R}^n \to \mathbb{R}$, this is the **gradient**:

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \partial f / \partial x_1 \\ \partial f / \partial x_2 \\ \vdots \\ \partial f / \partial x_n \end{bmatrix}$$



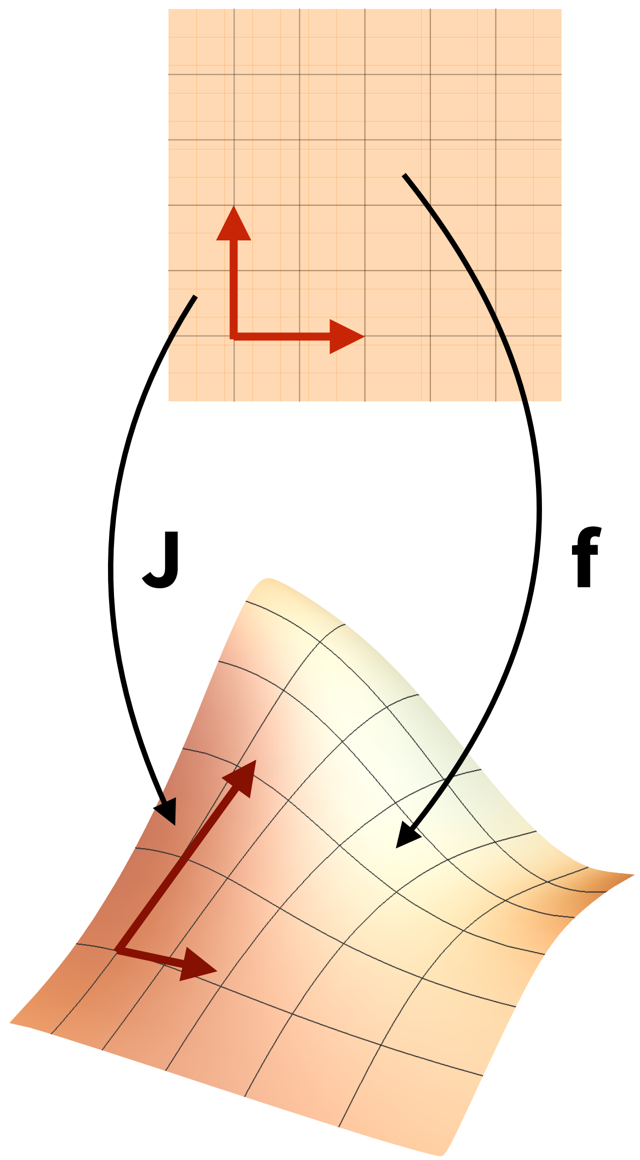
Points in direction of steepest ascent

Important fact (via Taylor series): $f(\mathbf{x}+\mathbf{h}) \approx f(\mathbf{x}) + \nabla f(\mathbf{x}) \cdot \mathbf{h}$

The Jacobian

For a vector-valued function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$, the derivative is an $m \times n$ matrix called the **Jacobian**:

$$\mathbf{J}(\mathbf{x}) = \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 & \cdots & \partial f_1/\partial x_n \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 & \cdots & \partial f_2/\partial x_n \\ \vdots & \vdots & \ddots & \vdots \\ \partial f_m/\partial x_1 & \partial f_m/\partial x_2 & \cdots & \partial f_m/\partial x_n \end{bmatrix}$$



Also denoted D**f**(**x**), or d**f**/d**x**.

Note: components of **f** go vertically, those of **x** go horizontally.
Why? So that $\mathbf{f}(\mathbf{x}+\mathbf{h}) \approx \mathbf{f}(\mathbf{x}) + \mathbf{J}(\mathbf{x})\,\mathbf{h}$

Vector calculus and the Jacobian

The Jacobian generalizes the classical vector calculus operators: **gradient**, **divergence**, and **curl**.

- $f : \mathbb{R}^n \to \mathbb{R}^1$: Jacobian is a $1{\times}n$ matrix, i.e. a row vector

$$\mathrm{d}f/\mathrm{d}\mathbf{x} = [\partial f/\partial x_1 \quad \partial f/\partial x_2 \quad \cdots \quad \partial f/\partial x_n]$$

$$\nabla f = (\mathrm{d}f/\mathrm{d}\mathbf{x})^\mathsf{T}$$

- $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$: Jacobian is a square $n{\times}n$ matrix

$$\begin{aligned}\nabla \cdot \mathbf{f} &= \partial f_1/\partial x_1 + \partial f_2/\partial x_2 + \cdots + \partial f_n/\partial x_n \\ &= \mathrm{tr}(\mathrm{d}\mathbf{f}/\mathrm{d}\mathbf{x})\end{aligned}$$

- $\mathbf{f} : \mathbb{R}^3 \to \mathbb{R}^3$: Jacobian is a square $3{\times}3$ matrix
  - How can you get the curl?

Jacobian identities

Product rule:

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}(ab) = a\frac{\mathrm{d}b}{\mathrm{d}\mathbf{x}} + b\frac{\mathrm{d}a}{\mathrm{d}\mathbf{x}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}(a\mathbf{u}) = a\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{x}} + \mathbf{u}\frac{\mathrm{d}a}{\mathrm{d}\mathbf{x}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}(\mathbf{u}^T\mathbf{v}) = \mathbf{u}^T\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}\mathbf{x}} + \mathbf{v}^T\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{x}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}(\mathbf{M}\mathbf{v}) = \mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{x}}$$

(only if $\mathbf{M}$ is constant!)

Chain rule:

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}f(a) = f'(a)\frac{\mathrm{d}a}{\mathrm{d}\mathbf{x}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}\mathbf{f}(a) = \left(\frac{\mathrm{d}}{\mathrm{d}a}\mathbf{f}(a)\right)\frac{\mathrm{d}a}{\mathrm{d}\mathbf{x}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}f(\mathbf{u}) = \left(\frac{\mathrm{d}}{\mathrm{d}\mathbf{u}}f(\mathbf{u})\right)\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{x}},$$

$$\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}\mathbf{f}(\mathbf{u}) = \left(\frac{\mathrm{d}}{\mathrm{d}\mathbf{u}}f(\mathbf{u})\right)\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}\mathbf{x}}$$

Conveniently, all the matrices and vectors have the right shape to multiply together

What is $\mathrm{d}\mathbf{x}/\mathrm{d}\mathbf{x}$?

A worked example and an exercise

Prove that the gradient of $\|\mathbf{x}\|$ is $\mathbf{x}/\|\mathbf{x}\|$.

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}\|\mathbf{x}\| &= \frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}\sqrt{\mathbf{x}^T\mathbf{x}} \\
&= \frac{1}{2\sqrt{\mathbf{x}^T\mathbf{x}}} \cdot \frac{\mathrm{d}}{\mathrm{d}\mathbf{x}}(\mathbf{x}^T\mathbf{x}) \\
&= \frac{1}{2\|\mathbf{x}\|}(\mathbf{x}^T\mathbf{I} + \mathbf{x}^T\mathbf{I}) \\
&= \frac{1}{2\|\mathbf{x}\|}(2\mathbf{x}^T) \\
&= \frac{\mathbf{x}^T}{\|\mathbf{x}\|}.
\end{aligned}$$

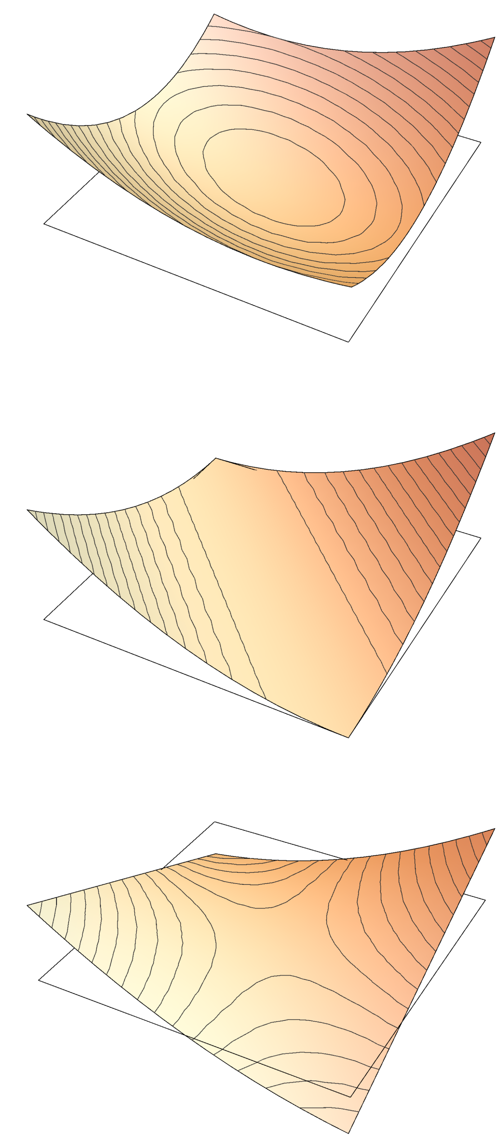
$$\begin{aligned}
\nabla\|\mathbf{x}\| &= \left(\frac{\mathrm{d}\|\mathbf{x}\|}{\mathrm{d}\mathbf{x}}\right)^T \\
&= \frac{\mathbf{x}}{\|\mathbf{x}\|}.
\end{aligned}$$

Exercise: Prove the Jacobian of $\mathbf{x}/\|\mathbf{x}\|$ is $(\mathbf{I} - (\mathbf{x}\mathbf{x}^T)/(\mathbf{x}^T\mathbf{x}))/\|\mathbf{x}\|$.

The Hessian

For a **scalar-valued** function $f : \mathbb{R}^n \to \mathbb{R}$, the matrix of second derivatives is called the **Hessian** (also denoted $D^2 f$ or $d^2 f / d\mathbf{x}^2$)

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



Tells you about the local "curvature" of f

Multivariate Taylor series:

$$f(\mathbf{x}+\mathbf{h}) = f(\mathbf{x}) + \nabla f(\mathbf{x}) \cdot \mathbf{h} + \tfrac{1}{2}\, \mathbf{h}^\top \mathbf{H}(\mathbf{x})\, \mathbf{h} + O(\|\mathbf{h}\|^3)$$

Implementation notes

We will need gradients, Jacobians, and Hessians fairly regularly. How to compute them?

- Whenever possible, derive **analytically**, then implement that
  - Use a CAS (Mathematica, Maxima, SageMath) if necessary
- Always **validate** with finite differences
  - Very easy to make a mistake in derivation / implementation! Very hard to find the bug later!
- Another option: **automatic differentiation** ("autodiff")
  - Library calculates derivatives for you! Tricky to use, slower to evaluate. Use if analytical formula is much too complicated

Outro: Applications to ODEs

Solving ODEs via finite differences

First-order ODE: $y'(t) = f(t, y)$,
initial conditions $y(t_0) = y_0$

We want to compute $y_1 = y(t_1)$, $y_2 = y(t_2)$, …

- Replace time derivative with forward difference:

$$(y_1 - y_0)/h = f(t_0, y_0)$$

$$\Rightarrow y_1 = y_0 + h\, f(t_0, y_0)$$ **Forward Euler**

- Backward difference $\Rightarrow y_1 = y_0 + h\, f(t_1, y_1)$ **Backward Euler**
    - This is an **implicit** method: must solve an equation to get y_1

Solving ODEs via quadrature rules

First-order ODE: $y'(t) = f(t, y)$,
initial conditions $y(t_0) = y_0$

Alternatively, interpret as integral $y_1 - y_0 = \int_{t_0}^{t_1} f(t, y(t))\,\mathrm{d}t$
and apply quadrature

- **(Implicit) midpoint method**: $y_1 - y_0 = h\, f(t_{½}, y_{½})$
  where $t_{½} = (t_0 + t_1)/2$, $y_{½} = (y_0 + y_1)/2$

- **Trapezoidal method**: $y_1 - y_0 = h/2\, (f(t_0, y_0) + f(t_1, y_1))$

Later we'll analyze the accuracy & stability of all these schemes

Next class

Mass-spring systems

Reading:

- Witkin and Baraff, *Physically Based Modeling*, Ch. “Differential Equation Basics” and “Particle Dynamics”
- Provot, “Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior”, 1995