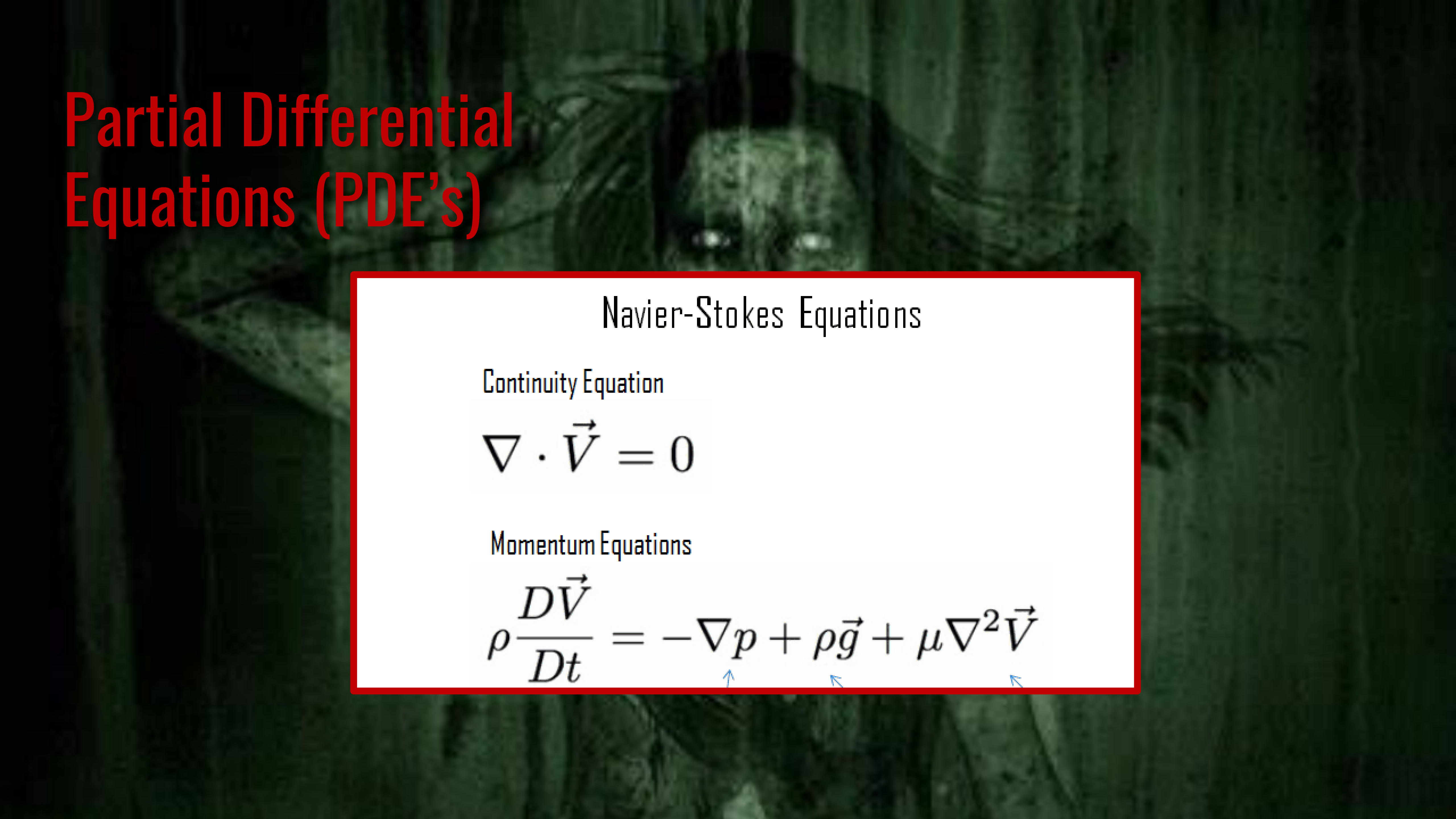


Partial Differential Equations (PDE’s)

Navier-Stokes Equations

Continuity Equation

$$\nabla \cdot \vec{V} = 0$$

Momentum Equations

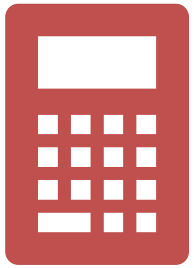
$$\rho \frac{D\vec{V}}{Dt} = -\nabla p + \rho \vec{g} + \mu \nabla^2 \vec{V}$$

Biomedical Engineering 5CCYB070

COMPUTATIONAL METHODS

Lecture 13 Vector Calculus and PDEs

Learning objectives



Intro to math tools to study PDEs

Some revision of calculus

Learn how PDEs can be represented and written

Appreciate the importance of auxiliary data for finding particular solutions of PDEs

Learn how PDEs can be categorized

Learn to solve PDEs by finite differences

Outline

Functions of multiple variables

- Partial derivatives / differentiation. Scalar / vector fields

Tools for differential calculus on scalar and vector fields

- Gradient / Divergence / Laplacian / Curl

Partial Differential Equations

- Categories of PDE / Order, linearity, homogeneity / Auxiliary data / ‘well-posedness’

Finite difference methods for solving a PDE

- Numerical estimates of derivatives / Forward Marching / Jacobi Iteration

Functions of multiple variables

- Temperature that depends on latitude and month of year

$$T = T(l, m)$$

- Charge that depends on location in 3-D

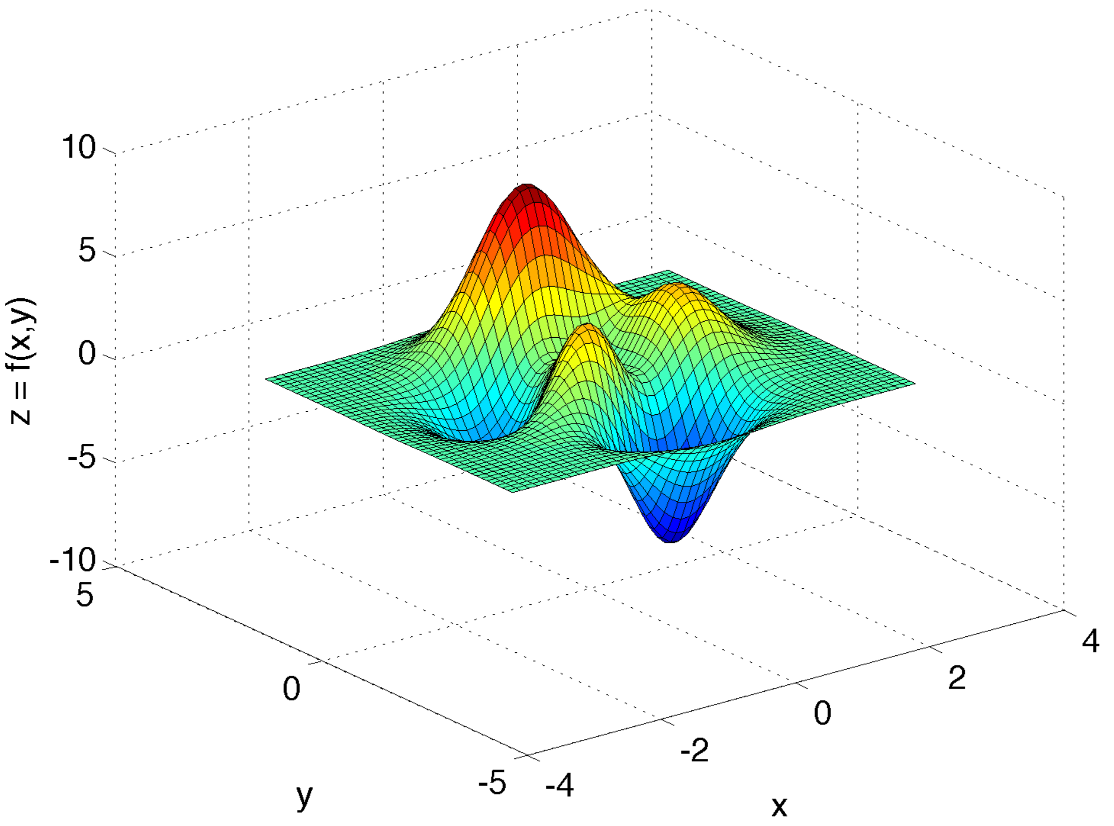
$$Q = Q(x, y, z)$$

- A surface plot of a function, height depends on x and y

$$z = f(x,y)$$

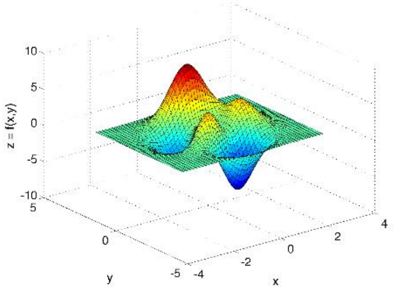
- E.g. *f* as the Matlab 'peaks' function

```
[x, y, z] = peaks()
surf(x, y, z)
```

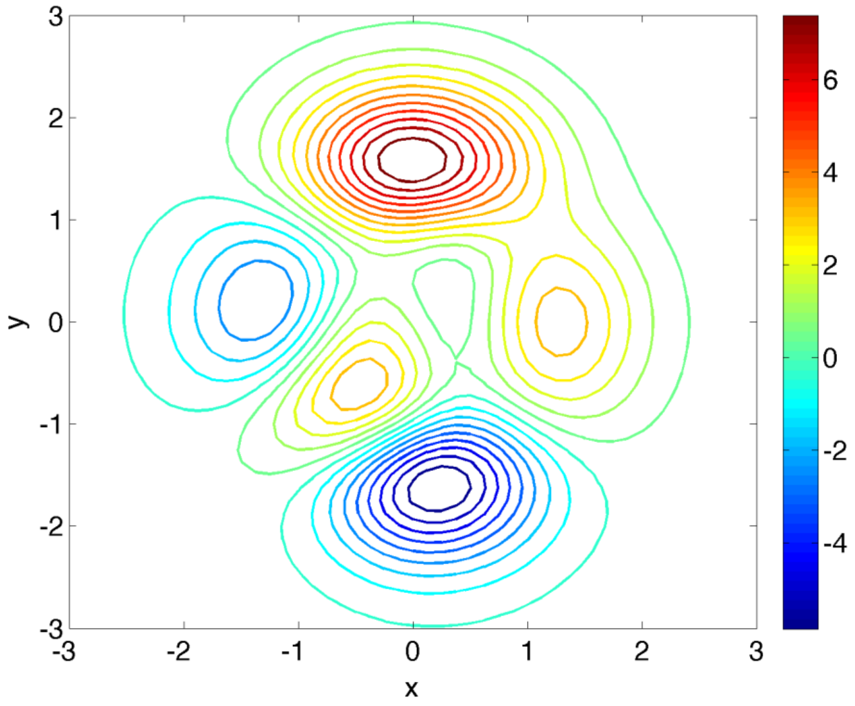


Visualising function of two variables

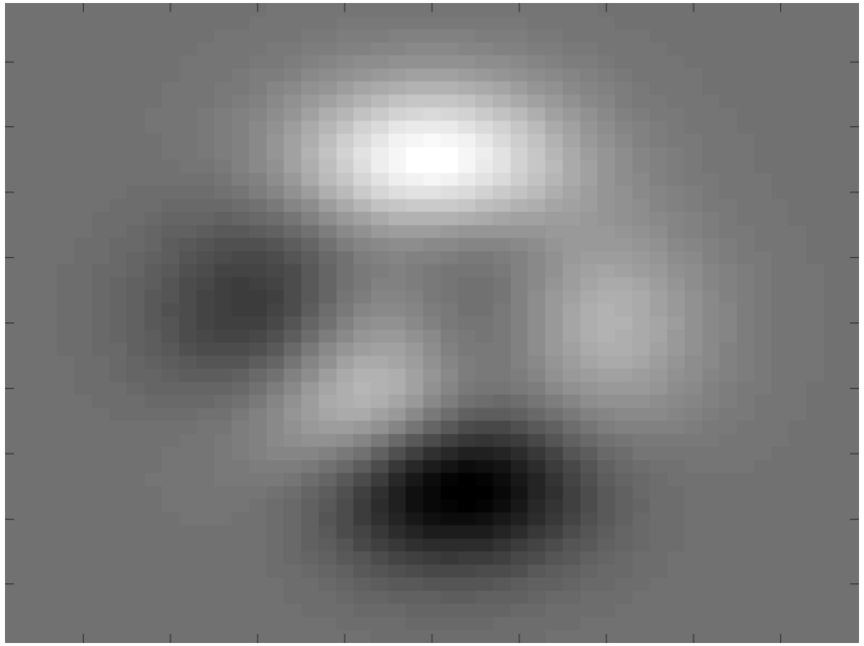
- Same peaks data, visualised in different ways …




```
[x,y,z] = peaks();
contour(x, y, z, 20)
```




```
[x,y,z] = peaks();
imagesc(z)
```



Multiple variables and differentiation

- Function of a single variable: Easy …

$$f(x) = 4 + \sin 2x \qquad \Rightarrow \qquad \frac{df}{dx} = 2\cos 2x$$

- Function of multiple variables: Choice to differentiate w.r.t* different variables:

$$Q = Q(x, y, z) \quad \rightarrow \quad \frac{\partial Q}{\partial x} \qquad \frac{\partial Q}{\partial y} \qquad \frac{\partial Q}{\partial z}$$

Three variables … three ***Partial** Derivatives*

Differentiation vs Partial differentiation

Straight 'd' vs Curly '∂'

** w.r.t. = with respect to*

Partial derivatives

- E.g. if we have $$T(l,m) = 2l + e^{-3l}\ \cos(5m)$$

$$\frac{\partial}{\partial l} \qquad \frac{\partial}{\partial m}$$

$$\frac{\partial T}{\partial l} = 2\ -\ 3e^{-3l}\ \cos(5m) \qquad \frac{\partial T}{\partial m} = -5e^{-3l}\ \sin(5m)$$

- There are two partial derivatives for this function, one for each variable
- More specifically, there are two **First-Order** partial derivatives

Higher order partial derivatives

$$f(x, y) = y^2 \sin x$$

$$\frac{\partial}{\partial x} \qquad\qquad \frac{\partial}{\partial y}$$

$$y^2 \cos x \qquad\qquad 2y \sin x$$

$$\frac{\partial f}{\partial x} \qquad\qquad \frac{\partial f}{\partial y}$$

Higher order partial derivatives

$$
\begin{tikzcd}
 & & & f(x,y) = y^2 \sin x \arrow[dll, "\frac{\partial}{\partial x}"'] \arrow[drr, "\frac{\partial}{\partial y}"] & & & \\
 & y^2 \cos x \arrow[dl, "\frac{\partial}{\partial x}"'] \arrow[dr, "\frac{\partial}{\partial y}"] & & & & 2y \sin x \arrow[dl, "\frac{\partial}{\partial x}"'] \arrow[dr, "\frac{\partial}{\partial y}"] & \\
-y^2 \sin x & & 2y \cos x & & 2y \cos x & & 2 \sin x
\end{tikzcd}
$$

$$\frac{\partial^2 f}{\partial x^2} \qquad \boxed{\frac{\partial^2 f}{\partial y \partial x} \qquad \frac{\partial^2 f}{\partial x \partial y}} \qquad \frac{\partial^2 f}{\partial y^2}$$

Mixed partials

Higher order partial derivatives

- More compact notation
  - Use subscripts to show what partial derivatives taken

- 'Well behaved' function: Equal mixed partial derivatives

$$f_{xx} \qquad f_{yx} = f_{xy} \qquad f_{yy}$$

$$\frac{\partial^2 f}{\partial x^2} \qquad \frac{\partial^2 f}{\partial y \partial x} = \frac{\partial^2 f}{\partial x \partial y} \qquad \frac{\partial^2 f}{\partial y^2}$$

Geometry in 2-D and 3-D

- Scalar: A quantity with no dimension
- Vector: Indicates a magnitude and direction
- Vectors: different numbers of components depending on dimension

2-D

$$\hat{i} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$\hat{j} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\alpha\,\hat{i} + \beta\hat{j}$$

$$x\,\hat{i} + y\hat{j}$$

$$2\,\hat{i} + 5\hat{j}$$

3-D

$$\hat{i} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\hat{j} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\hat{k} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

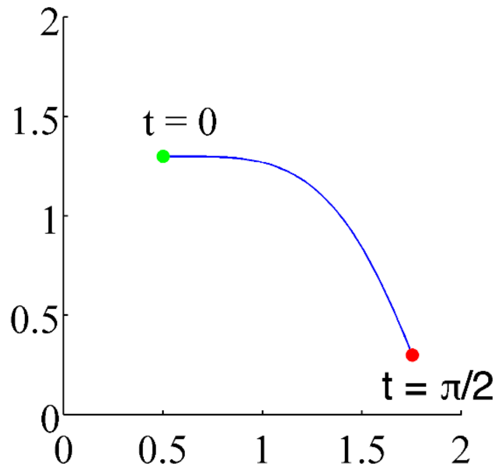
$$-3\,\hat{i} + 4\hat{k}$$

$$x\,\hat{i} + y\,\hat{j} + z\,\hat{k}$$

Vectors, scalars and functions

- Depending on inputs and outputs, we have a few combinations

Input: Scalar
Output: Vector



Curve

$$\vec{r}(t) = \begin{pmatrix} r_1(t) \\ r_2(t) \\ r_3(t) \end{pmatrix}$$

$$\vec{r}(t) = (r_1(t), r_2(t), r_3(t))^T$$

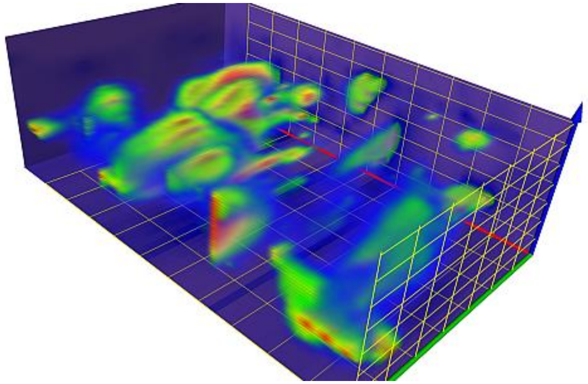
$$\vec{r}(t) = r_1(t)\hat{i} + r_2(t)\hat{j} + r_3(t)\hat{k}$$

```
t = linspace(0,pi/2, 50);
r1 = 0.5 + sqrt(t);
r2 = 0.3 + cos(t);
plot(r1, r2);
```

Vectors, scalars and functions

- Depending on inputs and outputs, we have a few combinations

Input: Vector
Output: Scalar

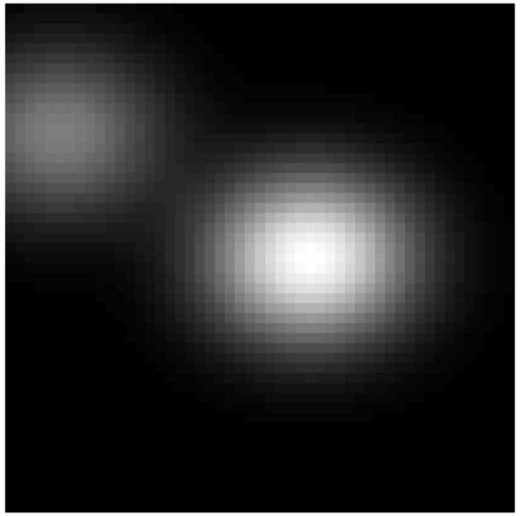


Scalar field

$$T = T(x, y, z)$$

$$T = T(\vec{x})$$

$$T = T(\vec{x}) \text{ where } \vec{x} = (x, y, z)^T$$

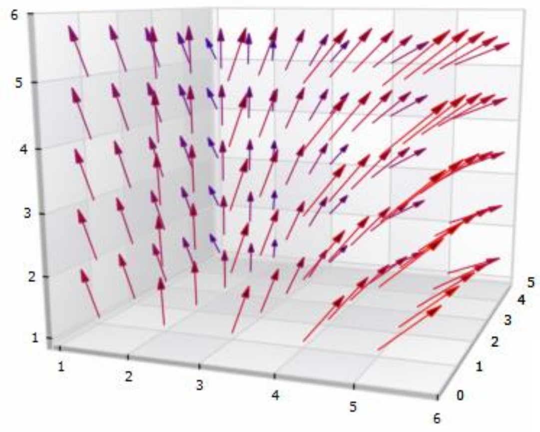


$$u = u(\vec{r}) \text{ where } \vec{r} = (x, y)^T$$

Vectors, scalars and functions

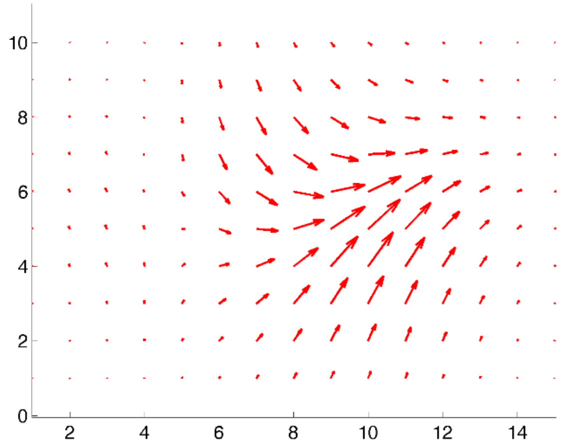
- Depending on inputs and outputs, we have a few combinations

Input: Vector
Output: Vector



Vector field

$$\vec{F} = \begin{pmatrix} F_1(x, y, z) \\ F_2(x, y, z) \\ F_3(x, y, z) \end{pmatrix}$$



$$\vec{F} = \vec{F}(\vec{x}) = \vec{F}(x, y, z)$$

$$\vec{F} = F_1(\vec{x})\,\hat{i} + F_2(\vec{x})\,\hat{j} + F_3(\vec{x})\,\hat{k}$$

$$\vec{F} = F_1(x, y, z)\,\hat{i} + F_2(x, y, z)\,\hat{j} + F_3(x, y, z)\,\hat{k}$$

Partial Differential Equations (PDE's)

Navier-Stokes Equations

Continuity Equation

$$\nabla \cdot \vec{V} = 0$$

Momentum Equations

$$\rho \frac{D\vec{V}}{Dt} = -\nabla p + \rho \vec{g} + \mu \nabla^2 \vec{V}$$

Outline

Functions of multiple variables

- Partial derivatives / differentiation. Scalar / vector fields

Tools for differential calculus on scalar and vector fields

- Gradient / Divergence / Curl / Laplacian

Partial Differential Equations

- Categories of PDE / Order, linearity, homogeneity / Auxiliary data / 'well-posedness'

Finite difference methods for solving a PDE

- Numerical estimates of derivatives / Forward Marching / Jacobi Iteration

Measures

There are measures to describe the behaviour of fields

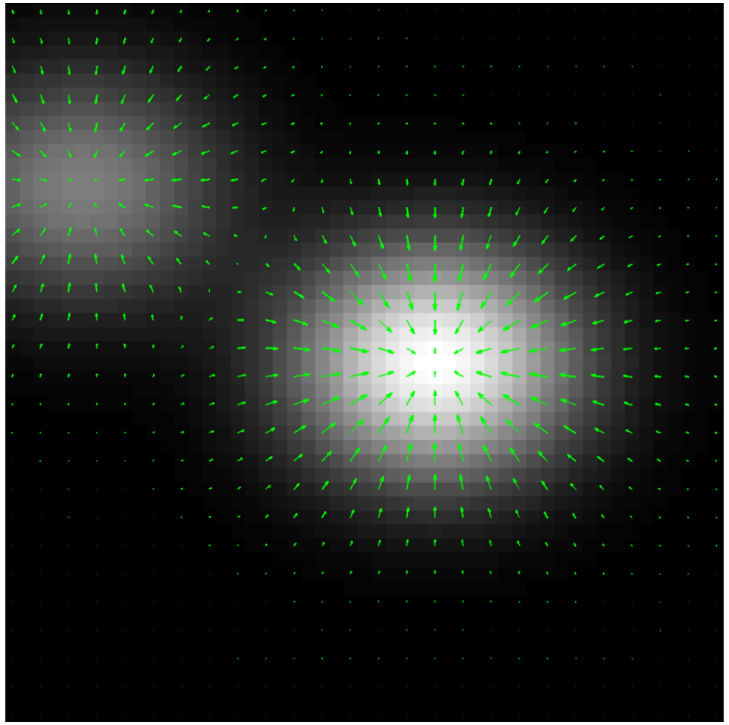
- Gradient of a scalar field
  - how much it changes in a particular direction
- Divergence of a vector field
  - how much ‘stuff’ is being created/destroyed
- Curl of a vector field
  - how rotation it ‘exerts’ at each location

Gradient of a scalar field

- For a given scalar field $\phi(x, y, z)$

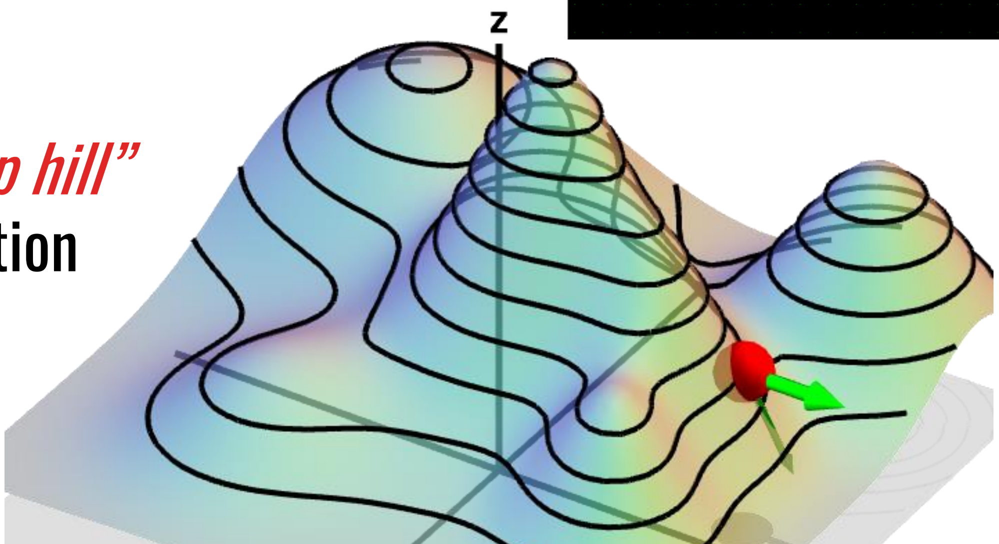
- The gradient gives a vector at each location

$$\mathrm{grad}\,\phi = \frac{\partial \phi}{\partial x}\,\hat{i} + \frac{\partial \phi}{\partial y}\,\hat{j} + \frac{\partial \phi}{\partial z}\,\hat{k}$$



Gradient has direction & magnitude

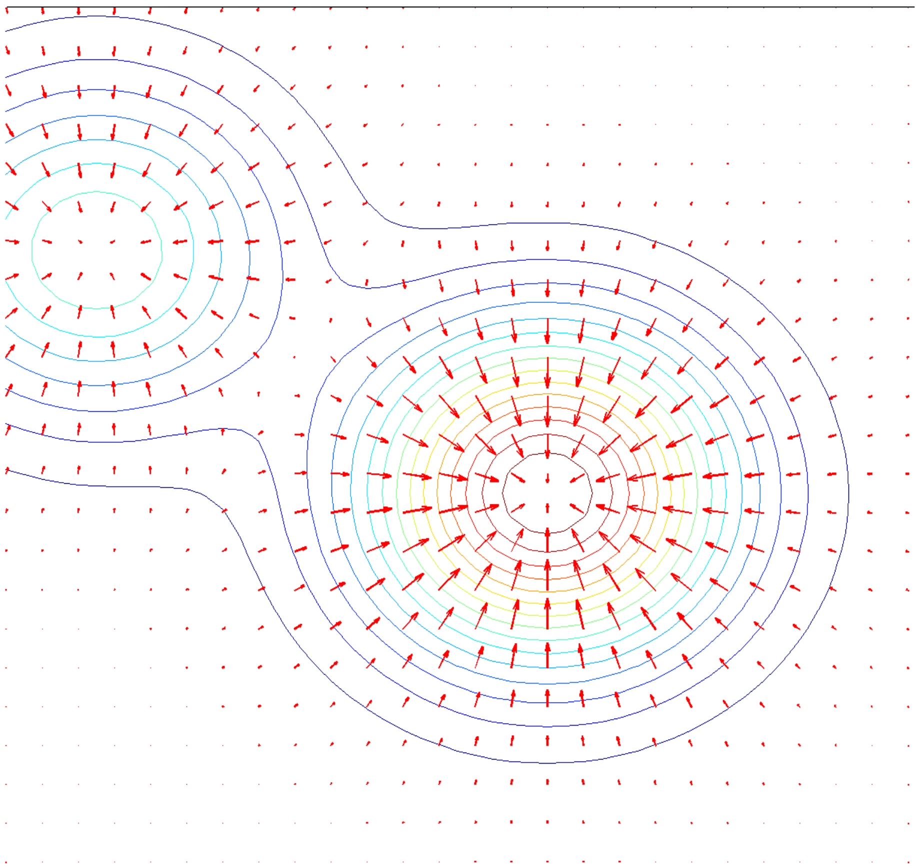
Direction: of **maximum** increase in φ *"up hill"*
Magnitude: rate of increase in that direction



Gradient of a scalar field

- For a given scalar field $\phi(x, y, z)$
- The gradient gives a vector at each location

$$\operatorname{grad}\phi = \frac{\partial \phi}{\partial x}\,\hat{i} + \frac{\partial \phi}{\partial y}\,\hat{j} + \frac{\partial \phi}{\partial z}\,\hat{k}$$



Cuts at right angles across

- the iso-lines in 2-D
- the iso-surfaces in 3-D

'flat' region: gradient = zero vector: $\vec{0}$

Gradient of a scalar field

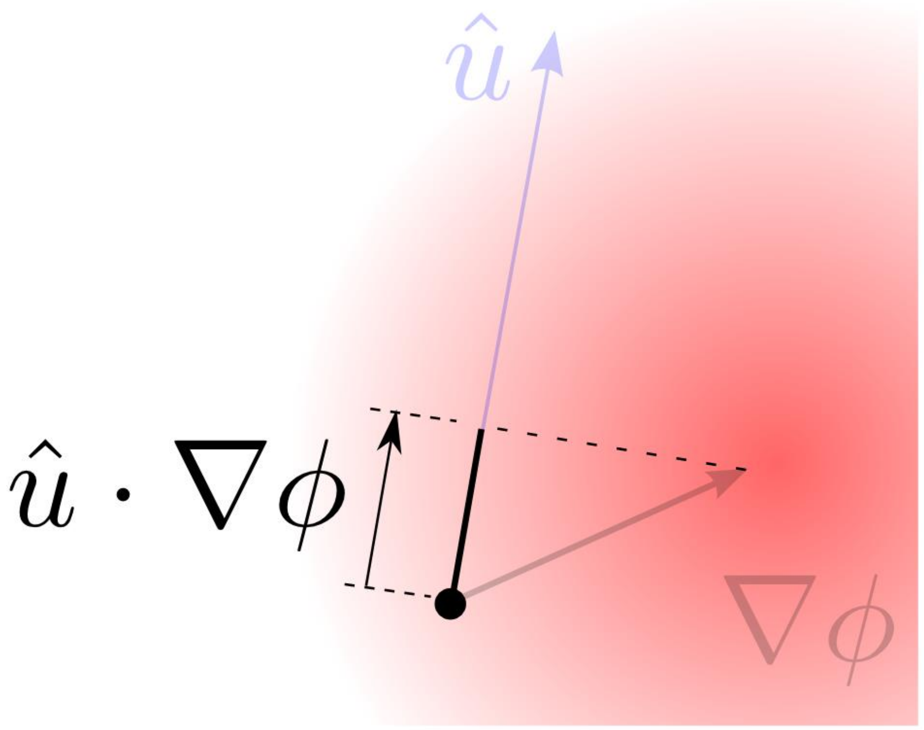
- Grad = rate and direction of **maximum** change $\mathrm{grad}\,\phi = \frac{\partial \phi}{\partial x}\,\hat{i} + \frac{\partial \phi}{\partial y}\,\hat{j} + \frac{\partial \phi}{\partial z}\,\hat{k}$
- We might want the rate of change in *another* direction (i.e. not just the maximum change direction)

  - For *any* given direction (blue unit vector)

$$\frac{d\phi}{ds} = \hat{u} \cdot \nabla \phi$$

scalar = vector · vector

(*s* is a distance unit)



Gradient of a scalar field

$$\mathrm{grad}\,\phi = \frac{\partial \phi}{\partial x}\,\hat{i} + \frac{\partial \phi}{\partial y}\,\hat{j} + \frac{\partial \phi}{\partial z}\,\hat{k}$$

Express in terms of an operator

$$\mathrm{grad}\,\phi = \left(\frac{\partial}{\partial x}\,\hat{i} + \frac{\partial}{\partial y}\,\hat{j} + \frac{\partial}{\partial z}\,\hat{k}\right)\phi$$

$$\mathrm{grad}\,\phi = \nabla\,\phi$$

'Del' or 'nabla' operator

$$\boxed{\nabla \longrightarrow \frac{\partial}{\partial x}\,\hat{i} + \frac{\partial}{\partial y}\,\hat{j} + \frac{\partial}{\partial z}\,\hat{k}}$$

- Gradient obtained by '*multiplying*' scalar field by Del
- Del is not a function, it *acts* on functions
- Can also act on vector fields as we will see later

Gradient of a scalar field

- Examples

2-D

$$\phi(x, y) = x \cos y$$

$$\nabla \phi(x, y) = \begin{pmatrix} \cos y \\ -x \sin y \end{pmatrix}$$

$$\nabla \phi(x, y) = (\cos y)\, \hat{i} - (x \sin y)\, \hat{j}$$

$$\nabla \phi = \hat{i} \cos y - \hat{j}\, x \sin y$$

All are equivalent

3-D

$$u(x, y, z) = x^2 + y e^z$$

$$\nabla u = 2x\, \hat{i} + e^z\, \hat{j} + y e^z\, \hat{k}$$

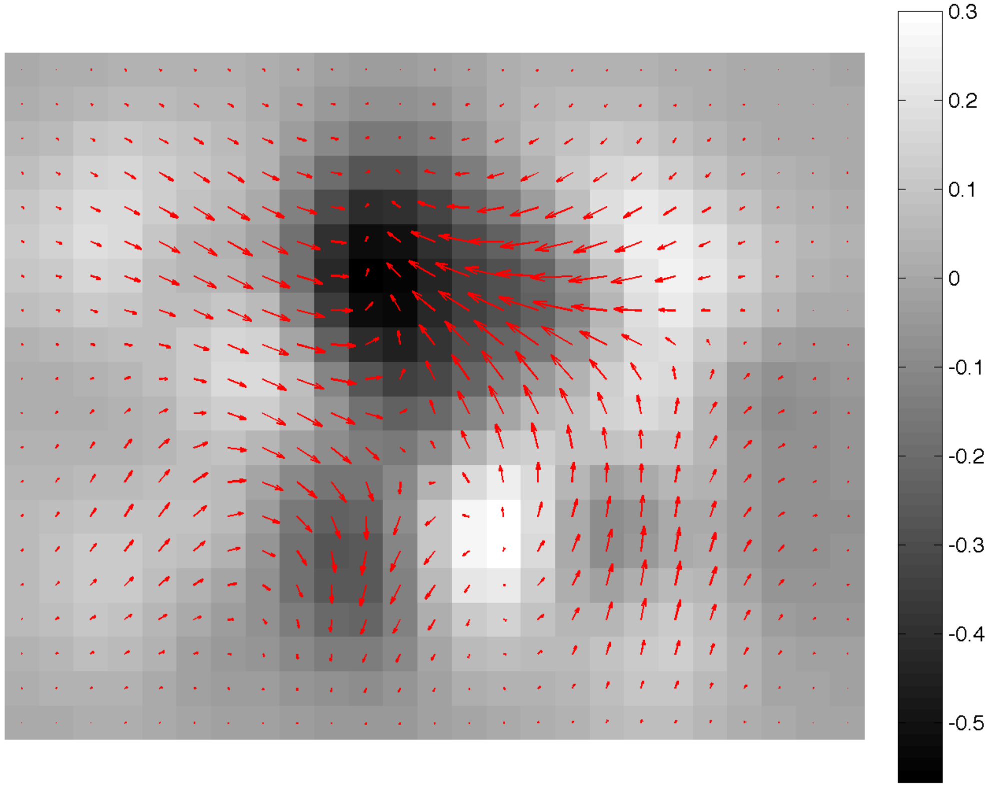
$$\nabla u = \begin{pmatrix} 2x\, \hat{i} \\ e^z\, \hat{j} \\ y e^z\, \hat{k} \end{pmatrix}$$

All are equivalent

Divergence of a vector field

- Measures how much ‘stuff ‘is being produced/destroyed at a location
  - Positive = Creation = Source
  - Negative = Destruction! = Sink

- Example visualised
  - **Input**: vector field, red arrows
  - **Output**: divergence of vector field, grey scale intensity



Divergence of a vector field

- Operator acting on a vector field

$$\vec{v} = \vec{v}(x, y, z) = [v_1(x, y, z), v_2(x, y, z), v_3(x, y, z)]^T$$

- Provides a scalar for every location

$$\underbrace{\operatorname{div} \vec{v}}_{\text{vector input}} = \underbrace{\frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v_3}{\partial z}}_{\text{scalar output}}$$

Divergence of a vector field

- Divergence of a vector field, expressed with the Del operator:

$$\boxed{\nabla \longrightarrow \operatorname{div} \vec{v} = \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v_3}{\partial z}}$$

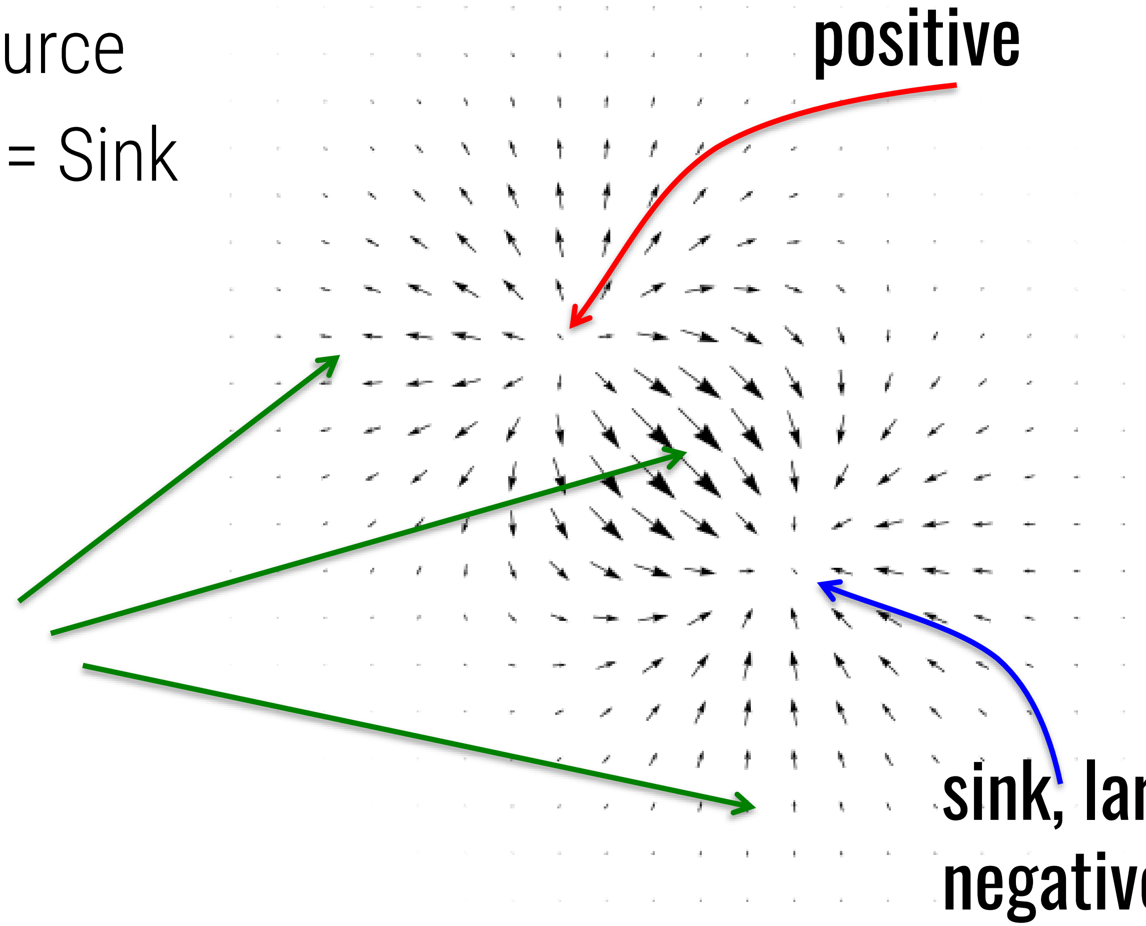
- I.e. divergence obtained by '*hitting*' the vector field on the left with Del and the dot product

$$\operatorname{div} \vec{v} = \nabla \cdot \vec{v} = \left(\frac{\partial}{\partial x}\,\hat{i} + \frac{\partial}{\partial y}\,\hat{j} + \frac{\partial}{\partial z}\,\hat{k}\right) \cdot \left(v_1\,\hat{i} + v_2\,\hat{j} + v_3\,\hat{k}\right) = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}$$

$$= \frac{\partial v_1}{\partial x} + \frac{\partial v_2}{\partial y} + \frac{\partial v_3}{\partial z}$$

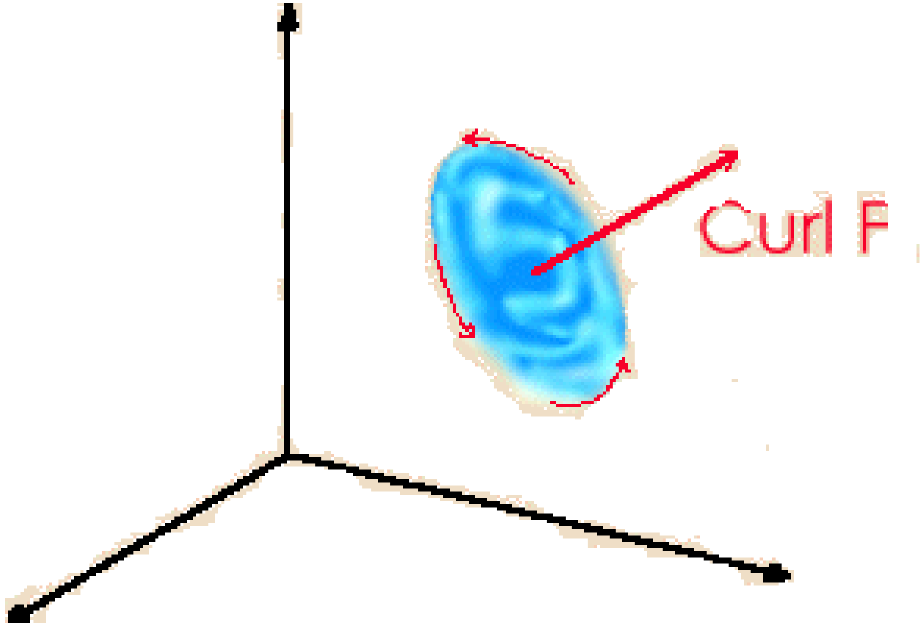
Divergence of a vector field

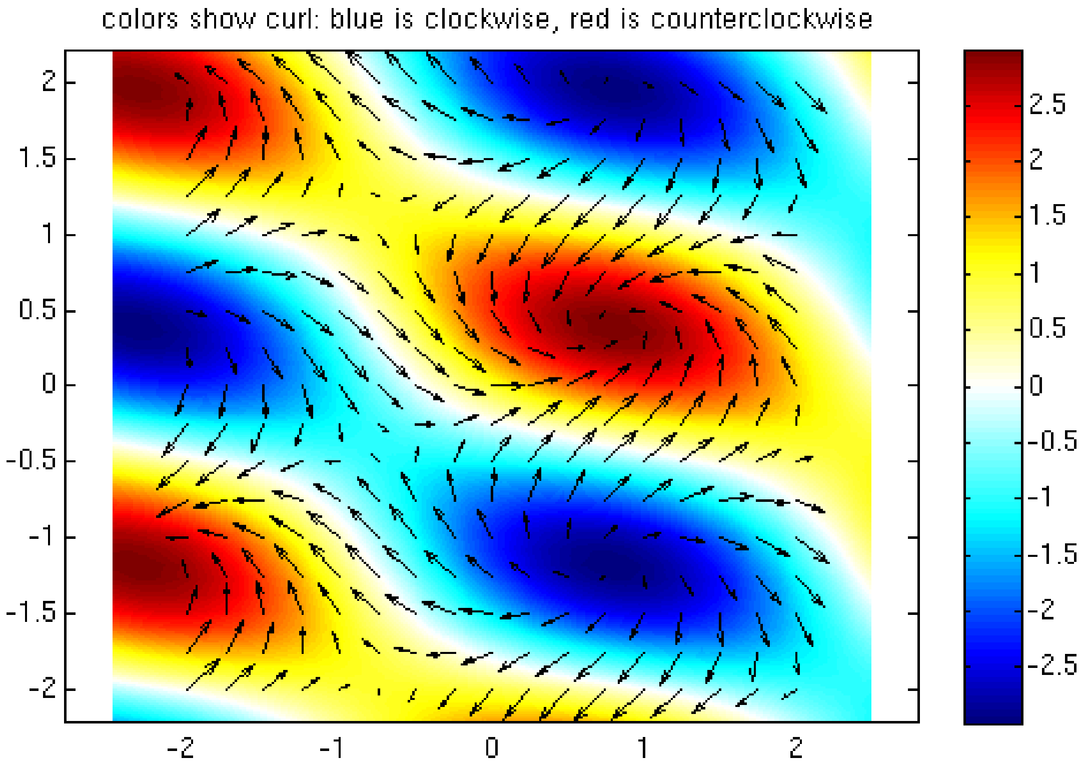
- Measures how much ‘stuff ‘is being produced/destroyed at a location
- ‘flux’
  - Positive = Creation = Source
  - Negative = Destruction! = Sink



Curl of a vector field

- Measures how much a vector field is 'turning'

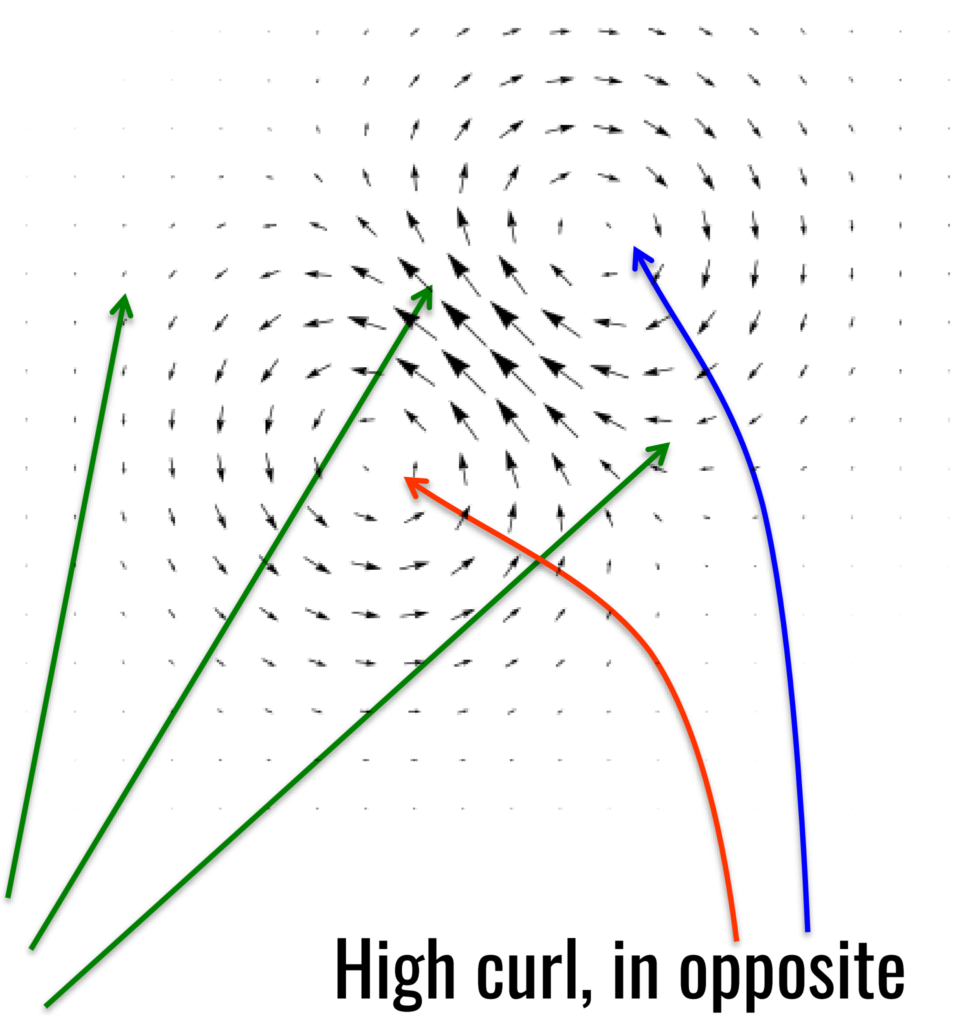




**Anti-clockwise, looking along axis, if we use a left-hand coordinate frame*

Curl of a vector field

- Measures how much a vector field is ‘turning’
- Viewing the vector field as a flow, it measures **how much a ‘small stick’ will tend to spin if dropped in**
- Represented by a vector
  - **Magnitude**: Amount of spin
  - **Direction**: Axis of spin*



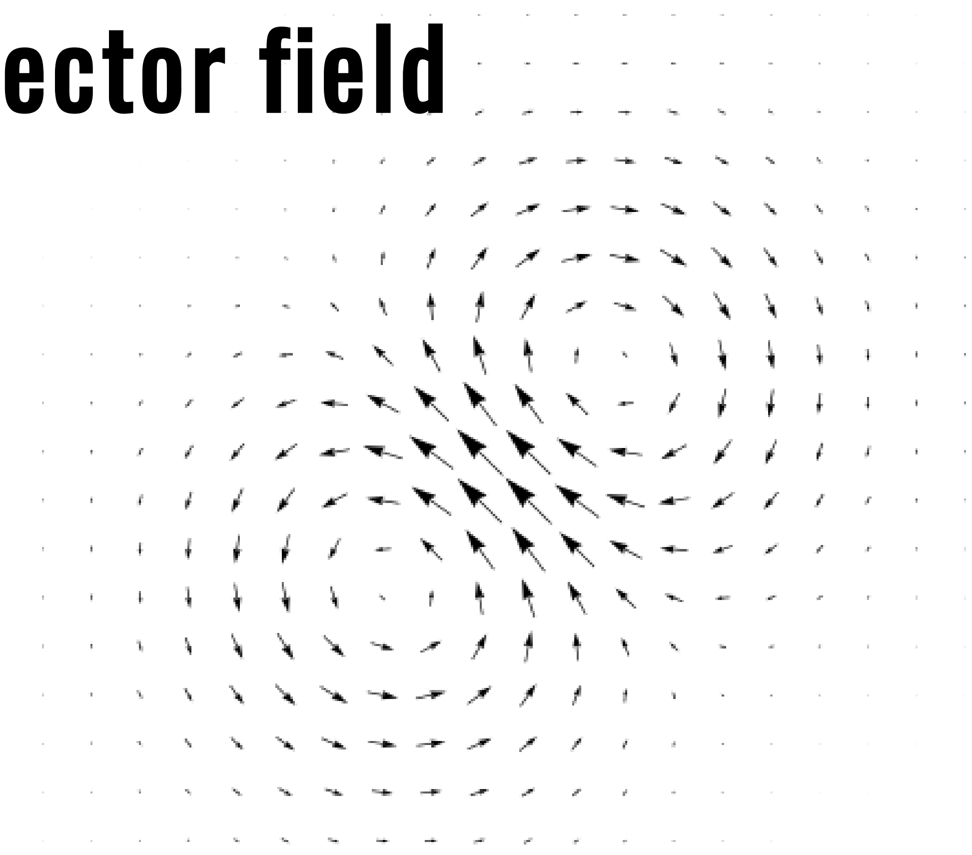
**Anti-clockwise, looking along axis, if we use a left-hand coordinate frame*

Curl of a vector field

- Defined using Del again!

$$\operatorname{curl} \vec{v} = \nabla \times \vec{v}$$

$$\operatorname{curl} \vec{v} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ v_1 & v_2 & v_3 \end{vmatrix}$$



$$\operatorname{curl} \vec{v} = \underbrace{\left(\frac{\partial v_3}{\partial y} - \frac{\partial v_2}{\partial z}\right)\hat{i} + \left(\frac{\partial v_1}{\partial z} - \frac{\partial v_3}{\partial x}\right)\hat{j} + \left(\frac{\partial v_2}{\partial x} - \frac{\partial v_1}{\partial y}\right)\hat{k}}_{\text{vector output}}$$

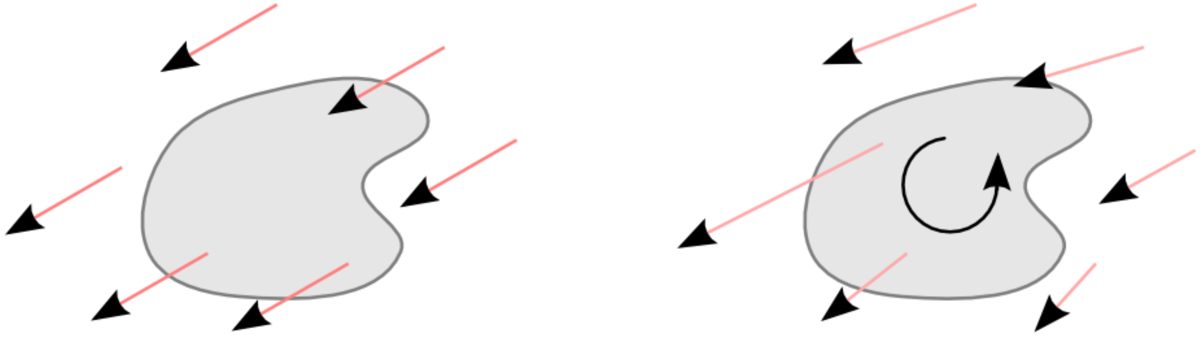
vector input

**Anti-clockwise, looking along axis, if we use a left-hand coordinate frame*

Curl of a vector field

- Illustration of curl affecting a small object in a flow

$$\nabla \times \vec{v} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ v_1 & v_2 & v_3 \end{vmatrix}$$



$\nabla \times \vec{v} = \vec{0}$ $\qquad$ $\nabla \times \vec{v} \neq \vec{0}$

Note subtle differences between vector fields

Combining Scalar and Vector Fields

- Given two scalar fields and two vector fields

$$\phi, \psi, \vec{a}, \vec{b}$$

New scalar fields

New vector fields

$$\phi + \psi \quad \phi\psi \quad \vec{a}\cdot\vec{b} \qquad \phi\vec{a} \quad \vec{a}+\vec{b} \quad \vec{a}\times\vec{b}$$

grad

$$\nabla(\phi + \psi)$$
$$\nabla(\phi\psi)$$
$$\nabla\left(\vec{a}\cdot\vec{b}\right)$$

div

$$\nabla\cdot(\phi\vec{a})$$
$$\nabla\cdot\left(\vec{a}+\vec{b}\right)$$
$$\nabla\cdot\left(\vec{a}\times\vec{b}\right)$$

curl

$$\nabla\times(\phi\vec{a})$$
$$\nabla\times\left(\vec{a}+\vec{b}\right)$$
$$\nabla\times\left(\vec{a}\times\vec{b}\right)$$

Combining Scalar and Vector Fields

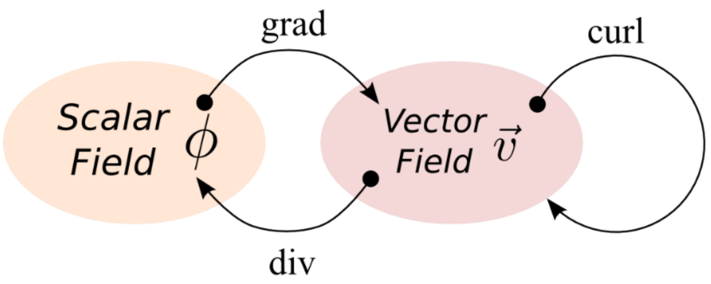
- Express div/grad/curl of combination in terms of div/grad/curl of individual fields
- E.g.

$$\underbrace{\nabla \cdot (\phi\, \vec{a})}_{\text{Divergence of a new vector field (We expect a scalar field)}} = \underbrace{\nabla\phi \cdot \vec{a}}_{\text{Dot product of a vector field (grad) and another vector field}} + \underbrace{\phi\, \nabla \cdot \vec{a}}_{\text{A scalar field multiplied by another (divergence)}}$$

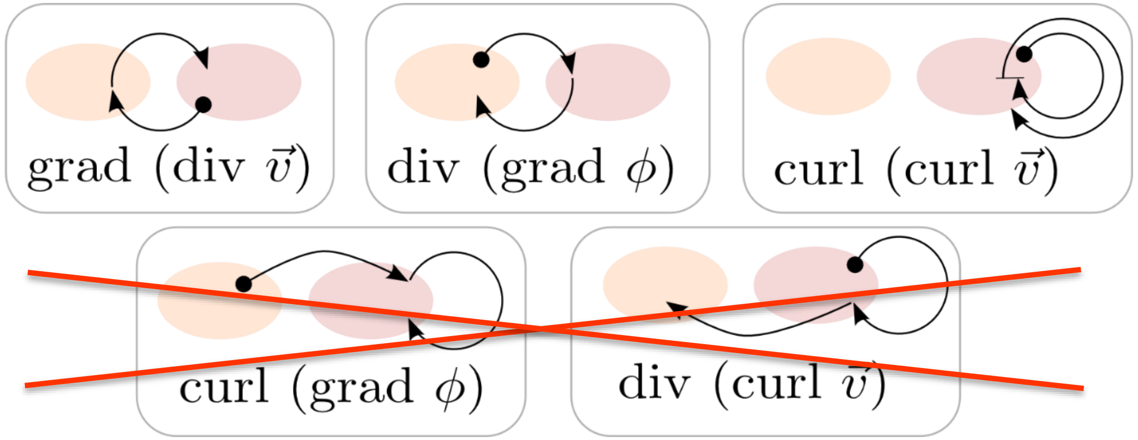
- Proof in notes, along with expressions for other combinations

Combining the div, grad and curl operators

- Rules on inputs and outputs
- Only five combinations possible



$\nabla(\nabla \cdot \vec{v})$ $\quad$ $\nabla \cdot (\nabla \phi)$ $\quad$ $\nabla \times (\nabla \times \vec{v})$



$\nabla \times (\nabla \vec{v})$ $\quad$ $\nabla \cdot (\nabla \times \vec{v})$

$\downarrow$ $\quad$ $\downarrow$

$\vec{0}$ $\quad$ $\vec{0}$

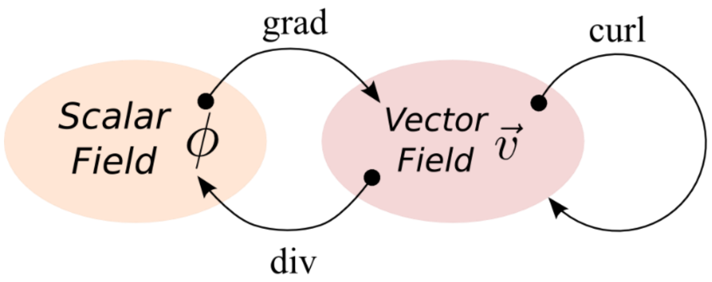
Two combinations are always zero regardless of input field

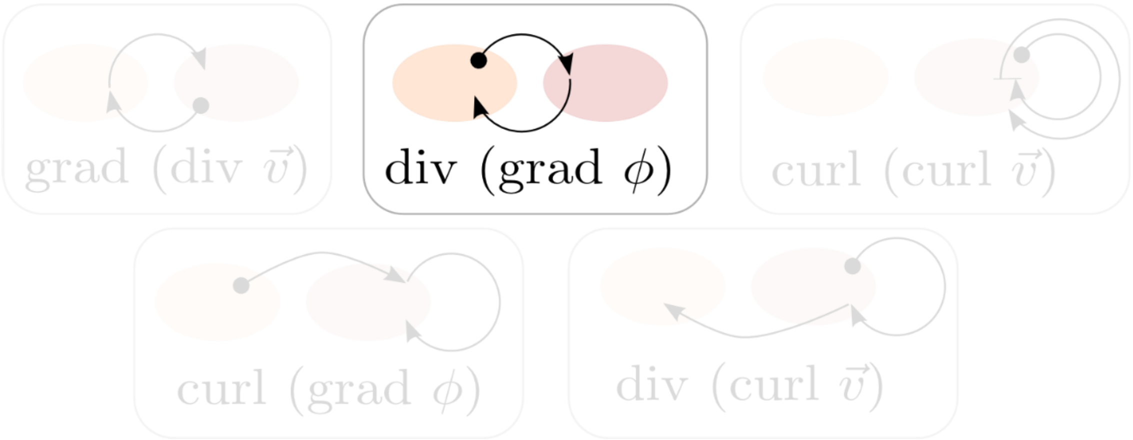
Combining the div, grad and curl operators

One **special** combination, divergence of gradient

- Turns up a lot!
- Gets its own name / symbol

$$\nabla \cdot (\nabla \phi)$$





$$\nabla \cdot (\nabla \phi) \quad = \quad \nabla^2 \phi \quad = \quad \Delta \phi$$

The Laplacian of the scalar field φ

Partial Differential Equations (PDE's)

$$\rho g_x - \frac{\partial p}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) = \rho\frac{Du}{Dt}$$

$$\rho g_y - \frac{\partial p}{\partial y} + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) = \rho\frac{Dv}{Dt}$$

$$\rho g_z - \frac{\partial p}{\partial z} + \mu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) = \rho\frac{Dw}{Dt}$$

$$\rho\frac{D\vec{V}}{Dt} = -\nabla p + \rho\vec{g} + \mu\nabla^2\vec{V}$$

Outline

Functions of multiple variables

- Partial derivatives / differentiation. Scalar / vector fields

Tools for differential calculus on scalar and vector fields

- Gradient / Divergence / Laplacian / Curl

Partial Differential Equations

- Categories of PDE / Order, linearity, homogeneity / Auxiliary data / ‘well-posedness’

Finite difference methods for solving a PDE

- Numerical estimates of derivatives / Forward Marching / Jacobi Iteration

Partial Differential Equations

- ODEs
  - Function of a single variable **y** = **y** (**t**)
  - Usual derivative, has a 'straight d'

$$y\frac{dy}{dt} = t^3 + \sin t$$

- In a PDE we seek a function of *more than one variable*

$$2s\frac{\partial u}{\partial s} - t\frac{\partial u}{\partial t} = 0 \quad \Rightarrow \quad u(s,t) = ?$$

Partial Derivatives are represented by **Curly** ∂'s

Partial Differential Equations

We say

$$\text{find } u(s,t) \text{ where } 2s\frac{\partial u}{\partial s} - t\frac{\partial u}{\partial t} = 0$$

- is a *first-order PDE*
  - All partial derivatives are first order
  - No terms with second order derivatives, e.g. $\dfrac{\partial^2 u}{\partial s^2}$ $\dfrac{\partial^2 u}{\partial s \partial t}$
  - And no third, fourth, … or higher order derivatives, etc.

Partial Differential Equations

We say

$$\text{find } u(s,t) \text{ where } 2s\frac{\partial u}{\partial s} - t\frac{\partial u}{\partial t} = 0$$

- is a *linear PDE*
  - No powers greater than 1 **for the *u* terms**

$$\boxed{e^t\frac{\partial u}{\partial s} - t^3\frac{\partial u}{\partial t} + u = st}$$

$$\boxed{\cos t\,\frac{\partial u}{\partial s} - t\frac{\partial u}{\partial t} + u^3 = 0}$$

$$\boxed{\frac{\partial u}{\partial s}\frac{\partial u}{\partial t} - 3u = 0}$$

$$\boxed{s\frac{\partial u}{\partial t} = u}$$

$$\boxed{s\left(\frac{\partial u}{\partial s}\right)^2 - t\frac{\partial u}{\partial t} = 0}$$

Linear PDE

Non-Linear PDE

First order linear **PDE**

- General form when *u* has **two** variables

$$\boxed{A\frac{\partial u}{\partial s} + B\frac{\partial u}{\partial t} + Cu + D = 0}$$

- Where we are given

$$A = A(s,t) \quad B = B(s,t) \quad C = C(s,t) \quad D = D(s,t)$$

- E.g. $2s\frac{\partial u}{\partial s} - t\frac{\partial u}{\partial t} = 0$

$$u_s - 3u_t = se^t$$

First order linear PDE

- General form when *u* has two variables

$$\boxed{A\frac{\partial u}{\partial s} + B\frac{\partial u}{\partial t} + Cu + D = 0}$$

- All 'coefficients' directly involve *u* except for *D*. If *D* = 0, we say that the PDE is *homogeneous.* Otherwise, *non-homogeneous*

E.g *homogeneous* *non-homogeneous*

$$2su_s - tu_t = 0 \qquad u_s - 3u_t = se^t$$

$$D = 0 \qquad D = -se^t$$

First order linear PDE

- What if *u* has more variables? E.g. $u = u\,(\,x, s, t\,)$ ... General form of 1st order linear PDE becomes

$$\boxed{A\frac{\partial u}{\partial x} + B\frac{\partial u}{\partial s} + C\frac{\partial u}{\partial t} + Du + E = 0}$$

- Homogeneity depends on E
- Readily extended to more variables …

Second-order linear PDEs

- If we have a function f of two variables x and y

$$A\frac{\partial^2 f}{\partial x^2} + B\frac{\partial^2 f}{\partial x \partial y} + C\frac{\partial^2 f}{\partial y^2} + D\frac{\partial^2 f}{\partial y \partial x} + E\frac{\partial f}{\partial x} + F\frac{\partial f}{\partial y} + G\, f + H = 0$$

- Or $\boxed{A\, f_{xx} + B\, f_{xy} + C\, f_{yy} + D\, f_{yx} + E\, f_x + F\, f_y + G\, f + H = 0}$

- Most commonly mixed partials are equal. So only need one of f_{xy}, f_{yx}

$$Af_{xx} + Bf_{xy} + Cf_{yy} + Df_x + Ef_y + Ff + G = 0$$

- Once again, $A, \ldots, G$ can be functions of x and y

PDE examples: Categorisation

- Examples : *Order* / *Linearity* / *Homogeneity*

$$\text{find } u(x,t) \text{ where } u_t + 3u\, u_x = 0$$

First-order, non-linear and homogeneous

$$\text{find } f(x,y) \text{ where } f_{xy} + f_x \sin x - e^y = 0$$

Second-order, linear and non-homogeneous

Principle of superposition

- In homogeneous linear PDE's, for example: $u_t - 3u_x = 0$
- $f = \alpha v + \beta w$

  is a solution

$$
\boxed{
\begin{aligned}
f_t - 3f_x &= \frac{\partial}{\partial t}(\alpha v + \beta w) - 3\frac{\partial}{\partial x}(\alpha v + \beta w) \\
&= \alpha v_t + \beta w_t - 3(\alpha v_x + \beta w_x) \\
&= \alpha(v_t - 3v_x) + \beta(w_t - 3w_x) \\
&= \alpha \times 0 + \beta \times 0 \\
&= 0
\end{aligned}
}
$$

- We have 'superposed' *u* and *w* to get *f*
- **Important reason for focusing on homogeneous linear PDEs**

Auxiliary conditions: ODEs

- Return to an Ordinary Differential Equation (ODE) example:

$$\frac{du}{dt} = 3u\,, \quad t > 0$$

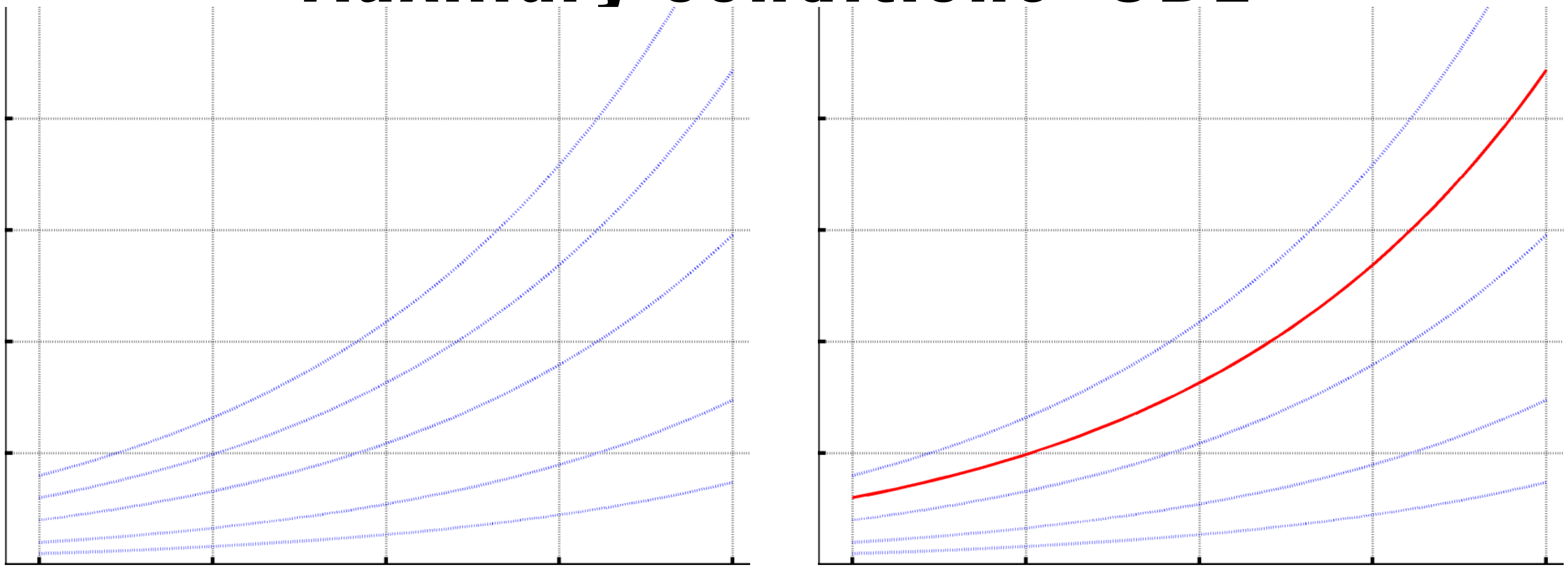
- There are many solutions to this

$$u = e^{3t} \qquad u = 5\,e^{3t} \qquad u = -1.2\,e^{3t}$$

- Generally, we have a '*family of solutions*'

$$u = \alpha\, e^{3t} \text{ for } \alpha \in \mathbb{R}$$

Auxiliary conditions: ODE



$$u = \alpha\, e^{3t} \text{ for } \alpha \in \mathbb{R}$$

- Can fix a **particular** solution if we have an *initial condition*

$$\text{find } u(t) \text{ where } \frac{du}{dt} = cu\,, \quad t > 0 \text{ and } \boxed{u(0) = 3}$$

Auxiliary conditions: PDE

Now consider a simple PDE

$$\boxed{u_t = x \cos t}$$

- Solve for ***u*(*x*,*t*)** by integrating…
- Get an arbitrary '*constant*' of integration – any function!

$$\int u_t \, dt = \int x \cos t \, dt \Rightarrow \quad u(x,t) = x \sin t \boxed{+ \phi(x)}$$

- Must not depend on *t* . . . but it *can* depend on *x*. As before, many possible solutions to the PDE

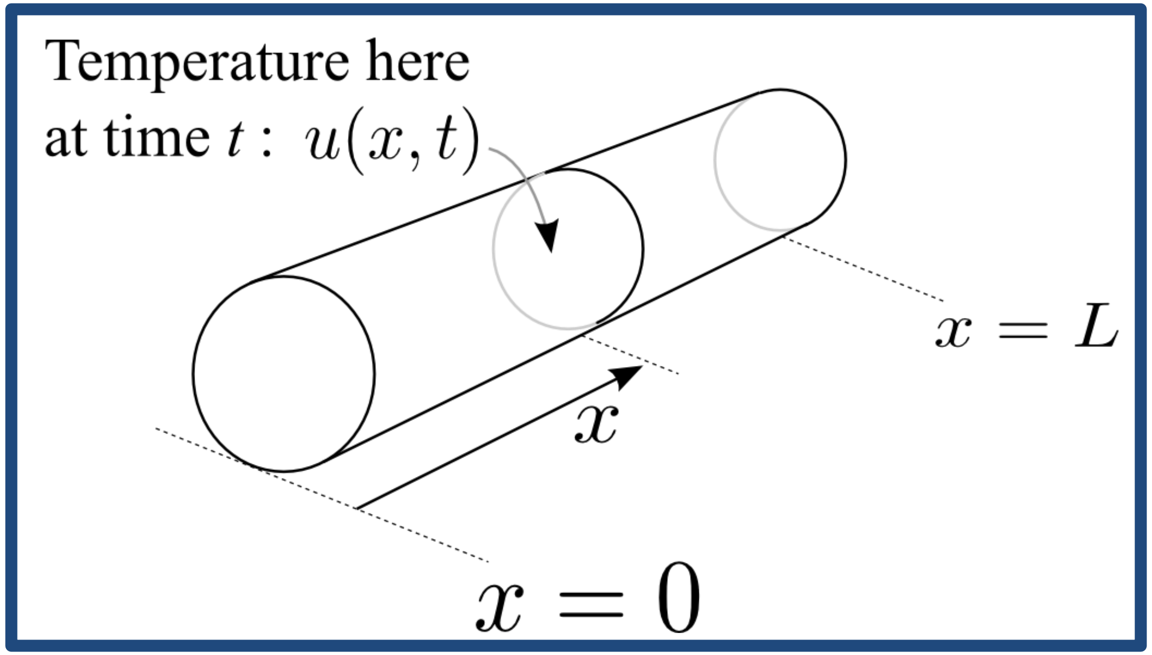
$$u_t = \frac{\partial}{\partial t}(x \sin t + \phi(x)) = \frac{\partial}{\partial t} x \sin t + \frac{\partial}{\partial t}\phi(x) = x \cos t + 0 = x \cos t$$

Example PDE

- 1-D heat equation

$$\boxed{u_t = k u_{xx}}$$

- Second-order, linear, homogeneous
- Two possible solutions (out of many! You can confirm them)
  - *Note how different they are*



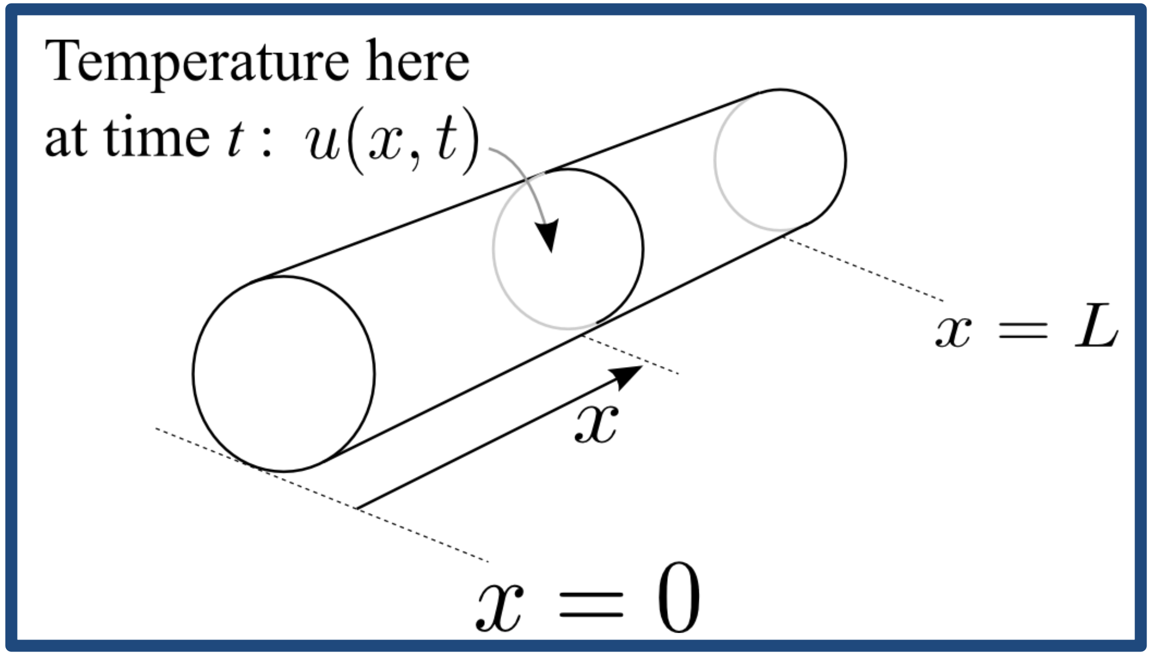
$$u_1(x,t) = x^2 + 2t \qquad \text{and} \qquad u_2(x,t) = e^{-t}\sin x$$

Example PDE

- 1-D heat equation

$$\boxed{u_t = k u_{xx}}$$

- Need *auxiliary conditions* / *auxiliary data* to fix a solution
- Different types of auxiliary data:
  - Boundary conditions $\quad u(0,t) = 0, \qquad u(L,t) = 0, \qquad t > 0$
  - Initial conditions $\quad u(x,0) = f(x)$



A 'Well-posed' PDE

- We say that the heat equation, with auxiliary conditions, is a **well-posed problem**

$$\boxed{\begin{aligned} u_t &= k u_{xx} \\ u(0,t) &= 0, \qquad u(L,t) = 0,\ t > 0 \\ u(x,0) &= f(x), \qquad \text{for } 0 < x < L \end{aligned}}$$

…Means

  - We can find a specific solution to the problem
  - The solution is 'well behaved', small changes in the data (auxiliary conditions) lead to small changes in the solution

- Much of subject concerns which combinations of PDE and auxiliary data lead to a well-posed problem

A 'Well-posed' PDE

- We say that the heat equation, with auxiliary conditions, is a well-posed problem

$$
\boxed{\begin{aligned}
u_t &= k u_{xx} \\
u(0,t) &= 0, \qquad u(L,t) = 0,\ t > 0 \\
u(x,0) &= f(x), \qquad \text{for } 0 < x < L
\end{aligned}}
$$

- It can be described in terms of its auxiliary data as a *Boundary Initial Value Problem* (Or an *Initial Boundary Value Problem*)
- Link for further description of well-posed PDEs with examples

  http://www.phy.ornl.gov/csep/pde/node6.html

Types of Boundary condition

Dirichlet boundary conditions:

- Specify the *value* of the function at each point on the boundary
  - Laplace Eqn.

$$\text{find } u \text{ where } \Delta u = 0, \ \text{for } (x,y,z) \in \Omega$$

$$\boxed{\text{subject to } u(x,y,z) = f(x,y,z), \ \text{for } (x,y,z) \in \partial\Omega}$$

Neumann boundary conditions:

- Specify the component of the gradient across the boundary
  - Laplace Eqn.

$$\text{find } u \text{ where } \Delta u = 0, \ \text{for } (x,y,z) \in \Omega$$

$$\boxed{\text{subject to } \frac{\partial u}{\partial n} = g(x,y,z), \ \text{for } (x,y,z) \in \partial\Omega}$$

Types of Boundary condition

Cauchy boundary conditions:

- Both value of the function ***and*** the gradient at each point on the boundary
  - E.g. solve in *u* subject to $$\boxed{\begin{aligned} u(\vec{x}) &= f(\vec{x}) \\ \frac{\partial u(\vec{x})}{\partial n} &= g(\vec{x}) \end{aligned} \quad \text{for} \quad \vec{x} \in \partial\Omega}$$

Robin boundary conditions:

- Specify a ***linear combination*** of the value and gradient.
  - E.g. solve a PDE in *u* subject to $$\boxed{\alpha u(\vec{x}) + \beta \frac{\partial u(\vec{x})}{\partial n} = h(\vec{x}) \quad \text{for} \quad \vec{x} \in \partial\Omega}$$

Second-order linear PDEs

- Used to model a wide variety of situations, e.g. both recent examples (Heat Equation, Laplace Equation)
- Have their own categories - based on coefficient functions

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial x \partial y} + C\frac{\partial^2 u}{\partial y^2} + D\frac{\partial u}{\partial x} + E\frac{\partial u}{\partial y} + Fu + G = 0$$

- Define *discriminant* $\boxed{B^2 - 4AC}$

A 'SOL-PDE' is

- Hyperbolic if $B^2 - 4AC > 0$
- Parabolic if $B^2 - 4AC = 0$
- Elliptic if $B^2 - 4AC < 0$

Remember, *A*, *B* and *C* are functions, so we can have a PDE that is in different categories in different regions of the (*x*,*y*) plane

Second-order linear PDEs

$$A\frac{\partial^2 u}{\partial x^2} + B\frac{\partial^2 u}{\partial x \partial y} + C\frac{\partial^2 u}{\partial y^2} + D\frac{\partial u}{\partial x} + E\frac{\partial u}{\partial y} + Fu + G = 0$$

A 'SOL-PDE' is

- Hyperbolic if $B^2 - 4AC > 0$
- Parabolic if $B^2 - 4AC = 0$
- Elliptic if $B^2 - 4AC < 0$

Category determines type of boundary condition needed to ensure PDE is Well-Posed

Equation Type	Conditions
Hyperbolic	Cauchy
Parabolic	Dirichlet or Neumann
Elliptic	Dirichlet or Neumann

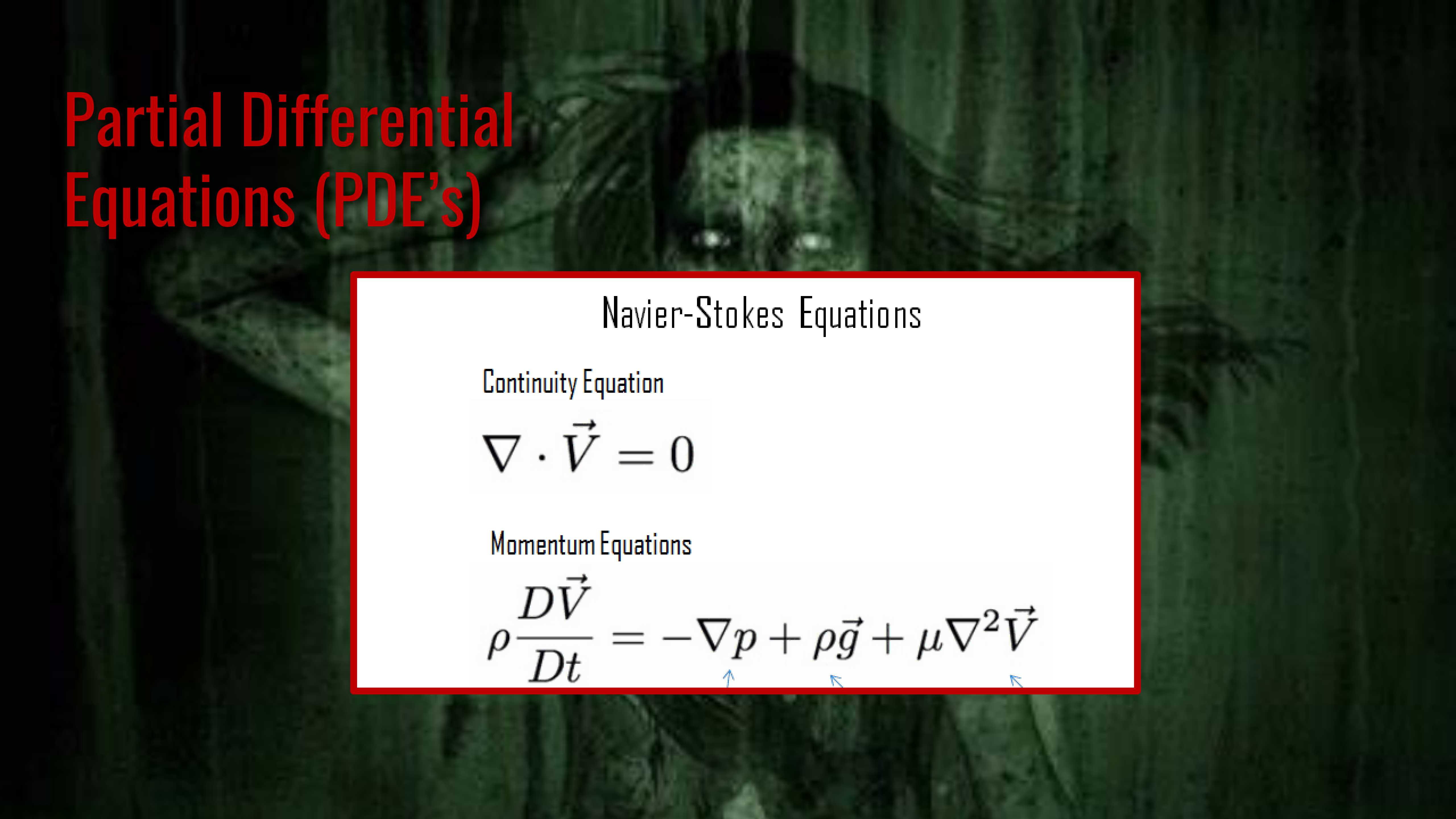
PDE categorisations: Recap

- Linear vs. non-linear
- First-order, second-order, …
- Homogeneous vs. Non-homogeneous

- Second order linear PDEs
  - hyperbolic
  - parabolic
  - elliptic

- Auxiliary data
  - Initial conditions
  - Boundary conditions
    - Dirichlet/Neumann
    - Robin/Cauchy

- Well-posed (vs. ill-posed)
- Depends on type of PDE & nature of auxiliary data



Partial Differential Equations (PDE's)

Navier-Stokes Equations

Continuity Equation

$$\nabla \cdot \vec{V} = 0$$

Momentum Equations

$$\rho \frac{D\vec{V}}{Dt} = -\nabla p + \rho \vec{g} + \mu \nabla^2 \vec{V}$$

Outline

Functions of multiple variables

- Partial derivatives / differentiation. Scalar / vector fields

Tools for differential calculus on scalar and vector fields

- Gradient / Divergence / Laplacian / Curl

Partial Differential Equations

- Categories of PDE / Order, linearity, homogeneity / Auxiliary data / ‘well-posedness’

Finite difference methods for solving a PDE

- Numerical estimates of derivatives / Forward Marching / Jacobi Iteration

Estimate partial derivatives numerically

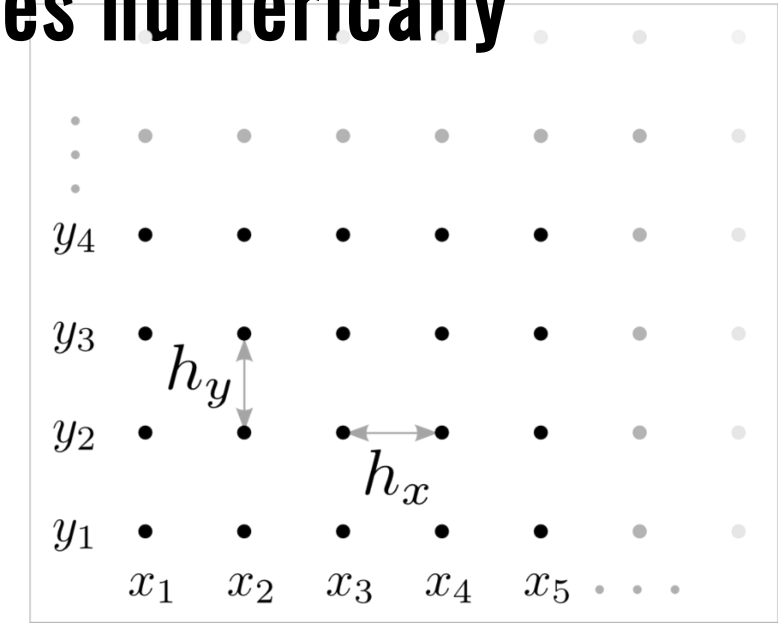
- Samples measured on a lattice

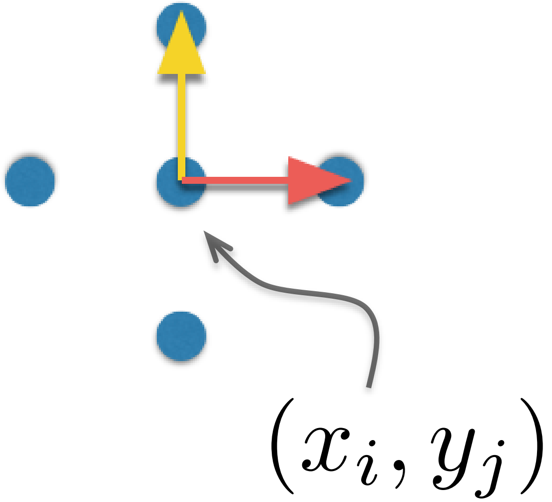
$$f(x_i, y_j) = z_{i,j}$$

- 1st order partial derivative estimates

$$\left.\frac{\partial f}{\partial x}\right|_{(x_i, y_j)} \approx \frac{z_{i+1,\,j} - z_{i,\,j}}{h_x}$$

$$\left.\frac{\partial f}{\partial y}\right|_{(x_i, y_j)} \approx \frac{z_{i,\,j+1} - z_{i,\,j}}{h_y}$$





- These are estimated using **forward differences**

Estimate partial derivatives numerically

- Samples measured on a lattice

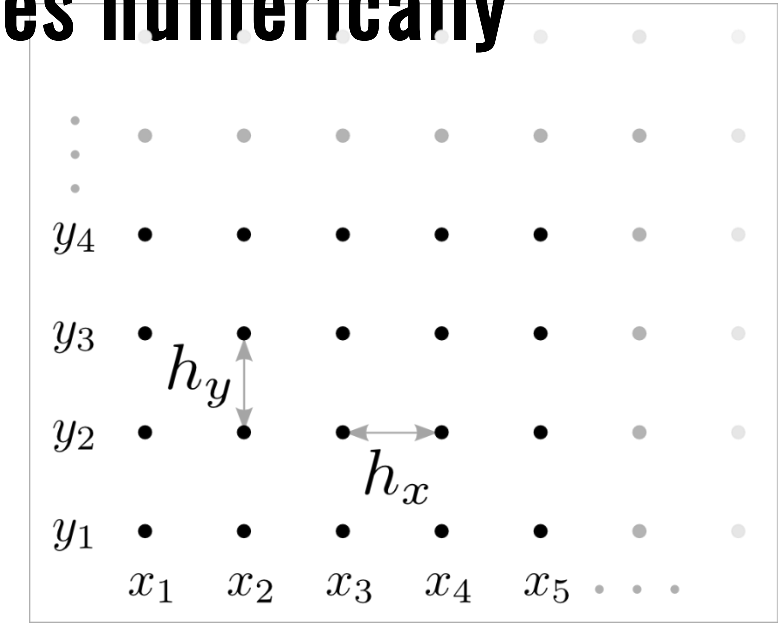
$$f(x_i, y_j) = z_{i,j}$$

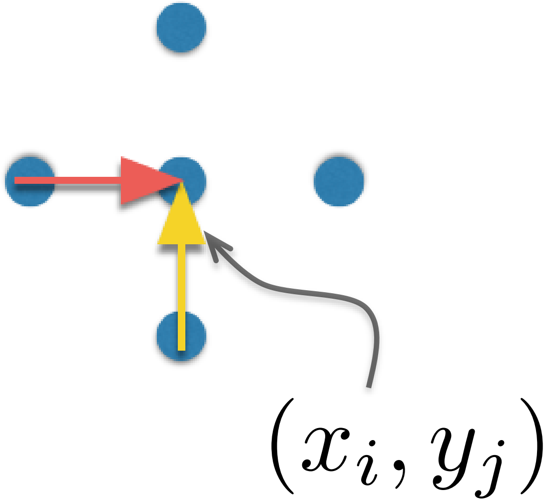
- 1st order partial derivative estimates

$$\left.\frac{\partial f}{\partial x}\right|_{(x_i, y_j)} \approx \frac{z_{i,j} - z_{i-1,j}}{h_x}$$

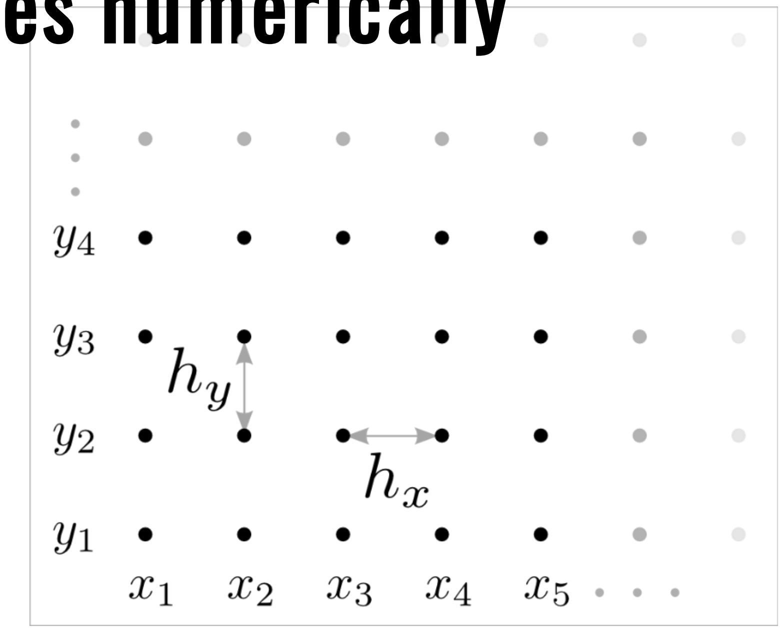
$$\left.\frac{\partial f}{\partial y}\right|_{(x_i, y_j)} \approx \frac{z_{i,j} - z_{i,j-1}}{h_y}$$

- These are estimated using **backward differences**





Estimate partial derivatives numerically



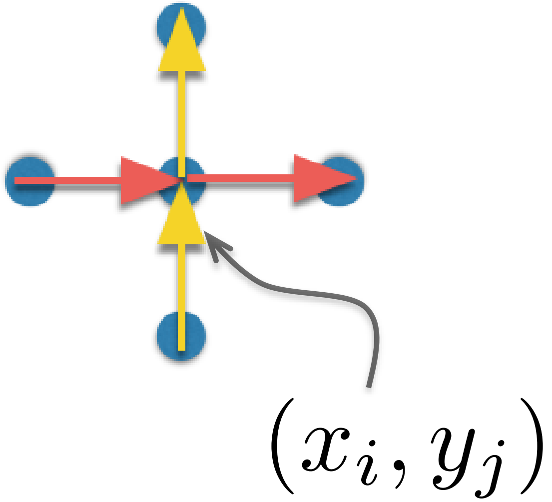
- Samples measured on a lattice

$$f(x_i, y_j) = z_{i,j}$$

- 1st order partial derivative estimates

$$\left.\frac{\partial f}{\partial x}\right|_{(x_i, y_j)} \approx \frac{z_{i+1,\,j} - z_{i-1,\,j}}{2h_x}$$

$$\left.\frac{\partial f}{\partial y}\right|_{(x_i, y_j)} \approx \frac{z_{i,\,j+1} - z_{i,\,j-1}}{2h_y}$$

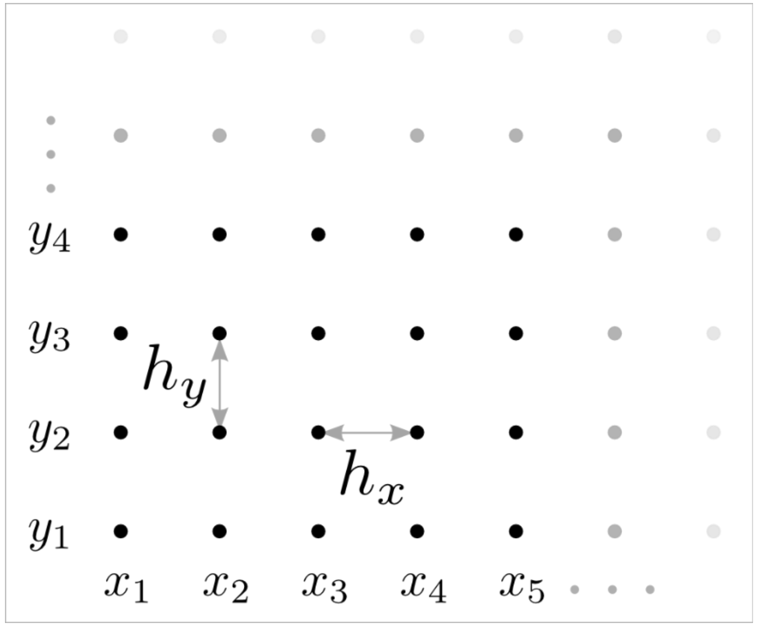


- These are estimated using **central differences**

Exercise: Estimate f_{xx}

What about second order partials, e.g. f_{xx}?

$$f(x_i, y_j) = z_{i,j}$$



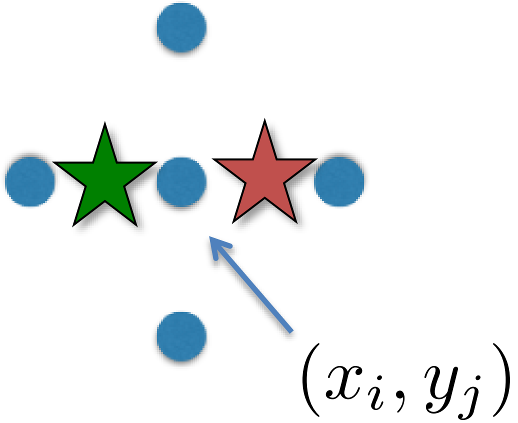
Exercise:

- Estimate f_x at horizontal mid-way point*:

  $(x_i + h_x/2\ , y_j)$

- Repeat to find an estimate for f_x at the mid-point:

  $(x_i - h_x/2\ , y_j)$



- Use both estimates of f_x to get an estimate for f_{xx} at $(x_i\ , y_j)$

* *Hint*: You can use central difference with a step size of $h_x/2$.

See also, notes section 1.2.4 for a method that uses Taylor expansion.

Example in MatLab

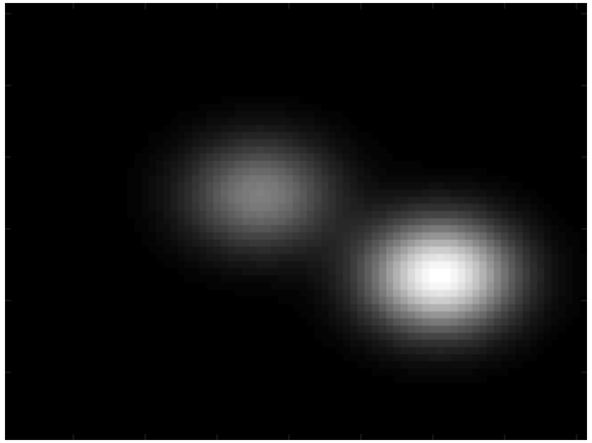
- First generate some data

```
% Define the grid
x = linspace(-4, 4, 51);
y = linspace(-4, 4, 51);

[x,y] = meshgrid(x, y);

% Simulate the data measurements.
z = exp( - (x+0.5).^2 - (y-0.5).^2 ) + 2 * exp( - (x-2).^2 - (y+1).^2 );

% Visualise data as an image.
imagesc(z), axis xy, colormap gray
```



See Computer Programming Manual Chapter 8 for meshgrid example

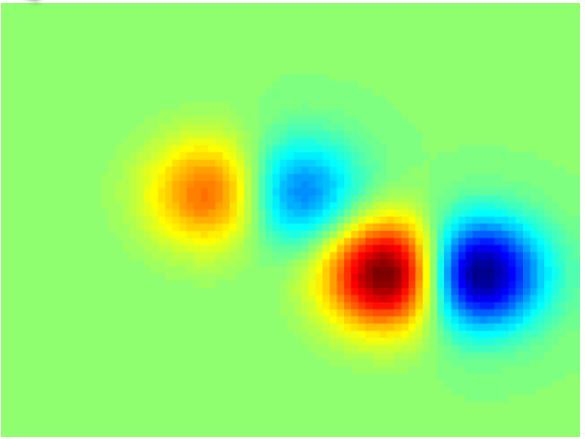
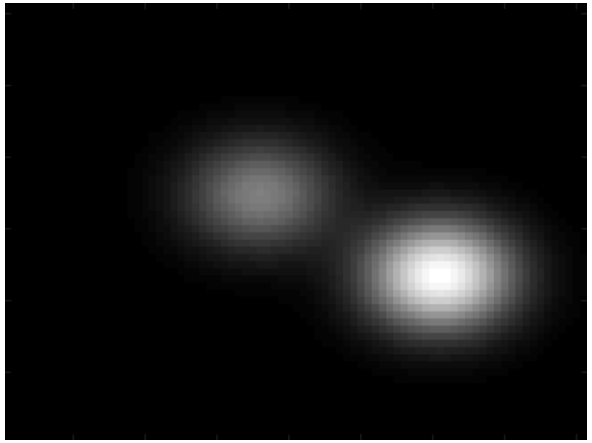
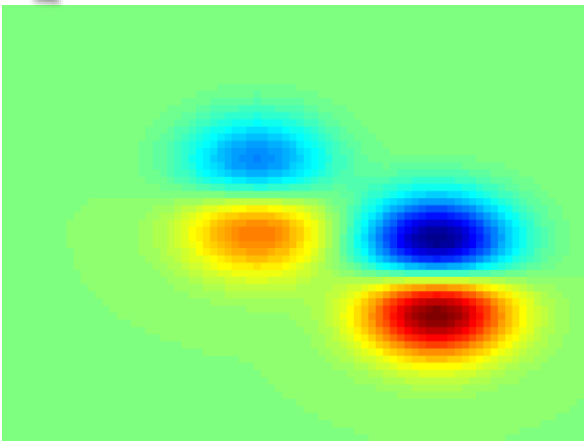
Example in MatLab

- Then estimate the partial derivatives using built in function

```
% Spacing of data in x and y directions.
dx = x(1,2) - x(1,1);
dy = y(2,1) - y(1,1);

% Estimate partial derivatives in x and y directions.
[dzdx, dzdy] = gradient(z, dx, dy);

% Plot partial derivative w.r.t. x and y
figure, imagesc(dzdx), axis xy
figure, imagesc(dzdy), axis xy
```

Finite differences for solving a PDE

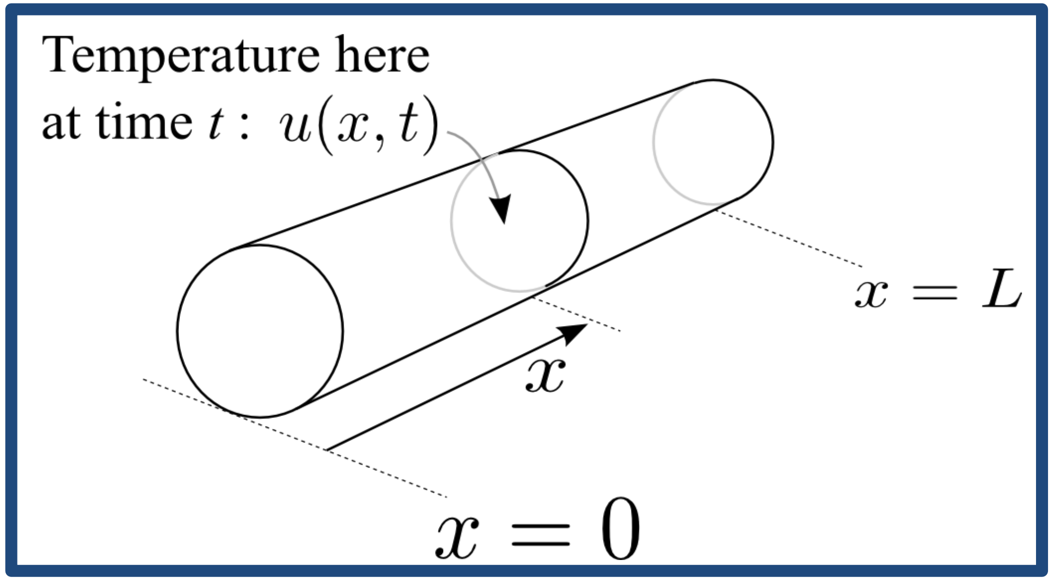
- Example: 1-D **Diffusion** equation: $u(x,t)$ is temperature along bar, D is a diffusion coefficient

$$\begin{array}{rlll} \text{solve} & \boxed{u_t = D u_{xx}} & 0 \le x \le 1, t > 0 & \\ \text{subject to} & u(0,t) = u(1,t) = 0 & t > 0 & \text{(Boundary condition)} \\ \text{and} & u(x,0) = f(x) & 0 \le x \le 1 & \text{(Initial condition)} \end{array}$$

- On a regular lattice
- Use a **discrete** x-t domain

$$x_j = j\delta_x, \quad j = 1, \ldots, N_x$$

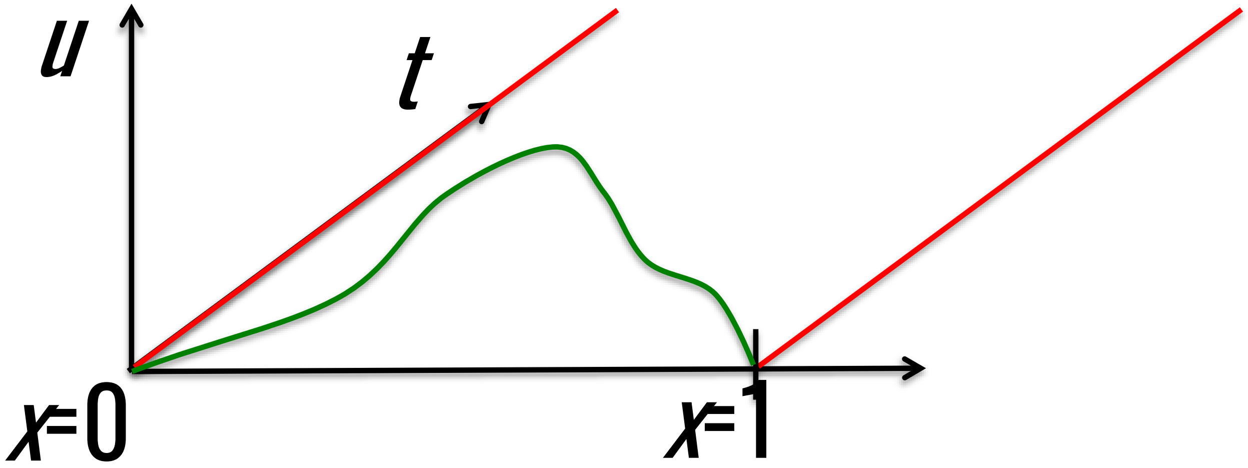
$$t_n = n\delta_t, \quad n = 1, \ldots, N_t$$



Finite differences for solving a PDE

- Think of solution u as defined 'over' the x-t domain.

$$\begin{aligned} \text{solve} \quad & u_t = D u_{xx} \qquad 0 \le x \le 1, t > 0 \\ \text{subject to} \quad & \boxed{u(0,t) = u(1,t) = 0 \qquad t > 0 \qquad \text{(Boundary condition)}} \\ \text{and} \quad & \boxed{u(x,0) = f(x) \qquad 0 \le x \le 1 \qquad \text{(Initial condition)}} \end{aligned}$$



FD Method for diffusion equation

- Discretization: **first** partial derivative w.r.t. time (*t*) and **second** partial derivative with respect to space (*x*)

$$u_t(x_j, t_n) \approx \frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\delta_t}$$

$$u_{xx}(x_j, t_n) \approx \frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{\delta_x^2}$$

- We can put these into the equation:

$$u_t = Du_{xx} \qquad 0 \le x \le 1, t > 0$$

Diffusion equation: Forward marching

- After substituting in estimates for *ut* and *uxx*

$$\frac{u(x_j, t_{n+1}) - u(x_j, t_n)}{\delta_t} = D\frac{u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)}{\delta_x^2}$$

- rearrange:

$$u(x_j, t_{n+1}) = u(x_j, t_n) + \frac{D\delta_t}{\delta_x^2}\left[u(x_{j+1}, t_n) - 2u(x_j, t_n) + u(x_{j-1}, t_n)\right]$$

Gives:

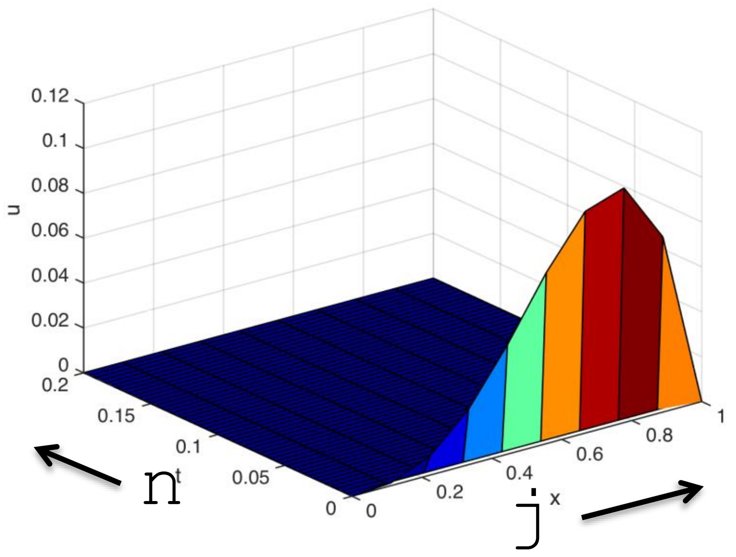
- Difference formula
- Update Equation

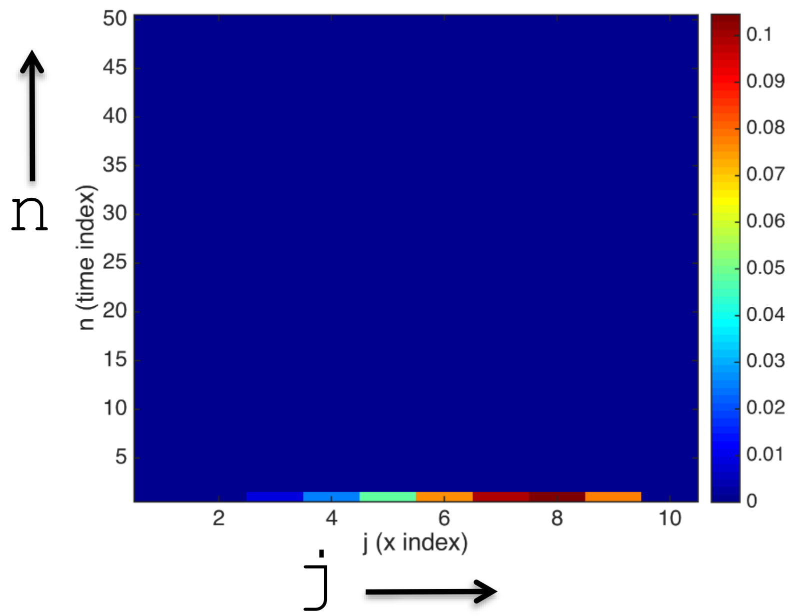
Solve the PDE using a 'forward marching' method

Can use value of u at a time-point to estimate the value at the next!

Code: Diffusion forward marching

- Using the MatLab code in the notes. Solution will be `U(j, n)`





- Initial condition

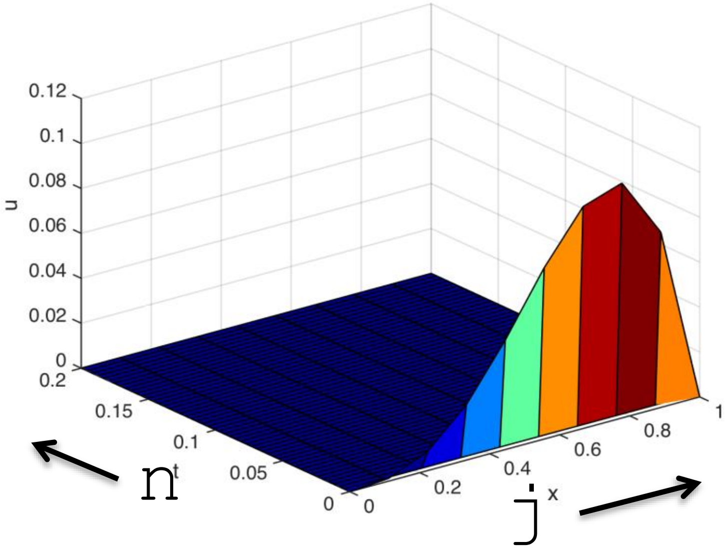
```
U(:,1) = initialConditionFunc(xVals);
```

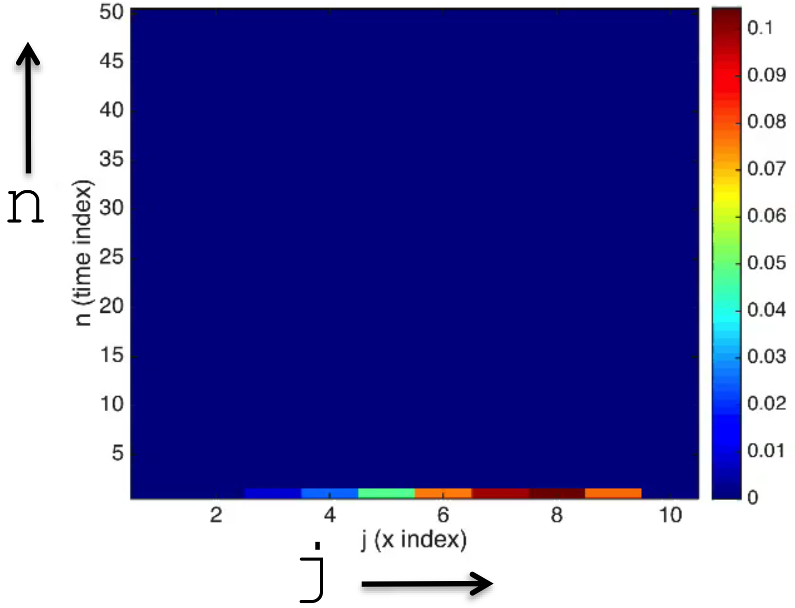
- Boundary condition

```
U(1,   :) = zeros(noOfTimePoints, 1);
U(end, :) = zeros(noOfTimePoints, 1);
```

Code: Diffusion forward marching

- Using the MatLab code in the notes. Solution will be `U(j, n)`

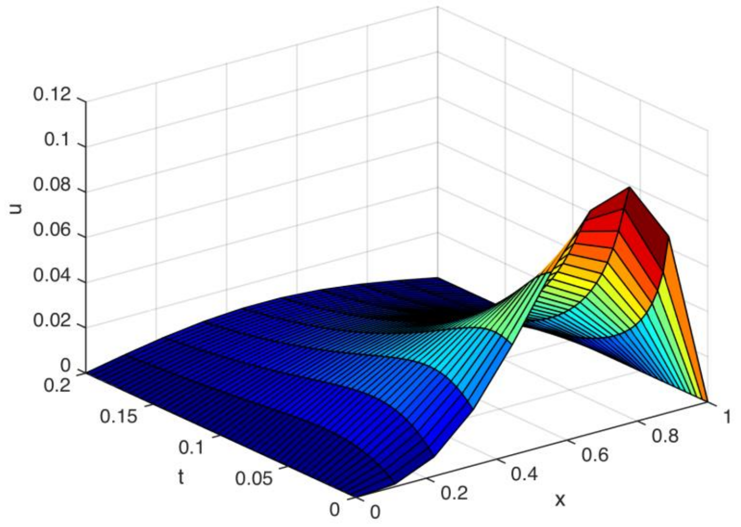


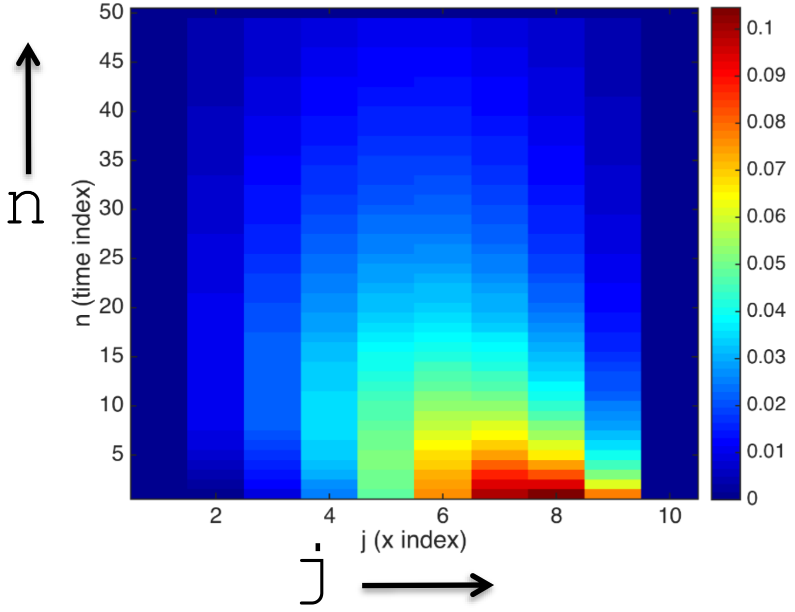



```
% Forward march to fill the array
for n = 1:noOfTimePoints-1
  for j = 2:noOfXPoints-1
    U(j, n+1) = U(j,n) + r * ( U(j-1,n) - 2*U(j,n) + U(j+1,n) );
  end
end
```

Code: Diffusion forward marching

- Using the MatLab code in the notes. Solution will be `U(j, n)`

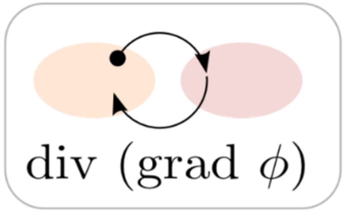





```
% Forward march to fill the array
for n = 1:noOfTimePoints-1
  for j = 2:noOfXPoints-1
    U(j, n+1) = U(j,n) + r * ( U(j-1,n) - 2*U(j,n) + U(j+1,n) );
  end
end
```

FD method: Laplace Equation

$$\nabla \cdot (\nabla \phi)$$



- Focus on 2-D version of this equation:
  - No time dependency
  - Has a boundary condition

$$\text{solve } \boxed{u_{xx} + u_{yy} = 0,} \text{ for } (x, y) \in \Omega$$
$$\text{where } u(x, y) = f(x, y) \text{ for } (x, y) \in \partial\Omega$$

- Use central difference estimates for 2nd order partial derivatives

$$u_{xx} \approx \frac{u_{i+1,\,j} - 2u_{i,\,j} + u_{i-1,\,j}}{h^2}$$
$$u_{yy} \approx \frac{u_{i,\,j+1} - 2u_{i,\,j} + u_{i,\,j-1}}{h^2}$$

FD method: Laplace Equation

- This gives the approximation

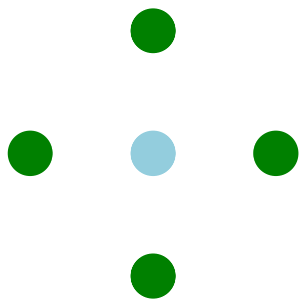
$$\frac{u_{i+1,\,j} - 2u_{i,\,j} + u_{i-1,\,j}}{h^2} + \frac{u_{i,\,j+1} - 2u_{i,\,j} + u_{i,\,j-1}}{h^2} \approx 0$$

- Which can be simplified:

$$u_{i+1,\,j} + u_{i-1,\,j} + u_{i,\,j+1} + u_{i,\,j-1} - 4u_{i,\,j} \approx 0$$

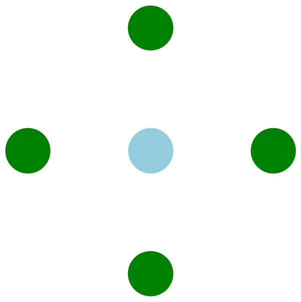
- To give finally:

$$u_{i,\,j} \approx \frac{1}{4}\left[u_{i+1,\,j} + u_{i-1,\,j} + u_{i,\,j+1} + u_{i,\,j-1}\right]$$



Laplace Equation: Jacobi iteration

- No time variable, but can still obtain an iterative update!



$$u_{i,\,j} \approx \frac{1}{4}\left[u_{i+1,\,j} + u_{i-1,\,j} + u_{i,\,j+1} + u_{i,\,j-1}\right]$$

$$u_{i,\,j}^{k+1} \approx \frac{1}{4}\left[u_{i+1,\,j}^{k} + u_{i-1,\,j}^{k} + u_{i,\,j+1}^{k} + u_{i,\,j-1}^{k}\right]$$

- Where *k* indicates the iteration steps
- Note: *every* point inside the region of interest gets updated at each iteration

Laplace Equation: Jacobi iteration

- Can implement it with the following algorithm (*pseudocode*)

```
Initialise U with BCs
k = 0
do
  U_next = U
  for each i, j
    if (i,j) inside boundary
      newVal = [U(i-1,j) + U(i,j-1) + U(i+1,j) + U(i,j+1)] / 4
      U_next(i,j) = newVal
  Udiff = |U_next - U|
  U = U_next
  k = k + 1
while k < maxIterations and Udiff > epsilon
```

Copy current estimate of solution array (includes boundary values)

Update points inside boundary in the copied array

Over-write solution estimate, increment iteration counter

Stopping condition(s)

Last remark

- Coursework on PDEs available!!!
  - To be assessed!
  - Great to learn the finite difference method (Jacobi iteration)

Biomedical Engineering 5CCYB070

COMPUTATIONAL METHODS

Lecture 7 Interpolation, approximation and extrapolation: I

01 Polynomial interpolation

02 Piecewise interpolation