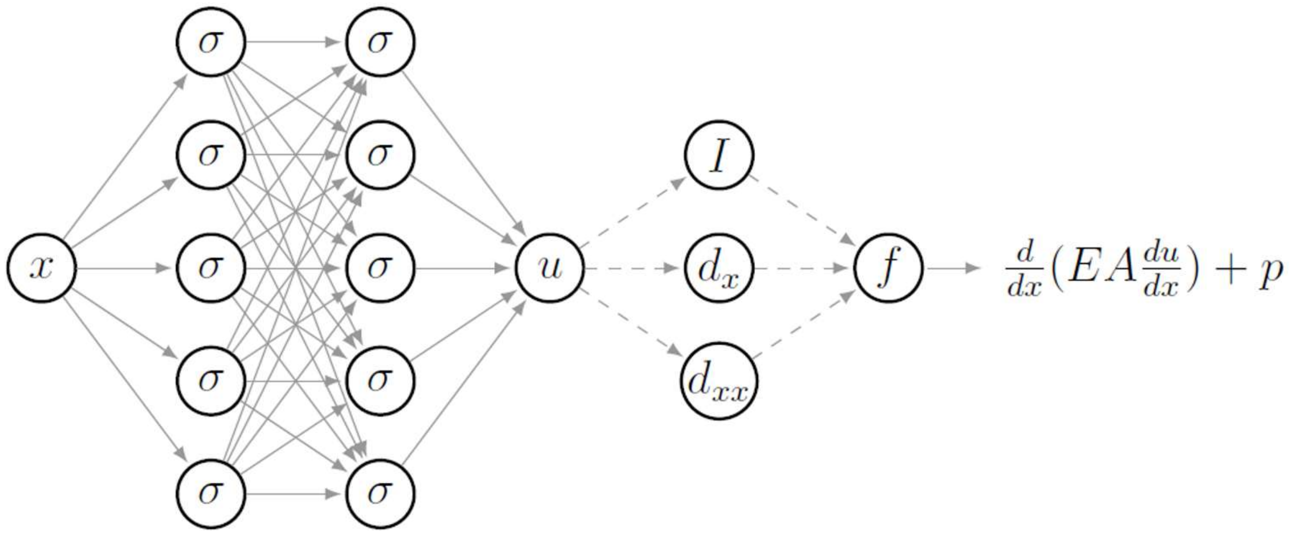
Planning of course:

- Dimensional reduction: singular value decomposition, principal component analysis, model reduction
- Regression, optimization, model selection
- Neural networks
- Physics informed neural networks, model discovery
- Projects


x
σ
u
I
d_x
d_{xx}
f
$\frac{d}{dx}(EA\frac{du}{dx}) + p$

Algorithm 4 Training a physics-informed neural network for the static solution of the problem described in Eq. (5.4).

```
Require: training data for boundary condition $\{x_b^i, u_b^i\}_{i=1}^{N_b}$
  generate $N_f$ collocation points with a uniform distribution $\{x_f^i\}_{i=1}^{N_f}$
  define network architecture (input, output, hidden layers, hidden neurons)
  initialize network parameters $\boldsymbol{\Theta}$: weights $\{\boldsymbol{W}^l\}_{l=1}^{L}$ and biases $\{\boldsymbol{b}^l\}_{l=1}^{L}$ for all layers $L$
  set hyperparameters for L-BFGS optimizer ($epochs$, learning rate $\alpha$, ...)

  for all $epochs$ do
      $\hat{\boldsymbol{u}}_b \leftarrow \boldsymbol{u}_{NN}(\boldsymbol{x}_b; \boldsymbol{\Theta})$
      $\boldsymbol{f} \leftarrow \boldsymbol{f}_{NN}(\boldsymbol{x}_f; \boldsymbol{\Theta})$
      compute $MSE_b$, $MSE_f$                                  ▷ cf. Eqs. (5.12) and (5.13)
      compute cost function: $C \leftarrow MSE_b + MSE_f$
      update parameters: $\boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta} - \alpha \frac{\partial C}{\partial \boldsymbol{\Theta}}$       ▷ L-BFGS
  end for
```

Continuous time problems, 1D transient heat

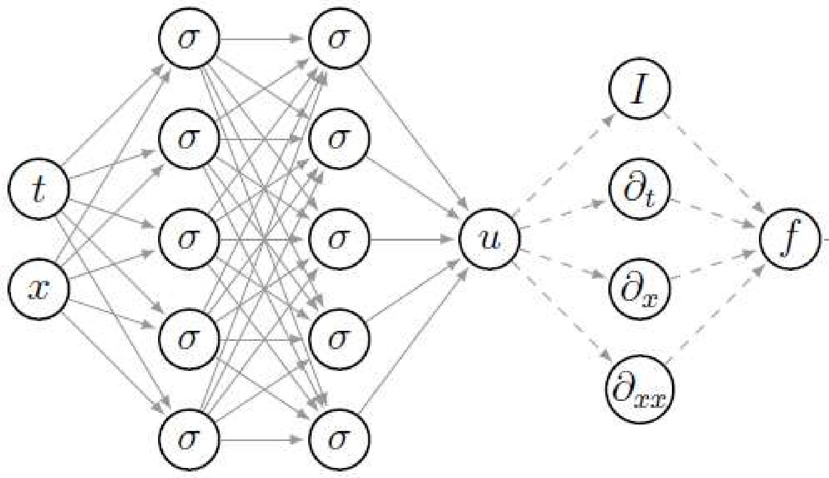
$$c\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(\kappa \frac{\partial u}{\partial x}\right) - s = 0 \qquad \text{on } \mathcal{T} \times \Omega, \qquad (5.14)$$

$$\kappa \frac{\partial u}{\partial x} = h \qquad \text{on } \mathcal{T} \times \Gamma_N, \qquad (5.15)$$

$$u = g \qquad \text{on } \mathcal{T} \times \Gamma_D, \qquad (5.16)$$

$$u(x,0) = u_0 \qquad \text{on } \Omega. \qquad (5.17)$$

$$c(u) = 1/2000\, u^2 + 500,$$
$$\kappa(u) = 1/100\, u + 7.$$



$$c\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}(\kappa\frac{\partial u}{\partial x}) - s$$

d.c

Fulfilled in points of x,t-space

$$s = \frac{\kappa u}{\sigma^2} + u\frac{x-p}{\sigma^2}\left[c\frac{\partial p}{\partial t} - \frac{x-p}{\sigma^2}\left(\kappa + u\frac{\partial \kappa}{\partial u}\right)\right]$$

Algorithm 5 Training a physics-informed neural network for the continuous solution of the problem described in Eq. (5.14).

```
Require: training data for initial condition $\{0, x_0^i, u_0^i\}_{i=1}^{N_0}$
Require: training data for boundary condition $\{t_b, x_b^i, u_b^i\}_{i=1}^{N_b}$
  generate $N_f$ collocation points with Latin-hypercube sampling $\{t_f, x_f^i\}_{i=1}^{N_f}$
  define network architecture (input, output, hidden layers, hidden neurons)
  initialize network parameters $\Theta$: weights $\{W^l\}_{l=1}^L$ and biases $\{b^l\}_{l=1}^L$ for all layers $L$
  set hyperparameters for Adam optimizer (Adam-$epochs$, learning rate $\alpha, \dots$)
  set hyperparameters for L-BFGS optimizer (L-BFGS-$epochs$, convergence criterion, ...

  procedure TRAIN
      compute $\{u_{NN}(0, x_0^i, \Theta)\}_{i=1}^{N_0}$
      compute $\{\frac{\partial}{\partial x} u_{NN}(t_b^i, x_b^i; \Theta)\}_{i=1}^{N_b}$
      compute $\{f_{NN}(t_f^i, x_f^i; \Theta)\}_{i=1}^{N_f}$
      compute $MSE_0$, $MSE_b$, $MSE_f$                          ▷ cf. Eqs. (5.27) to (5.29)
      evaluate cost function: $C \leftarrow MSE_0 + MSE_b + MSE_f$
      update parameters: $\Theta \leftarrow \Theta - \alpha \frac{\partial C}{\partial \Theta}$        ▷ Adam or L-BFGS
  end procedure

  for all Adam-$epochs$ do
      run TRAIN with Adam optimizer
  end for
  for all L-BFGS-$epochs$ do
      run TRAIN with L-BFGS optimizer
  end for
```

$$MSE_0 = \frac{1}{N_0} \sum_{i=1}^{N_0} \left(u_{NN}(0, x_0^i; \Theta) - u_0^i \right)^2$$

$$MSE_b = \frac{1}{N_b} \sum_{i=1}^{N_b} \left(\frac{\partial}{\partial x} u_{NN}\left(t_b^i, 0; \Theta\right) \right)^2 + \frac{1}{N_b} \sum_{i=1}^{N_b} \left(\frac{\partial}{\partial x} u_{NN}\left(t_b^i, 1; \Theta\right) \right)^2$$

$$MSE_f = \frac{1}{N_f} \sum_{i=1}^{N_f} \left(f_{NN}\left(t_f^i, x_f^i\right) \right)^2 \qquad f := c\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(\kappa \frac{\partial u}{\partial x}\right) - s$$