Lecture October 21

Some hints (code examples will be uploaded later) concerning project 2:

- write a simple gradient descent for exercise 1 first.

- implement thereafter the stochastic gradient descent. Play around with different learning rates, epochs and batches

- Feel free to use autograd

- For exercise 2, try first Sklearns MLP regressor in order to get a feel for what you should "expect."

$$H_{OLS} = \frac{2}{n} X^T X$$

$$\eta = \text{learning} \le \frac{1}{\lambda_{max}(H)}$$

ODE = ORDINARY DIFF eq.

PDE = PARTIAL DIFF eq.

ODE

$$m \frac{d^2 x}{dt^2} + \eta \frac{dx}{dt} + x(t) = f(t)$$

$$\frac{dy}{dx} = -\gamma\, y(x)$$

initial condition

$$y(x_0) = y_0$$

$$y(x) = y_0 e^{-\gamma x}$$

$$\Rightarrow F(x, u(x), Du, D^2 u) = 0$$

$F(x, y(x), \partial_x y, \partial_x y) = 0$

$$F = \frac{dy}{dx} + \gamma y(x)$$

Discretize into a Domain D for x and y

$$x \to x_i = x_0 + i\Delta x$$

$$i = 0, 1, 2, \ldots n$$

$$\Delta x = \frac{x_n - x_0}{n}$$

$$y(x) \to y(x_i) = y_i$$

Taylor-expand around

$$x_i \pm \Delta x$$

$$y(x_i \pm \Delta x) = y_{i \pm 1} = y_i \pm \Delta x \left.\frac{dy}{dx}\right|_{x=x_i} + \frac{(\Delta x)^2}{2!} \left.\frac{d^2y}{dx^2}\right|_{x=x_i} + O(\Delta x^3)$$

Famous Euler methods

$$y_{i+1} \simeq y_i + \Delta x \underbrace{\left.\frac{dy}{dx}\right|_{x=x_i}}_{y'}$$

$$\frac{dy}{dx} = -\gamma y(x)$$

$$y_{i+1} = y_i + \Delta x \left(-\gamma y(x)\right)$$

$$= y_i - \gamma \Delta x \, y_i$$

<u>in NNs</u>

Trial function

$$y_t(x, P)$$

$\uparrow$ set of adjustable parameters

$$\underset{P \in \mathbb{R}^m}{\arg\min} \sum_{i \in D} \left[F\left(x_i, y_t(x_i, P), \nabla y_t, \nabla^2 y_t\right)\right]^2$$

$$y_t(x, P) = A(x) + f(x, N(x, P))$$

satisfies the initial and/or boundary conditions.

our neural network

CNNs

second sample:

x has dimension $d = 10^6$

simple model with one hidden layer and $n = 10^4$ neurons.

output layer: one output node (True/False)

parameters:

input ⇒ hidden

weights $= 10^4 \cdot 10^6$

biases $= 10^4$

hidden to output

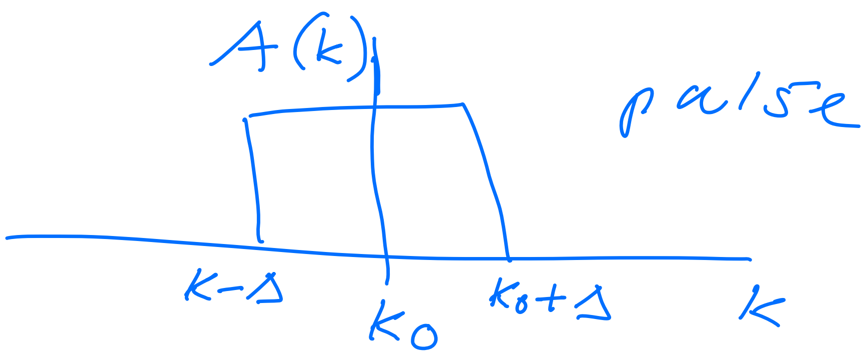
\# weights $= 10^4 \times 1 = 10^4$

\# biases $= 1$

\# parameters ; $10^{10} + 10^4 + 10^4 + 1$

$\simeq 10^{10}$

—— math of convolution =

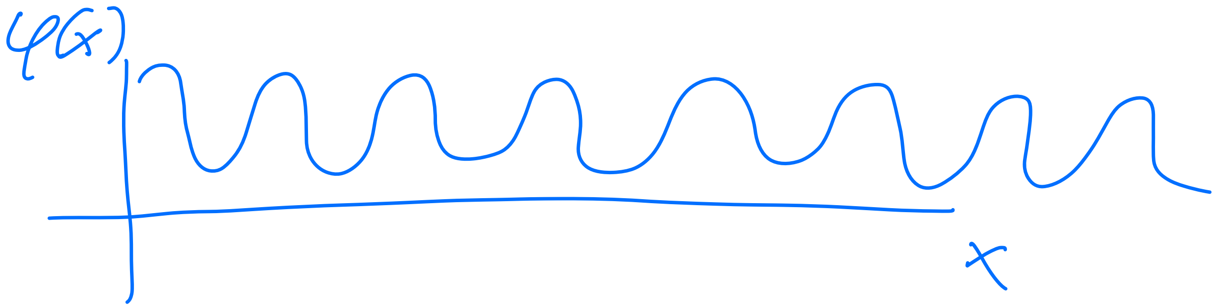


$$\varphi(x) = \int_{-\infty}^{\infty} \underbrace{A(k)}_{=1} \cos(2\pi k x)\, dk$$

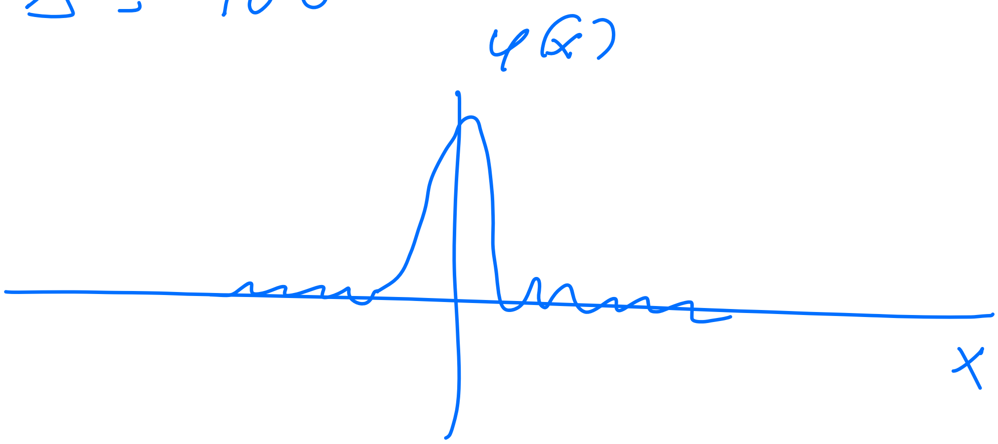
$$= \int_{k_0-\Delta}^{k_0+\Delta} \cos(2\pi k x)\, dk$$

$$= 2\Delta \cos(2\pi k_0 x) \frac{\sin(2\pi\Delta x)}{2\pi\Delta x}$$

$\Delta = 10^{-3}$



$\Delta = 100$



Convolution ;

$$s(t) = \int \underline{x(a)}\, w(t-a)\, da$$

weight function

— Polynomial multiplication

$$p(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2$$

$$s(t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3$$

$$z(t) = p(t)\, s(t) =$$

$$\delta_0 + \delta_1 t + \delta_2 t^2 + \delta_3 t^3$$

$$+ \delta_4 t^4 + \delta_5 t^5$$

$$\delta_0 = \alpha_0 \beta_0$$

$$\delta_1 = \alpha_1 \beta_0 + \alpha_0 \beta_1$$

$$\delta_2 = \alpha_0 \beta_2 + \alpha_1 \beta_1 + \alpha_2 \beta_0$$

$$\delta_3 = \alpha_1 \beta_2 + \alpha_2 \beta_1 + \alpha_0 \beta_3$$

$$\delta_4 = \alpha_2 \beta_2 + \alpha_1 \beta_3$$

$$\delta_5 = \alpha_2 \beta_3$$

$\alpha_i = 0$ except $i = \{0, 1, 2\}$

$\beta_i = 0$ —— i —— $\{0, 1, 2, 3\}$

∞

$$S_t = \sum_{i=-\infty} \alpha_i \beta_{t-i}$$

$$= (\alpha * \beta)_t$$

$$S_t = \sum_{\substack{i,j \\ (i+j=t)}} \alpha_i \beta_j$$

$$S = \begin{bmatrix} \alpha_0 & 0 & 0 & 0 \\ \alpha_1 & \alpha_0 & 0 & 0 \\ \alpha_2 & \alpha_1 & \alpha_0 & 0 \\ 0 & \alpha_2 & \alpha_1 & \alpha_0 \\ 0 & 0 & \alpha_2 & \alpha_1 \\ 0 & 0 & 0 & \alpha_2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}$$

$$= T(\alpha) \cdot \beta$$

$$= \tilde{T}(\beta) \cdot \alpha \qquad \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix}$$

$$\tilde{T}(\beta) = \begin{bmatrix} \beta_0 & 0 & 0 \\ \beta_1 & \beta_0 & 0 \\ \beta_2 & \beta_1 & \beta_0 \\ \beta_3 & \beta_1 & \beta_2 \\ 0 & \beta_3 & \beta_2 \end{bmatrix}$$