Poster Session - 207

ATHIC2023

Computational Deep Neural Network for Solving Differential Equation

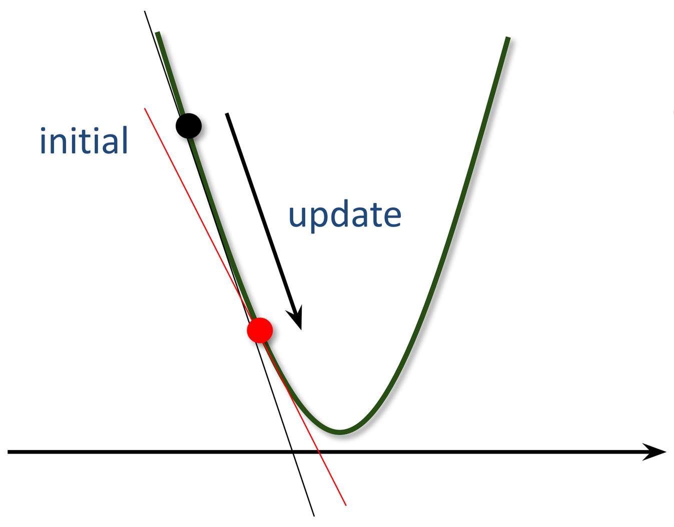
Hyunwoo Kim[1], Do Wan Kim[2], and Jin-Hee Yoon[1]
[1]Department of Physics, Inha University, Incheon 22212, Korea
[2]Department of Mathematics, Inha University, Incheon 22212, Korea

Introduction

- Deep neural network(DNN) is an algebraic structure to solve optimization problems.
- Generally, DNNs have used for classification and curve fit, using their non-linearity.
- Phenomena in nature are changing in time, so they are represented as differential equation.
- Existing models of DNNs have adopted sigmoid and ReLU as activation functions, but their properties are not appropriate to solve differential equations due to their compactness or limitation in differentiation.
- So we will introduce DNN model with modified activation function with no such limitation to solve the physics problem.

Mathematical Model

Gradient Descent



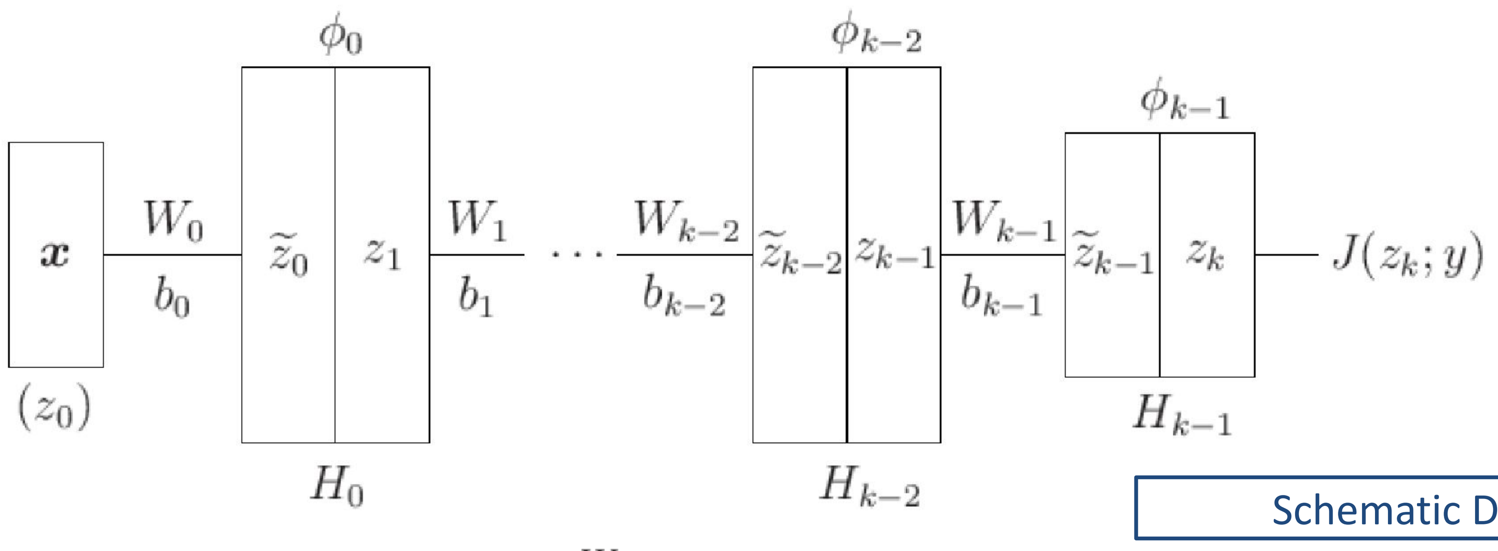
Define Loss Function

$$J(W) \equiv W^T\left(\sum_{i=1}^{N}(y_i-\hat{y}_i)\mathbf{x}_i\right)=\sum_{i=1}^{N}(y_i-\hat{y}_i)W^T\mathbf{x}_i \le 0$$

- W is a weight parameter
- Calculate the gradient of loss and find W that makes loss to have a minimum value

$$W^{\text{new}} = W + \eta\frac{\partial J}{\partial W}$$: η is a learning rate

Backpropagation



$$\mathbf{x} \xrightarrow[\mathbf{b}_0]{W_0} H_0^{\sigma_0}(\tilde{\mathbf{z}}_0,\mathbf{z}_1) \xrightarrow[\mathbf{b}_1]{W_1} \cdots \xrightarrow[\mathbf{b}_{K-1}]{W_{K-1}} H_{K-1}^{\sigma_{K-1}}(\tilde{\mathbf{z}}_{K-1},\mathbf{z}_K)$$

Schematic Diagram

- The flow diagram from input to output
- ϕ is an activation function.
- W is a weight parameter.
- b is a bias parameter.
- Train both of parameters.

Feed-Forward

$$\tilde{z}_i = W_i z_i + b_i \ (i = 0,1,2,\ldots,k-1)$$

$$z_i = \phi_{i-1}(\tilde{z}_{i-1}) \ (i = 0,1,2,\ldots,k-1,k)$$

Backpropagation

$$\frac{\partial J}{\partial W_{k-1}} = \frac{\partial J}{\partial z_k}\frac{\partial z_k}{\partial \tilde{z}_{k-1}}\frac{\partial \tilde{z}_{k-1}}{\partial W_{k-1}} = (z_k - y)[D\phi_{k-1}(z_k)]\tilde{z}_{k-1}$$

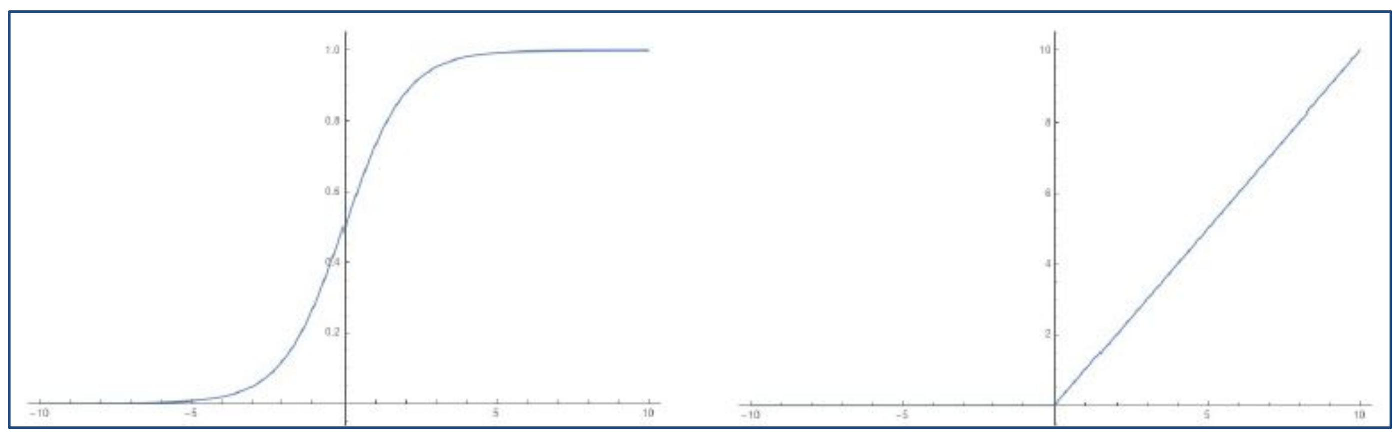
$$\frac{\partial J}{\partial W_0} = \frac{\partial J}{\partial W_{k-1}}\frac{\partial W_{k-1}}{\partial W_{k-2}}\cdots\frac{\partial W_1}{\partial W_0}$$

$$\frac{\partial J}{\partial b_0} = \frac{\partial J}{\partial b_{k-1}}\frac{\partial b_{k-1}}{\partial b_{k-2}}\cdots\frac{\partial b_1}{\partial b_0}$$

- Return to the opposite direction to update the parameters
- They can be represented as vector calculation.

$$(W_i,\mathbf{b}_i)^{\text{update}} = (W_i,\mathbf{b}_i) - \eta\frac{\partial J}{\partial(W_i,\mathbf{b}_i)}$$

Activation Function

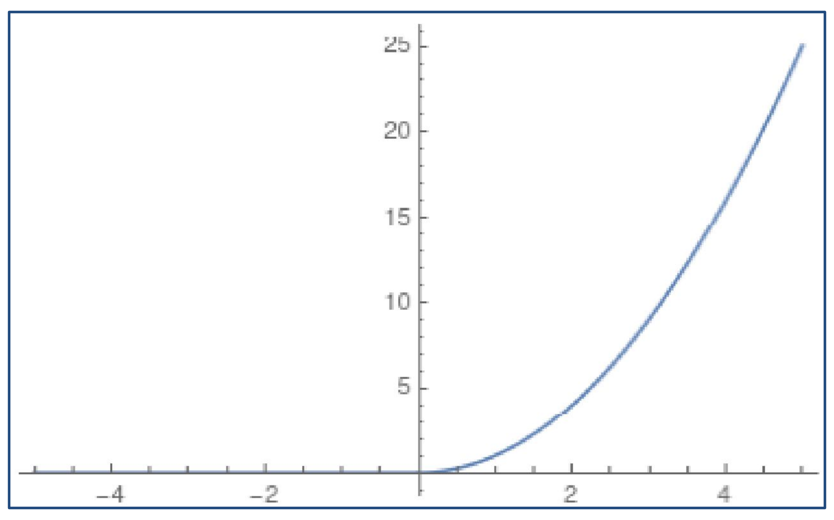


sigmoid | Rectified Linear Unit (ReLU)

- Sigmoid requires any S-shaped functions (e.g. inverse tangent), ReLU is represented by $\max(0,x)$.
- Sigmoid is a smooth function, but cannot create compact basis by linear combination.
- ReLU can create compact basis by linear combination, but the basis is not differentiable at certain points.

Mathematical model

Higher-order ReLU



$$\mathrm{ReLU}_{(1)}(x) = x^+$$

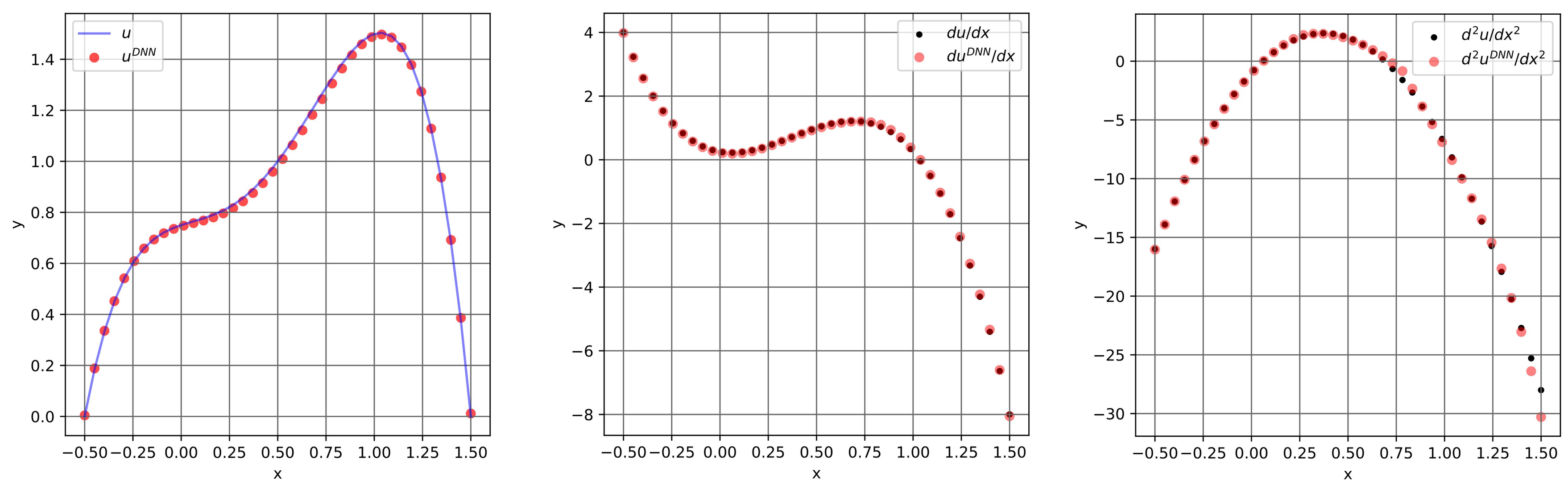
$$\mathrm{ReLU}_{(2)}(x) = \frac{1}{2!}xx^+$$

$$\mathrm{ReLU}_{(3)}(x) = \frac{1}{3!}x^2x^+$$

- Arbitrary continuous function can be approximated by combination of polynomials.
- Combination of activation functions of each hidden layer can do.

Results and Conclusion

1D elliptic equation



- original function
- first derivative
- second derivative

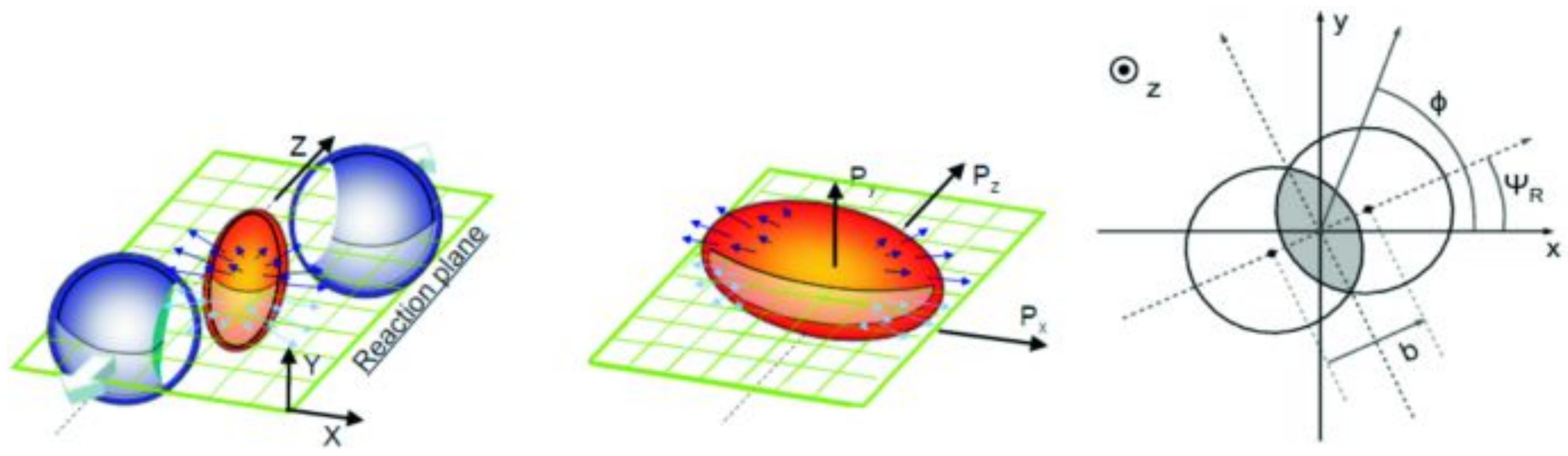
$$u_{xx} - f(x) = 0$$

$$f(x) = -(24x^2 - 18x + 1)$$

Boundary condition : $u\left(-\frac{1}{2}\right) = u\left(\frac{3}{2}\right) = 0$

- Training results after 5000 epoch, loss converges to 0.
- Use the finite difference method to calculate the derivative, and it can reduce computing time.
- Without uniform sampling, Effective interpolation is possible.

Future Works



Aggarwal, M.M. (2021). Elliptic Flow in Relativistic Heavy-Ion Collisions. In: Puri, R.K., Aichelin, J., Gautam, S., Kumar, R. (eds) Advances in Nuclear Physics. Springer Proceedings in Physics, vol 257. Springer, Singapore. https://doi.org/10.1007/978-981-15-9062-7_13

- In high-energy nuclear physics, the flow of the medium after the collision of two heavy ions is explained via hydrodynamics.
- Hydrodynamics is successful in describing final state observables.
- In hydrodynamics, the flow is represented by differential equation.
- Differential equations of hydrodynamics generally doesn't have analytic solutions.
- We can introduce DNN as a tool for solving such differential equations.

Summary

- Deep neural networks are algebraic structures that perform optimization processes.
- Linear algebra allows us to express the process of neural networks as array operations.
- Sigmoid and ReLU are generally used as an activation function of neural networks, but they are not appropriate to solve differential equation. So we proposed modified activation function.
- Using the higher-order ReLU, we can expect to solve differential equation effectively.
- Considering its advantages, DNNs can be improve their efficiency for another problems.
- Furthermore, applications in the field of high-energy nuclear physics can also be expected.