Shrinkage Initialization for Smooth Learning of Neural Networks

Miao Cheng

Email: miaocheng@acm.org
Guangxi Normal University
Guilin, China

ICBDC '24, Bangkok, Thailand

May 26, 2024

Introduction

- The success of intelligent systems has quite relied on wide applications of intelligent computing technologies.
- Neural learning approaches
  - Neural networks
  - Deep learning

Introduction

Particular Characteristics:

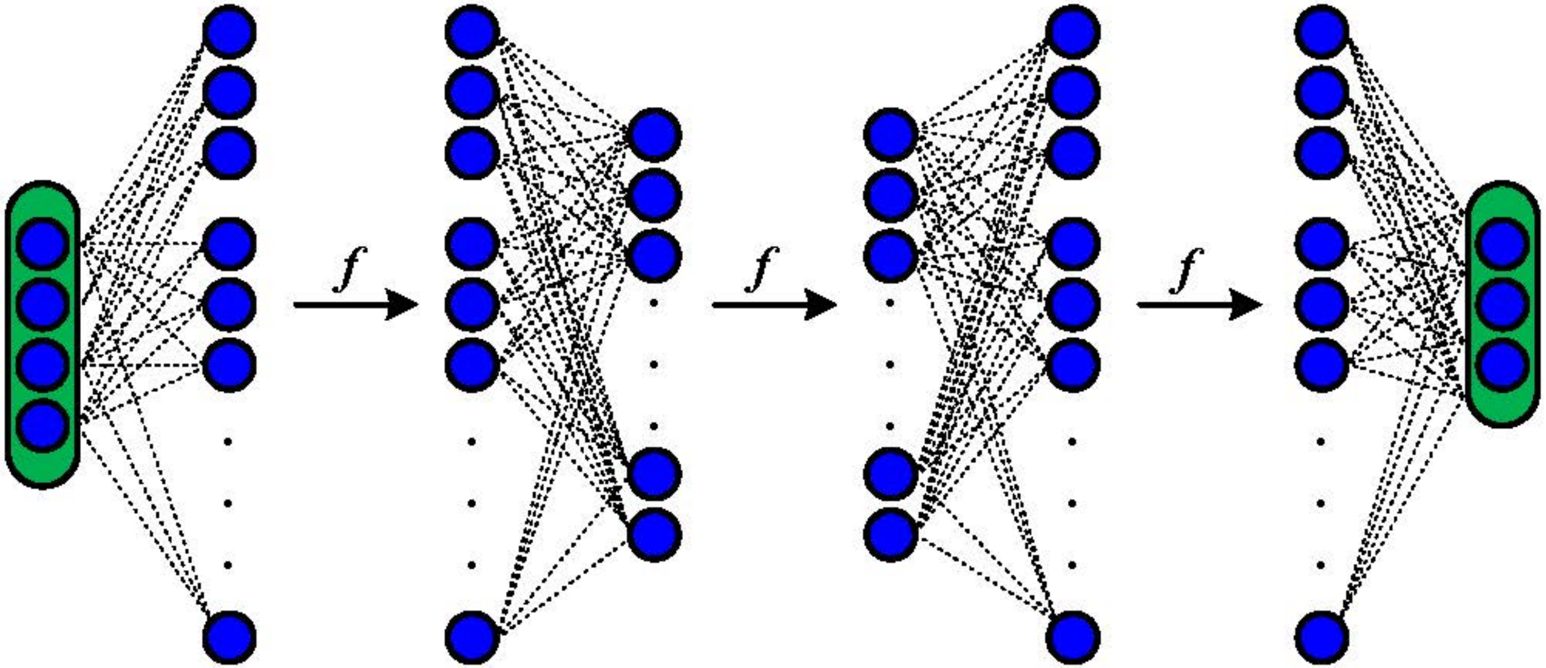
- To Learn the optimal networks for forward inference of information.
- Multi-layer structures of connected neurons with correspondence.

Introduction

- The training of networks has been difficult, due to the communicated optimization among multi-layer structures.
- Hence, backpropagation is introduced and brought to make the optimization of neural learning applicable.

Neural Learning

- The standard structure of networks



Neural Learning

Neural learning (training) can benefit from:

- The specifically choosen layers of neurons, e.g., neural transformations.

Neural Learning

Neural learning (training) can benefit from:

- The specifically choosen layers of neurons, e.g., neural transformations.
- The **activation** function of each connection for forward inference, e.g., Sigmoid, Tanh, ReLU.

Neural Learning

Neural learning (training) can benefit from:

- The specifically choosen layers of neurons, e.g., neural transformations.
- The **activation** function of each neuron for forward inference, e.g., Sigmoid, Tanh, ReLU.
- The ratio of **dropout** between the connected units.

Neural Learning

Neural learning (training) can benefit from:

- The specifically choosen layers of neurons, e.g., neural transformations.
- The **activation** function of each neuron for forward inference, e.g., Sigmoid, Tanh, ReLU.
- The ratio of **dropout** between the connected units.
- The particularly devised **initialization** of neural learning, if there exists.

Initialization

- **Batch Normalization:**
  Center and normalize each neuron output with respect to covariance.
- **Dynamic Initialization of Nonlinear Learning:**
  Initialize the transformation of networks with dynamically matching of neuron pair.
- **Layer-Sequential Unit Variance Initialization:**
  Enhanced dynamic initialization with batch normalization.

Initialization

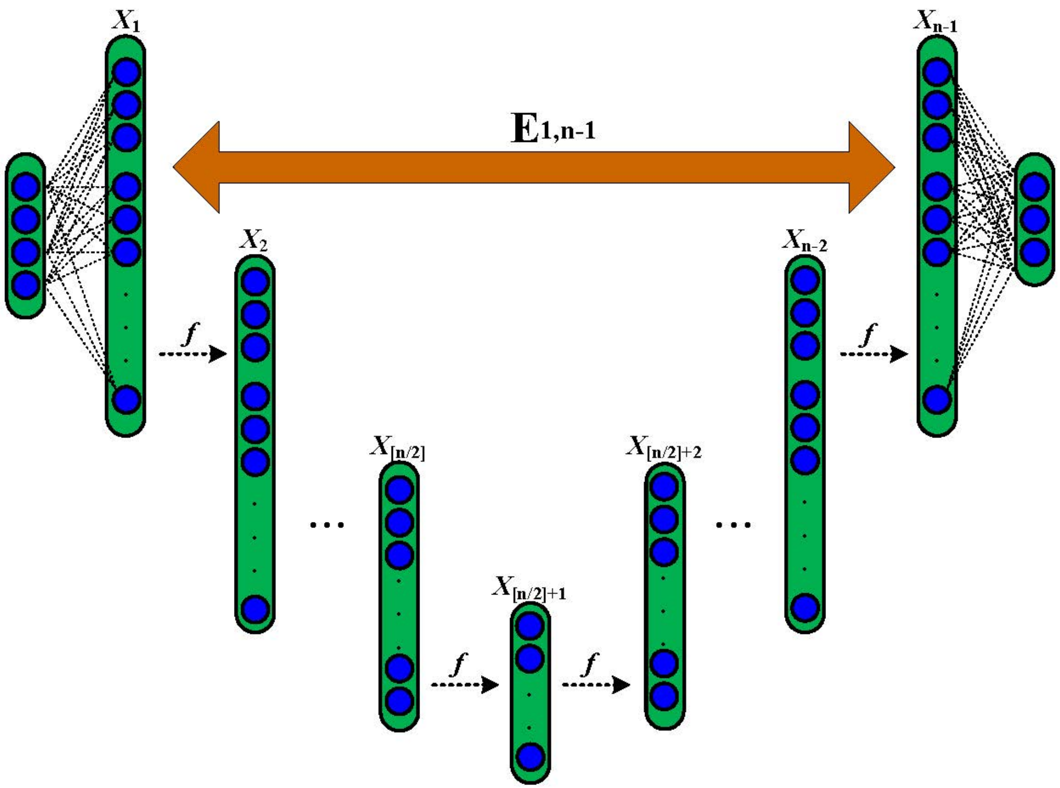
Highlights: the dynamic solutions are based on assumptions

- **Quite a few neurons:** prevented from the complicated perceptrons.
- **No activation:** The original inputs of each neuron.
- Particularly, the **smooth** inference among different layers of neurons.

Shrinkage Initialization

The transformation of neurons:

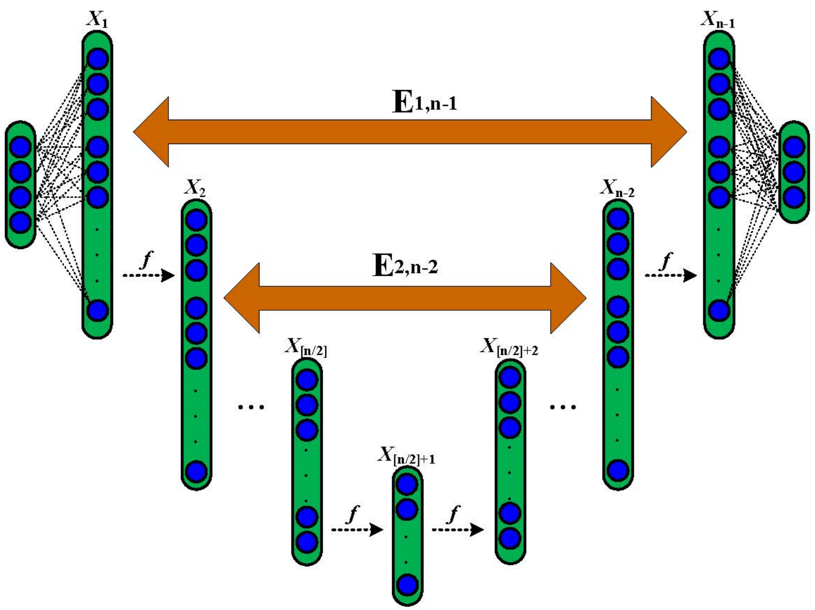
- The approximate alignment from the **source** neurons to the **target** ones.



Shrinkage Initialization

The transformation of neurons:

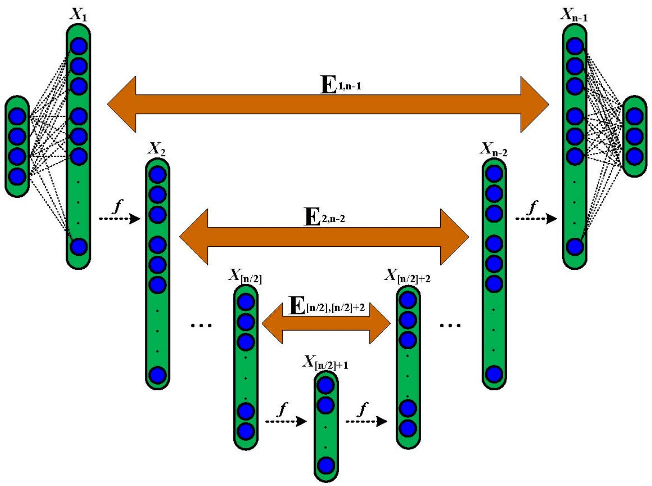
- The approximate alignment from the **source** neurons to the **target** ones.
- The **smoothness** can be approximated with *Sigmoid* for activation.



Shrinkage Initialization

The transformation of neurons:

- The approximate alignment from the **source** neurons to the **target** ones.
- The **smoothness** can be approximated with *Sigmoid* for activation.
- The original connections of neurons can be improved with **orthogonal rotations** [Saxe14][Mishkin16].



Shrinkage Initialization

- For the alignment between the i th layer and the j th layer: $X_i \rightarrow X_j$, which can be explained as
$$X_j = W_{i \rightarrow j} X_i. \tag{1}$$
- Thus, $\mathrm{E_{ij}}$ can be simply calculated as the regressive bridge, such as
$$\mathrm{E_{ij}} = X_j {X_i}^T \left(X_i {X_i}^T \right)^+ \tag{2}$$

Shrinkage Initialization

- Accordingly, the orthogonal rotations can be obtained by

$$\mathrm{E_{ij}} = U_{ij} S_{ij} V_{ij}^T \tag{3}$$

- Update connections of the adjacent neurons,

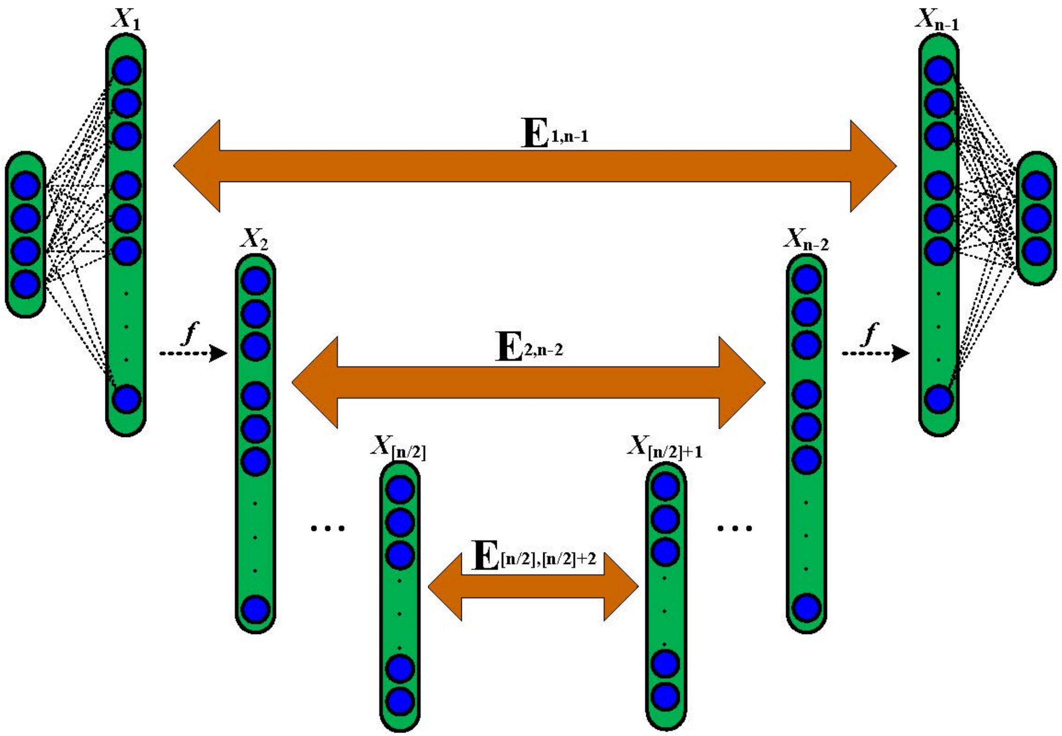
$$W_{i \to i+1} = W_{i \to i+1} V_{ij}^T \tag{4}$$

$$W_{j-1 \to j} = U_{ij} W_{j-1 \to j} \tag{5}$$

Shrinkage Initialization

- The number of neuron layers can be odd.

$$\mathrm{E_{ij}} = W_{i \to j} \tag{6}$$



Shrinkage Initialization

- The orthogonal rotations is defined as self-rotation,

$$W_{i \to j} = U_{ij} S_{ij} V_{ij}^T \tag{7}$$

which is updated as

$$W_{i \to j} = U_{ij} V_{ij}^T \tag{8}$$

Complexity Analysis

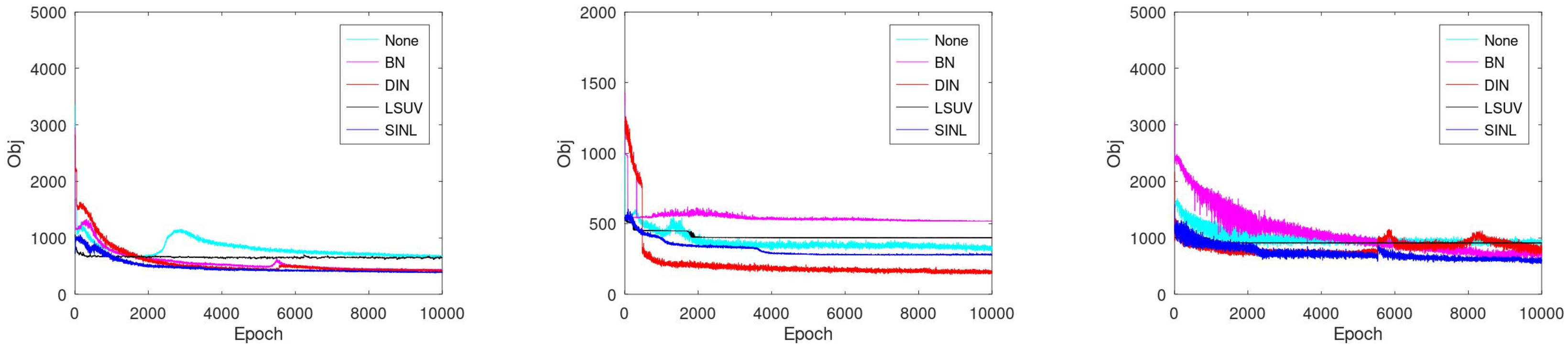
- The calculation of the alignment bridge between $X_i \in \mathbb{R}^{p \times n}$ and $X_j \in \mathbb{R}^{q \times n}$ requires $O\left(pqn + q^3\right)$.
- The decomposition of the bridge $W_{ij} \in \mathbb{R}^{q \times p}$ requires $O\left(p^2 q + pq^2\right)$.
- Note that, both $U \in \mathbb{R}^{q \times q}$ and $V \in \mathbb{R}^{p \times p}$ are **full-rank** orthogonal matrices.

Experiments

- **Data sets:**
  - **Coil 20**
  - **Monkey**
  - **Letter**
- **Methods:**
  - **Baseline(None)**
  - **BN[Ioffe15]**
  - **DIN[Saxe14]**
  - **LSUV[Mishkin16]**

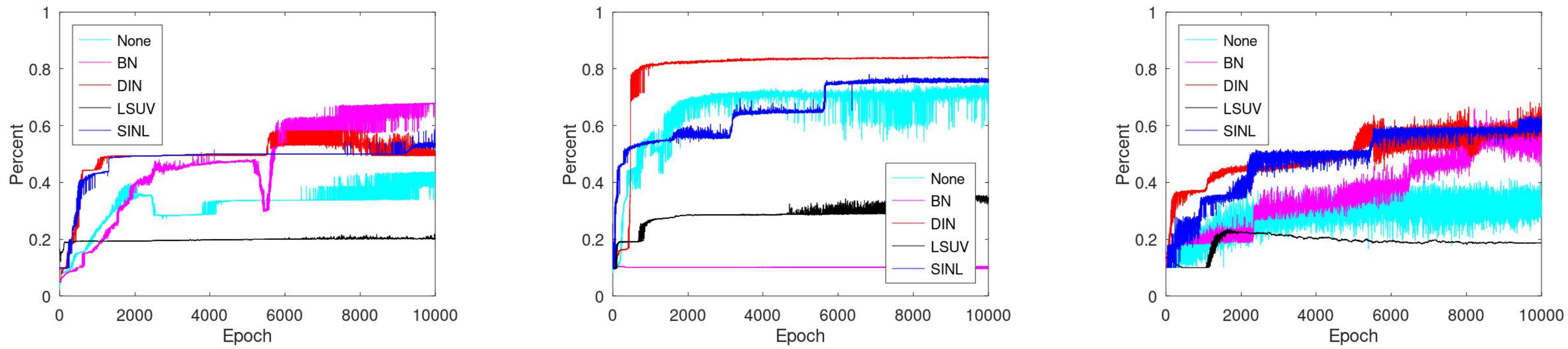
Experiments

The obtained objectives associated with the iterative epochs.



Experiments

The obtained accuracy associated with the iterative epochs.



Conclusion

- The existing initialization methods normally suffers from nonsmoothness of neural learning, and it is hardly to be **valided** in theory.
- In this work, a generalization framework to initialization of neural learning is presented, and the dilemma can be alleviated.
- Compared with existing solutions, the SINL approach is able to initialize the networks with **smooth** learning.

Thank You

ICBDC '24, Bangkok, Thailand