NEURAL INFORMATION PROCESSING SYSTEMS

LUCI
Lab of Use-inspired Computational Intelligence

R·I·T
B. THOMAS GOLISANO
College of COMPUTING AND INFORMATION SCIENCES

Joint Inference for Neural Network Depth and Dropout Regularization

Kishan K C[1], Rui Li[1], Mahdi Gilany[2]

[1] Rochester Institute of Technology

[2] Queens University

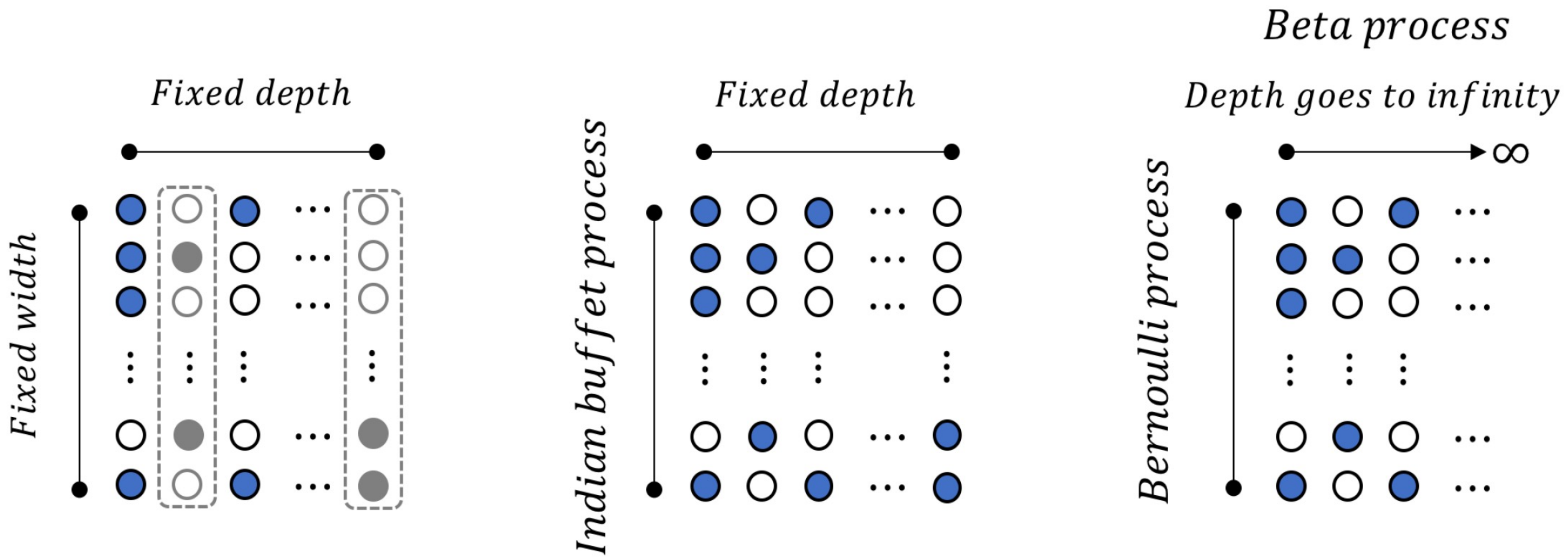
35th Conference on Neural Information Processing Systems (NeurIPS 2021)

Motivation

- Pre-determined backbone structures are the key
- Deep networks tend to be
  - Overfitted
  - Poorly calibrated with high confidence on incorrect predictions (Nguyen et al. 2015, Antorán et al. 2020)
- Current solutions
  - Dropout and its variants (Srivastava et al. 2014, Gal et al. 2017, Lee et al. 2019)
  - Structure selection methods (Srinivas et al. 2016, Dikov et al. 2019, Antorán et al. 2020)
- However,
  - Cannot scale the network beyond the pre-determined structure
  - Cannot achieve a balance between network depth and dropout regularization for uncertainty calibration

Our proposed solution

- Model the depth (number of hidden layers) as a Beta Process
- Modulate neuron activations with a conjugate Bernoulli Process
- Joint inference of network depth and neuron activations



A typical structure selection method

A dropout variants

Our proposed solution

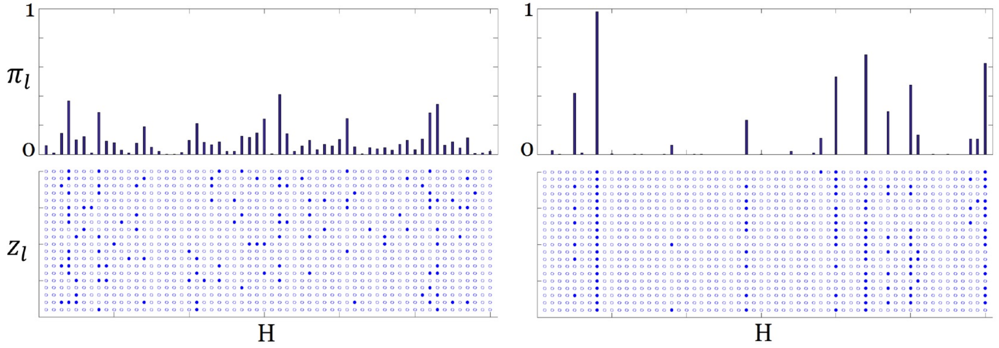
Beta-Bernoulli process over network structures

- Model the depth of a neural network as a Beta Process
  - Stick breaking construction of beta-Bernoulli Process (Paisley et al. 2010, Broderick et al. 2012)

$$\nu_l \sim \mathrm{Beta}(\alpha, \beta), \qquad \pi_l = \prod_{j=1}^{l} \nu_j, \qquad z_{ml} \sim \mathrm{Bernoulli}(\pi_l)$$

  - The prior over the network structures $\mathbf{Z}$:

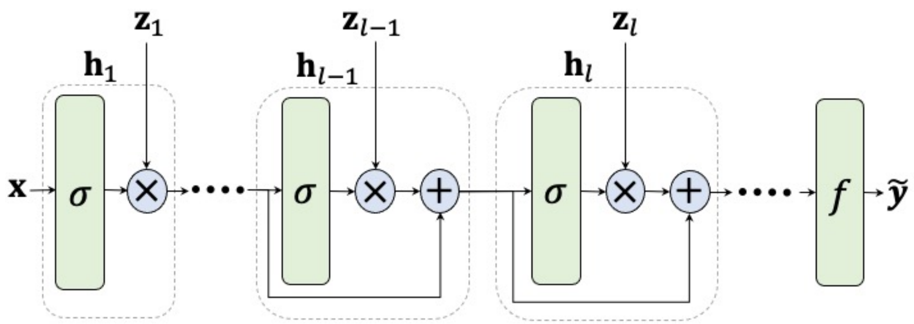
$$p(\mathbf{Z}, \boldsymbol{\nu} | \alpha, \beta) = p(\boldsymbol{\nu} | \alpha, \beta) p(\mathbf{Z} | \boldsymbol{\nu}) = \prod_{l=1}^{\infty} \mathrm{Beta}(\nu_l | \alpha, \beta) \prod_{m=1}^{M} \mathrm{Bernoulli}(z_{ml} | \pi_l)$$



$\alpha > \beta$ $\qquad$ $\alpha < \beta$

Network structure with infinite layers

- A neural network has the form



$$h_l = \sigma(\mathbf{W}_l \mathbf{h}_{l-1}) \otimes \mathbf{z}_l + \mathbf{h}_{l-1} \qquad l \in \{1,2,\ldots,\infty\}$$

- A Gaussian likelihood of the neural network for regression task

$$p(D|\mathbf{Z},\mathbf{W}) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{y}_n | f(\mathbf{x}_n; \mathbf{Z}, \mathbf{W}), s^2\mathbf{I})$$

Efficient inference

- The marginal Likelihood over network structures $\mathbf{Z}$ is

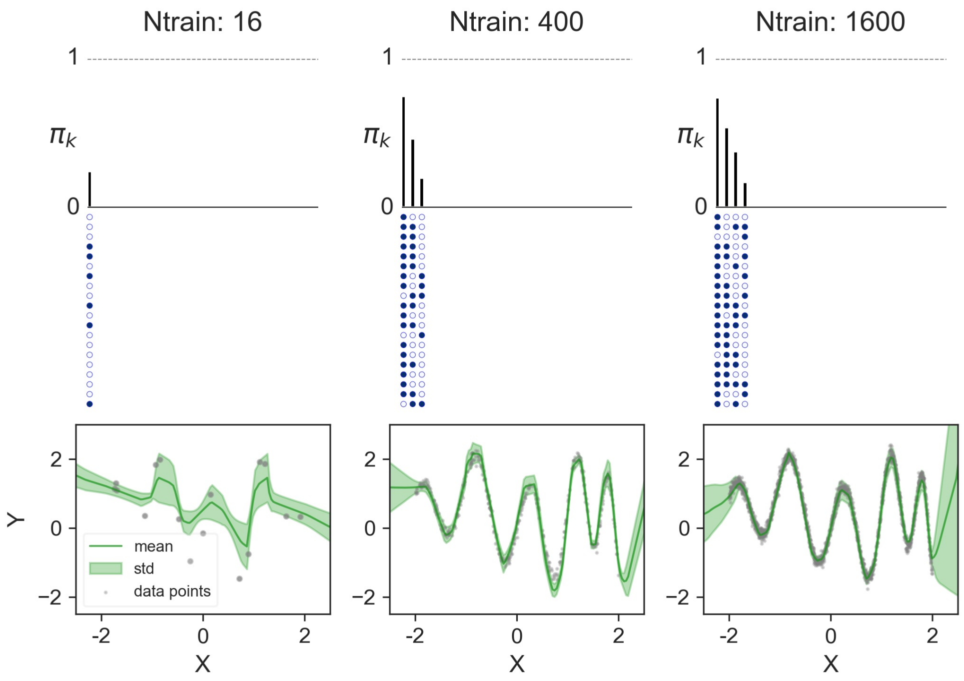
$$\mathrm{p}(\mathrm{D}|\mathbf{W}, L, \alpha, \beta) = \int \mathrm{p}(\mathrm{D}|\mathbf{Z}, \mathbf{W})\ \mathrm{p}(\mathbf{Z}, \boldsymbol{\nu}|\alpha, \beta)\ \mathrm{d}\mathbf{Z}\mathrm{d}\boldsymbol{\nu}$$

- Approximation with structured stochastic variational inference (Hoffman et al. 2013, 2015)

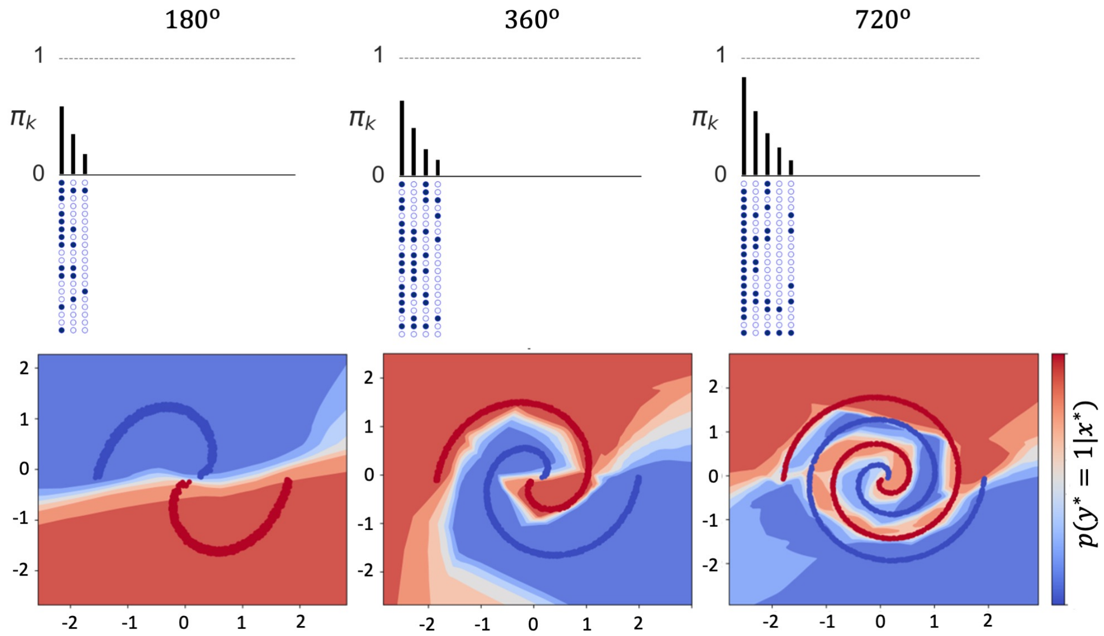
$$\log \mathrm{p}(\mathrm{D}|\mathbf{W}, L, \alpha, \beta) \geq \mathbb{E}_{q(\mathbf{Z},\boldsymbol{\nu})}[\log p(D|\mathbf{Z}, \mathbf{W})] - \mathrm{KL}[q(\mathbf{Z}|\boldsymbol{\nu})||p(\mathbf{Z}|\boldsymbol{\nu})] - \mathrm{KL}[q(\boldsymbol{\nu})||p(\boldsymbol{\nu})]$$

  - We use truncation level K for the variational distribution
  - Reparameterization of Beta and Bernoulli distribution (Jang et al. 2017, Maddison et al. 2017, Jankowiak et al. 2018)
  - We prove that optimizing ELBO is equivalent to Bayesian Information Criterion over the structure Z

Performance evaluation on synthetic data



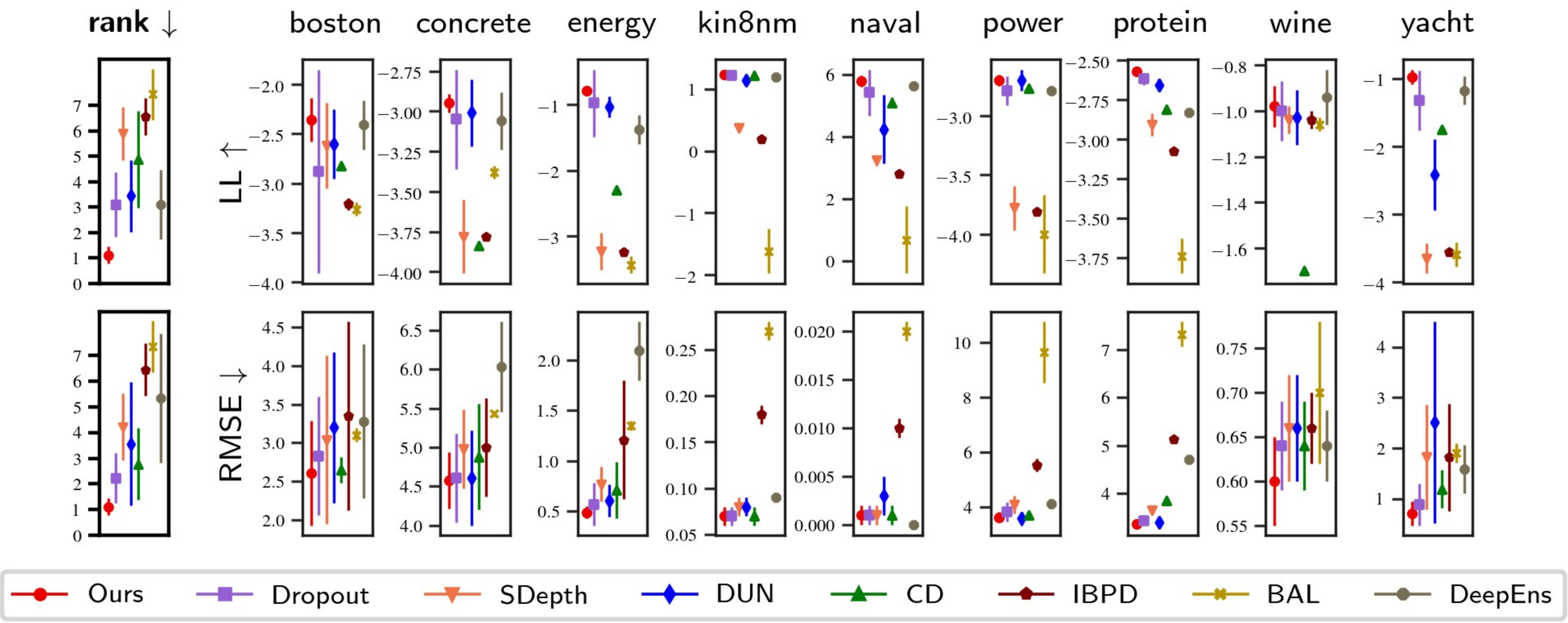
Periodic dataset
(Increasing data sizes)



Spirals dataset
(Increasing complexity)

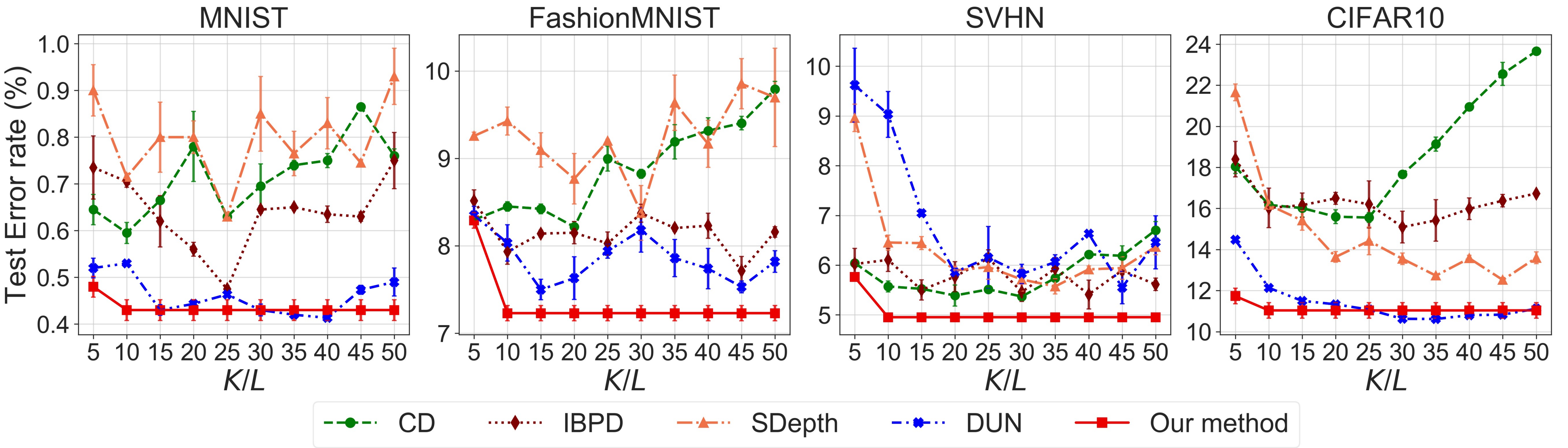
- If the training data size or its complexity increases, network structure grows to accommodate more information.

Performance comparison on UCI datasets



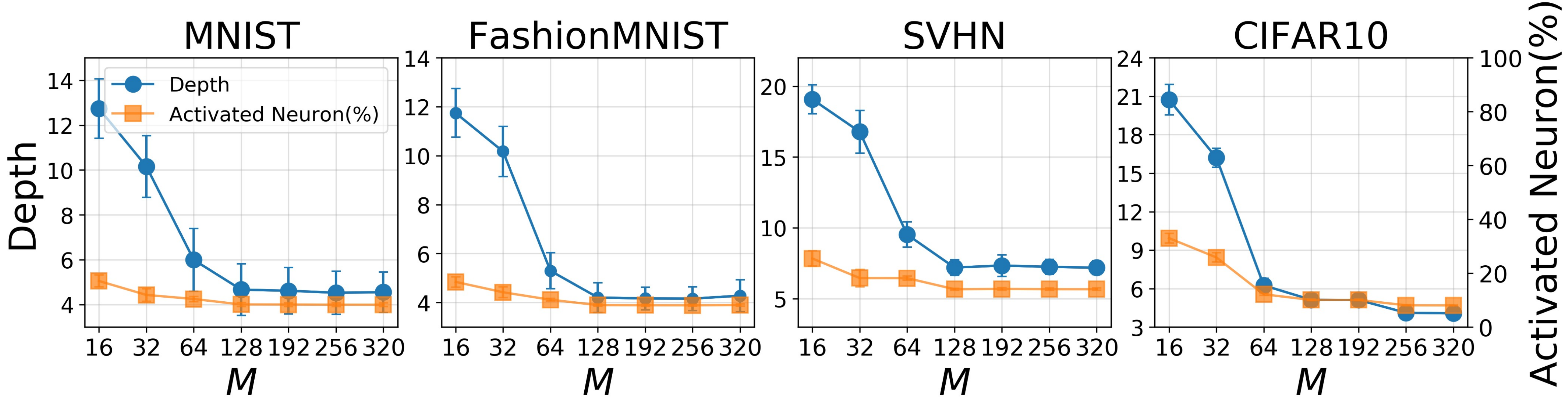
- Our method achieves the overall highest rank for both uncertainty calibration and prediction accuracy.

Effect of truncation level K



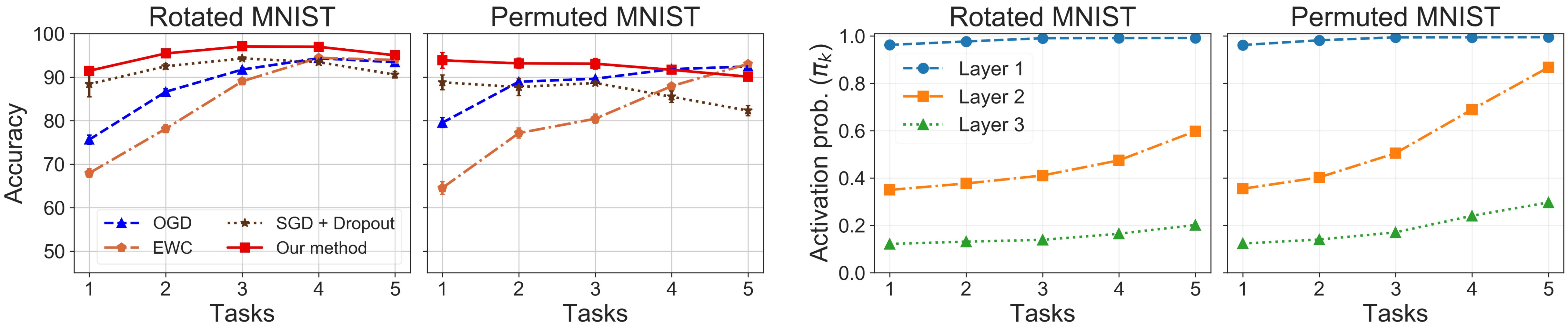
- The truncation level (K) of our method does not affect the performance.
- The depth (L) of other methods significantly affects the performance and should be set carefully.

Effect of maximum width M



- With smaller width M, our method results in deeper network structures to compensate for the relatively narrow layers.
- As M increases, the structures become shallower.

Case study on Continual learning



- Our method alleviates catastrophic forgetting by enabling network depth to dynamically augment to accommodate incrementally available information.

Conclusion

- General joint inference framework applicable for various neural networks
- Experimental results on MLPs and CNNs show that our method achieves superior performance by adapting network depth and neuron activations
- Our model can accommodate incrementally available information by enabling neural network structures to dynamically evolve

Thank you!