CVPR 2021 Tutorial

Normalization Techniques in Deep Learning: Methods, Analyses and Applications

Lei Huang

Beihang University, Beijing, China

北京航空航天大学人工智能研究院
Institute of Artificial Intelligence, Beihang University

2021-06-19

软件开发环境国家重点实验室
State Key Laboratory of Software Development Environment

Outline

01. Motivations of Normalization Techniques

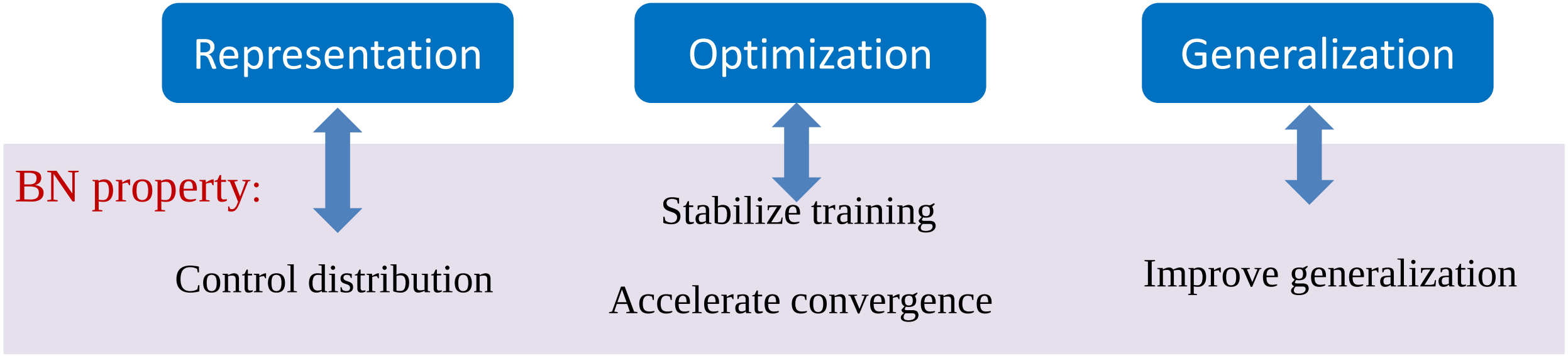
02. Introduction of Normalization Methods

03. Analyses of Normalization

04. Applications of Normalization

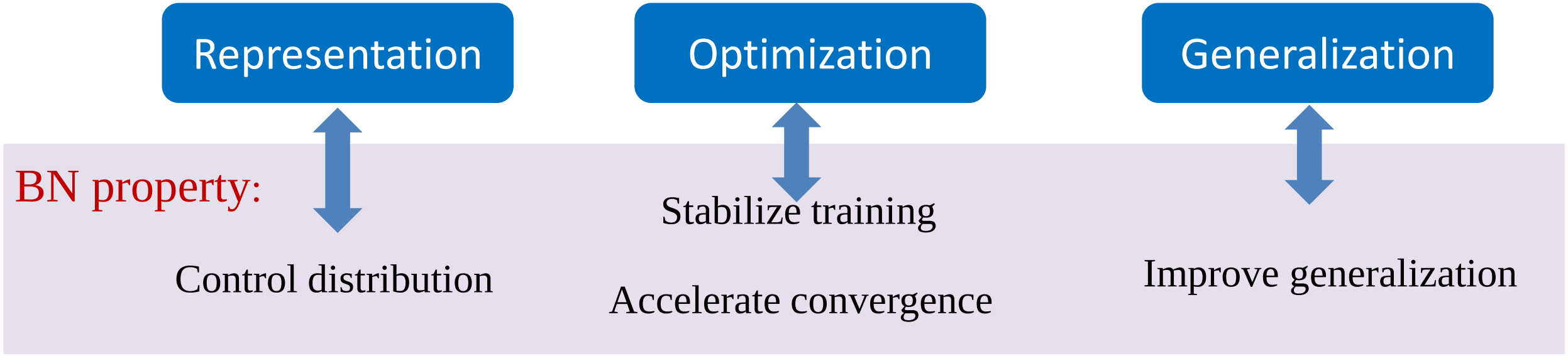
Analyses of Normalization

- Topics of deep learning theory



Analyses of Normalization

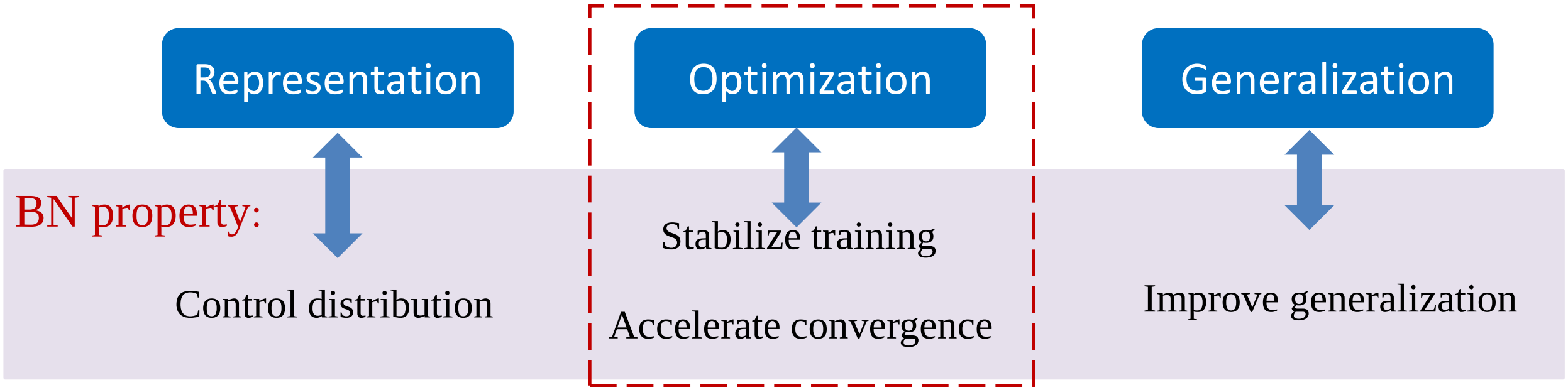
- Topics of deep learning theory



- Disclaims in this tutorials:
  - Not theoretic analysis in generalization bound or optimization theory
  - But empirical analysis with high-level theoretic insights.

Analyses of Normalization

- Topics of deep learning theory



- Disclaims in this tutorials:
  - Not theoretic analysis in generalization bound or optimization theory
  - But empirical analysis with high-level theoretic insights.

Scale Invariant Analysis

- Scale invariant for weight in a normalization layer

$$BN(\mathbf{x}; a\boldsymbol{W}) = BN(\mathbf{x}; \boldsymbol{W})$$

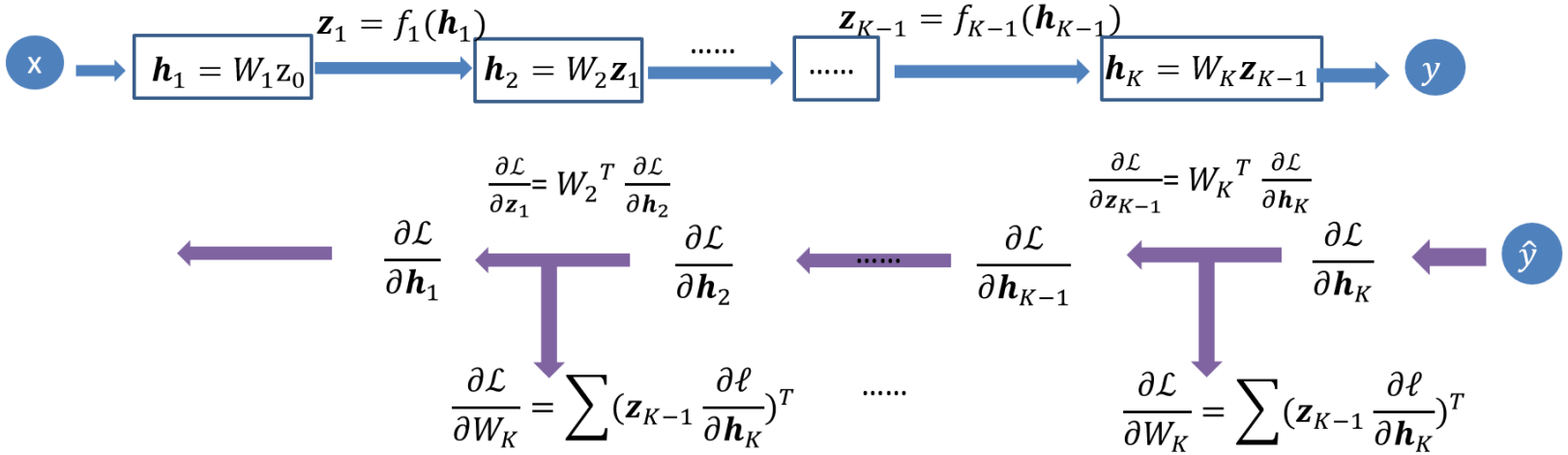
$$\frac{\partial BN(\mathbf{x}; a\boldsymbol{W})}{\partial (a\boldsymbol{W})} = \frac{1}{a} \frac{\partial BN(\mathbf{x}; \boldsymbol{W})}{\partial \boldsymbol{W}},$$

Applies also for LN or weight normalization

[Ioffe and Szegedy, ICML 2015; Ba et al, 2016]

Scale Invariant Analysis

- Stabilize training in a network
  - Scale dynamics for a network



$$\hat{W}_k = \alpha_k W_k$$

	Layer input	Output-gradient	Weight-gradient
Un-normalized	$\hat{x}_k = (\prod_{i=1}^{k} \alpha_i) x_k$	$\frac{\partial L}{\partial \hat{h}_k} = (\prod_{i=k+1}^{K} \alpha_i) \frac{\partial L}{\partial h_k}$	$\frac{\partial L}{\partial \hat{W}_k} = (\prod_{i=1, i \neq k}^{K} \alpha_i) \frac{\partial L}{\partial W_k}$
normalized	$\hat{x}_k = x_k$	$\frac{\partial L}{\partial \hat{h}_k} = \frac{1}{\alpha_k} \frac{\partial L}{\partial h_k}$	$\frac{\partial L}{\partial \hat{W}_k} = \frac{1}{\alpha_k} \frac{\partial L}{\partial W_k}$

Layer-wise Conditioning Analysis in Exploring the Learning Dynamics of DNNs [Huang et al, ECCV 2020]

Scale Invariant Analysis

- The auto-tuning effects of learning rate

Scale invariant

$$f(\lambda \boldsymbol{w}) = f(\boldsymbol{w}) \qquad \frac{\partial f(\lambda \boldsymbol{w})}{\partial \lambda \boldsymbol{w}} = \frac{1}{\lambda}\frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}}$$

$$< \boldsymbol{w}, \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}} > = 0$$

$$\| \boldsymbol{w_{t+1}} \|_2^2 = \| \boldsymbol{w_t} \|_2^2 + \eta^2 \| \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}} \|_2^2$$

Weight update:

$$\boldsymbol{w_{t+1}} = \boldsymbol{w_t} + \eta \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}}$$

$\boldsymbol{w}$ ↑ $\qquad \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}}$ ↓

[Neyshabur et al, ICLR 2016; Salimans et al, NeurIPS 2016; Sun et al, AAAI 2020]

Scale Invariant Analysis

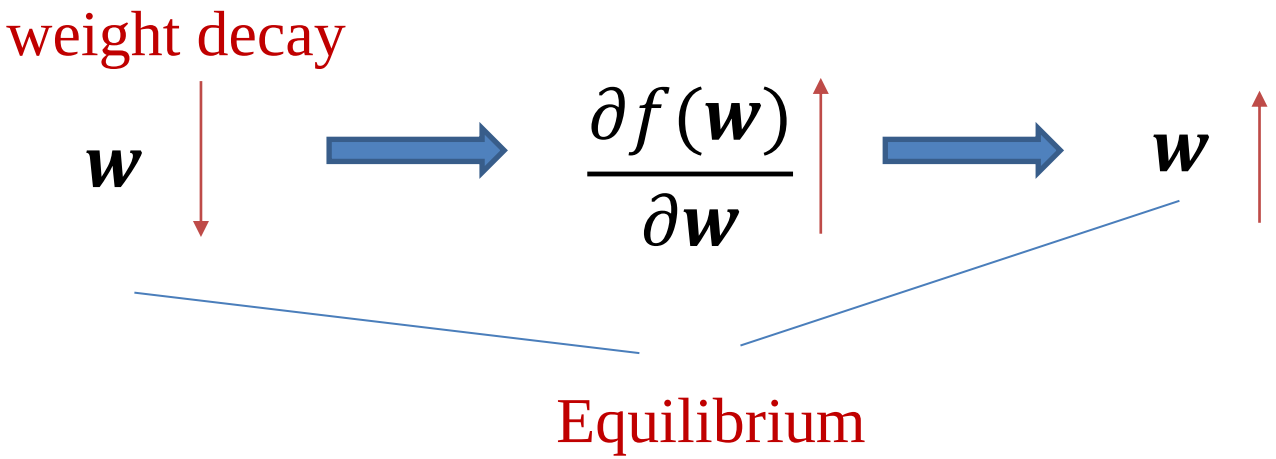
- The auto-tuning effects + weight decay

$$f(\lambda \boldsymbol{w}) = f(\boldsymbol{w})$$

$$\frac{\partial f(\lambda \boldsymbol{w})}{\partial \lambda \boldsymbol{w}} = \frac{1}{\lambda}\frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}}$$

$$< \boldsymbol{w}, \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}} > = 0$$

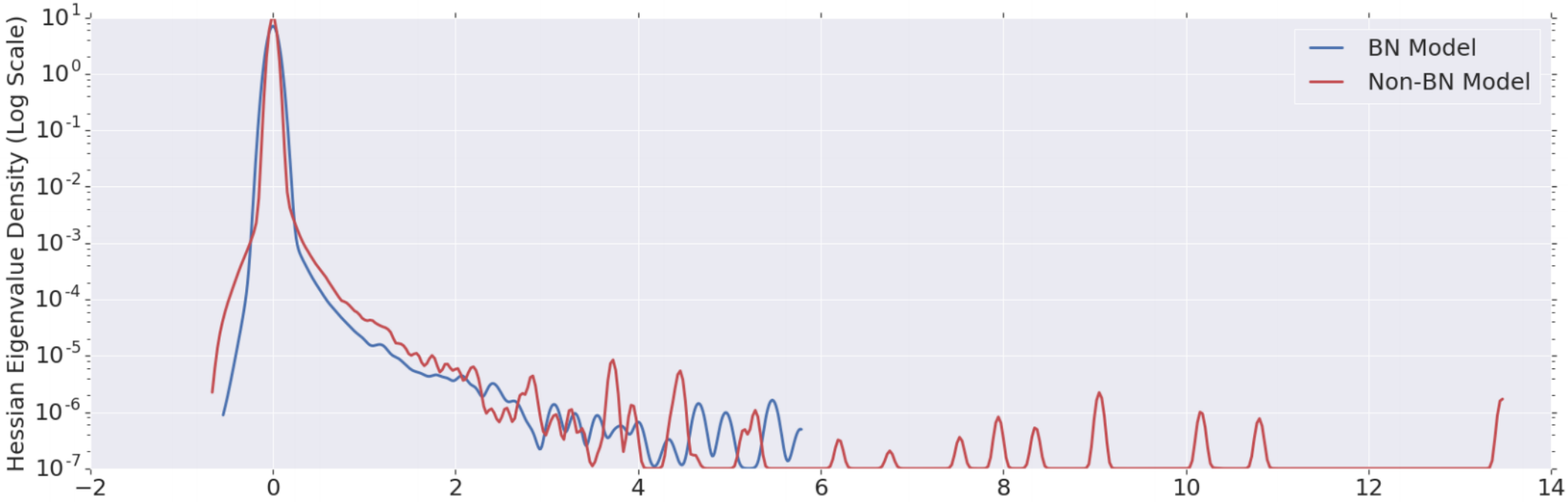
$$\| \boldsymbol{w_{t+1}} \|_2^2 = \| \boldsymbol{w_t} \|_2^2 + \eta^2 \| \frac{\partial f(\boldsymbol{w})}{\partial \boldsymbol{w}} \|_2^2$$



[Laarhoven et al, 2017; Huang et al, 2017; Arora et al, ICLR 2019]

Improved Conditioning in Optimization

- The full Hessian spectrum

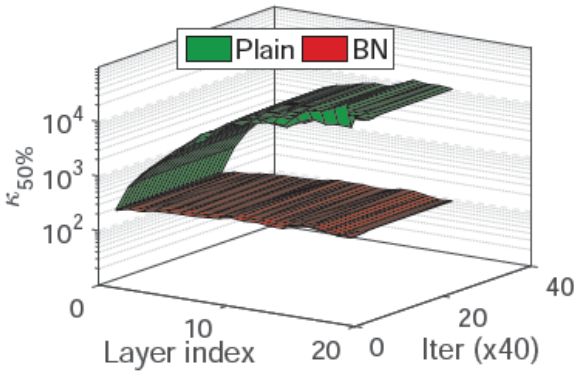


The eigenvalue comparison of the Hessian of the VGG network with BN (blue) and without BN (red)

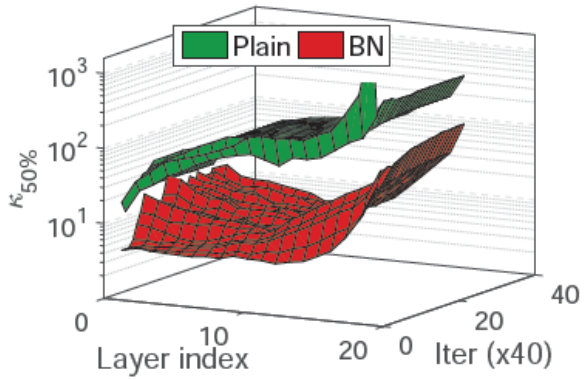
An Investigation into Neural Net Optimization via Hessian Eigenvalue Density [Ghorbani et al, ICML 2019]

Improved Conditioning in Optimization

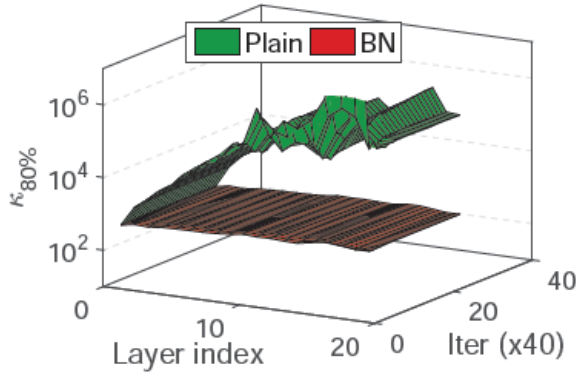
- The Layer-wise spectrum
  - Covariance of layer input Σ_x
  - Covariance of output-gradient $\Sigma_{\nabla h}$
  - Criteria 2: Σ_x and $\Sigma_{\nabla h}$ are well conditioned (in layer)



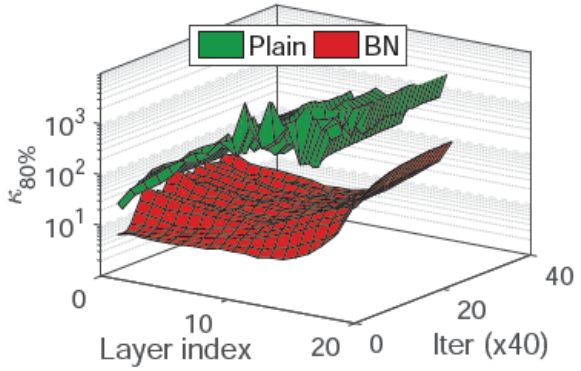
(a) $\kappa_{50\%}(\Sigma_x)$



(b) $\kappa_{50\%}(\Sigma_{\nabla \mathbf{h}})$



(c) $\kappa_{80\%}(\Sigma_x)$



(d) $\kappa_{80\%}(\Sigma_{\nabla \mathbf{h}})$

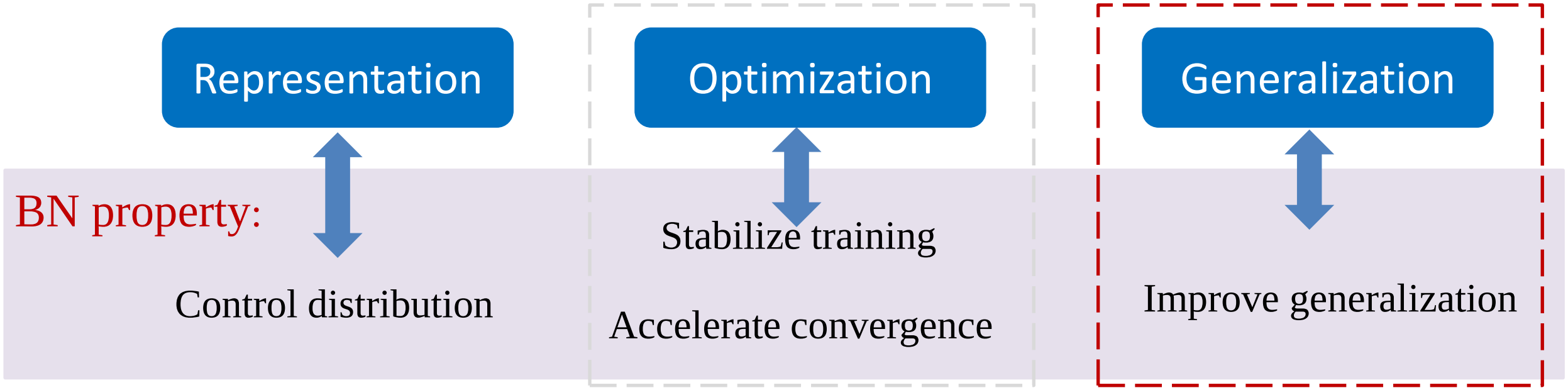
Layer-wise Conditioning Analysis in Exploring the Learning Dynamics of DNNs [Huang et al, ECCV 2020]

Theoretical Analysis under Assumptions

- Mean field theory of batch normalization
  - Yang et al, ICLR 2019
  - Wei et al, 2019
- Theoretical accelerated converge of BN on learning half-space with Gaussian inputs [Kohler et al, AISTATS 2019]
- Global convergence results for two-layer NN with ReLU and weight normalization [Dukler et al, ICML 2020]

Analyses of Normalization

- Topics of deep learning theory



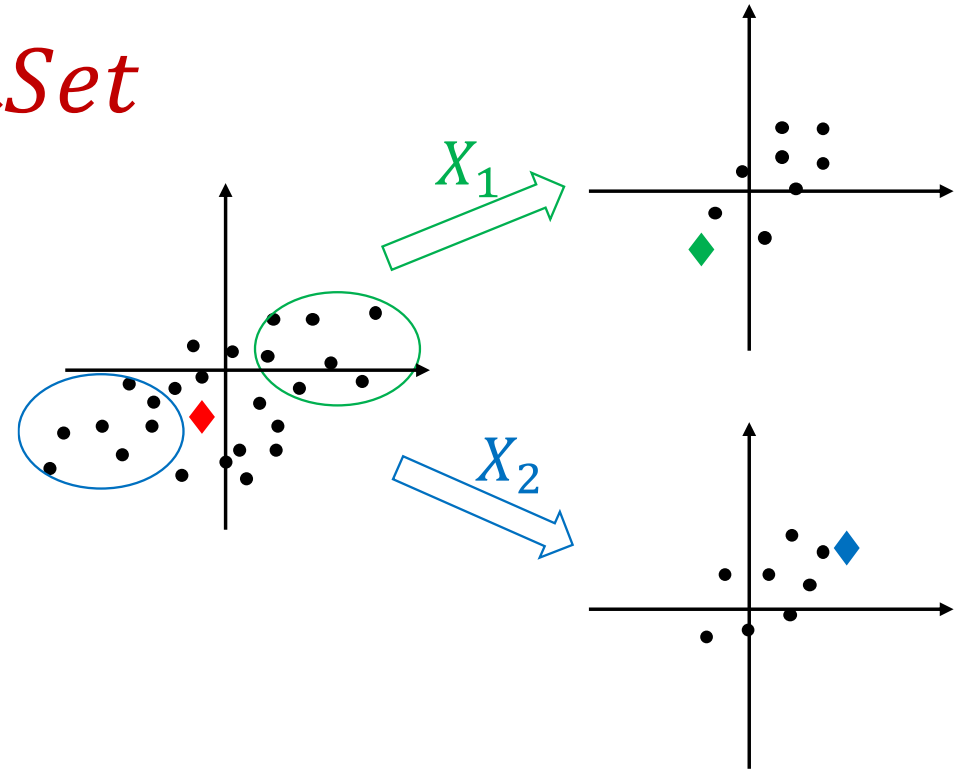
- Disclaims in this tutorials:
  - Not theoretic analysis in generalization bound or optimization theory
  - But empirical analysis with high-level theoretic insights.

The stochasticity

- Introduced Stochasticity
  - During training

$$X_m \sim DataSet$$

$$\hat{X}_m = \frac{X_m - \mu_{X_m}}{\sigma_{X_m}}$$



  - During inference

Population statistics $\{\hat{\mu}, \hat{\sigma}\}$:

$$\hat{\mu} = (1-\lambda)\hat{\mu} + \lambda\mu_{X_m}$$

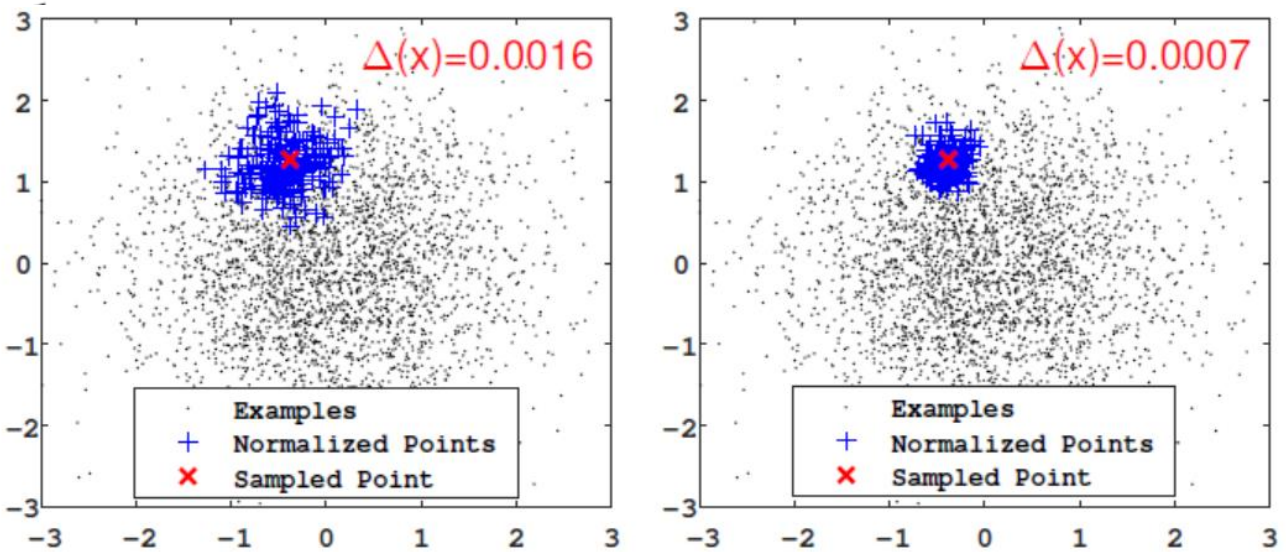
$$\hat{\sigma} = (1-\lambda)\hat{\sigma} + \lambda\sigma_{X_m}$$

Quantitively Evaluation

- Stochastic Normalization Disturbance (SND)

Empirical SND: $\widehat{\mathbf{\Delta}}_{\mathbf{G}}(\mathbf{x}) = \frac{1}{s}\sum_{i=1}^{s} \|\mathbf{G}(\mathbf{X}_i^B;\mathbf{x}) - \frac{1}{s}\sum_{j=1}^{s}\mathbf{G}(\mathbf{X}_j^B;\mathbf{x})\|$

$\{\mathbf{X}_j^B\}_{j=1}^{s}$ ~ Dataset



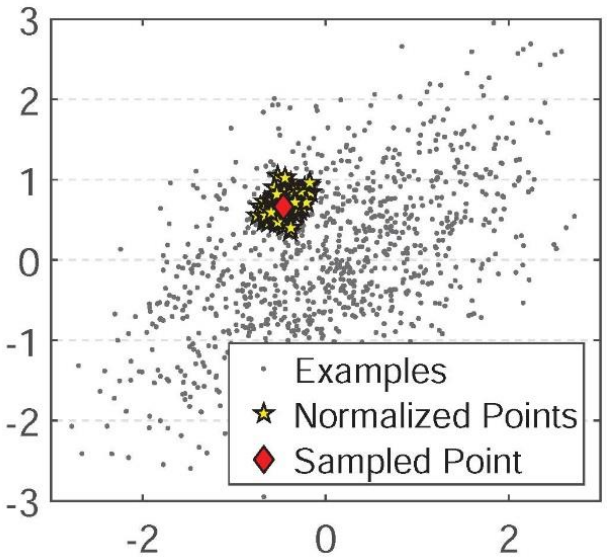
(a) batch size of 16 (b) batch size of 64

Iterative Normalization: Beyond Standardization towards Efficient Whitening [Huang et al, CVPR 2019]

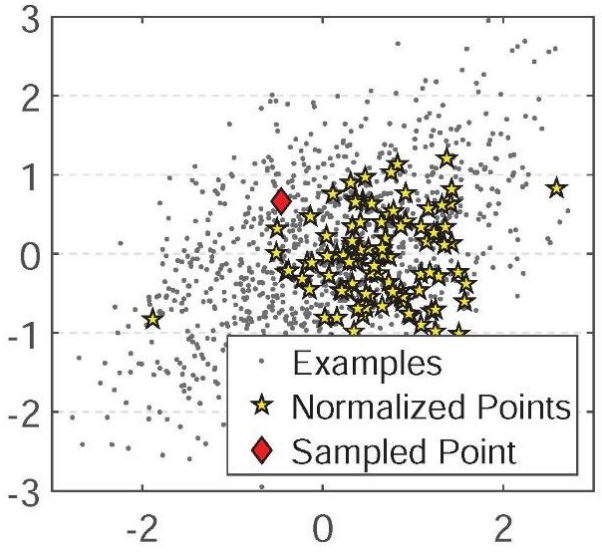
Qualitatively Visualization

- Stochastic outputs combining different mini-batches for different batch whitening

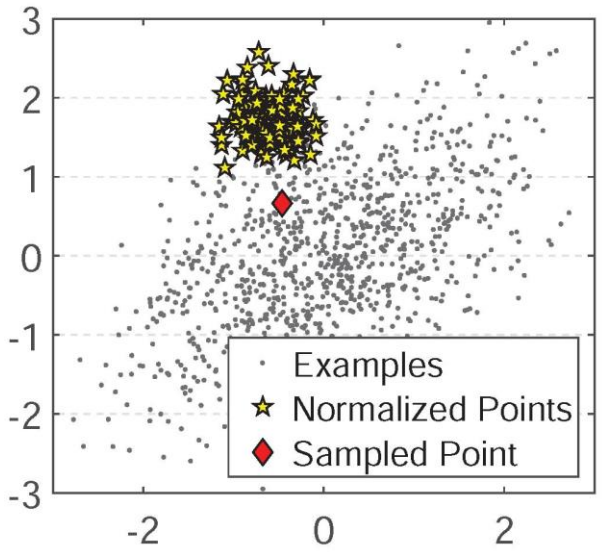
BN



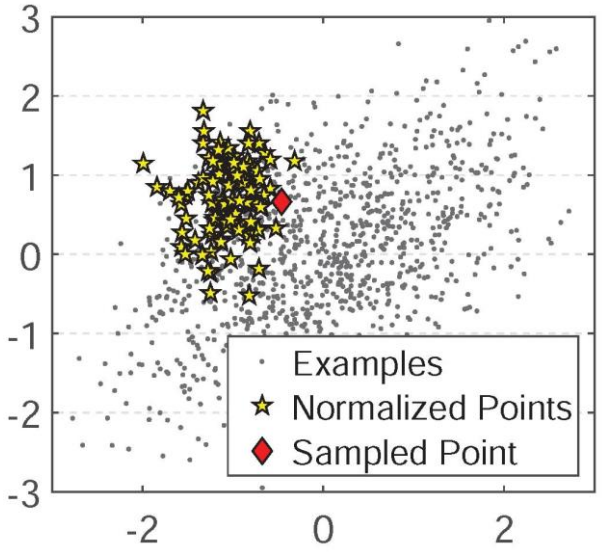
PCA



ZCA



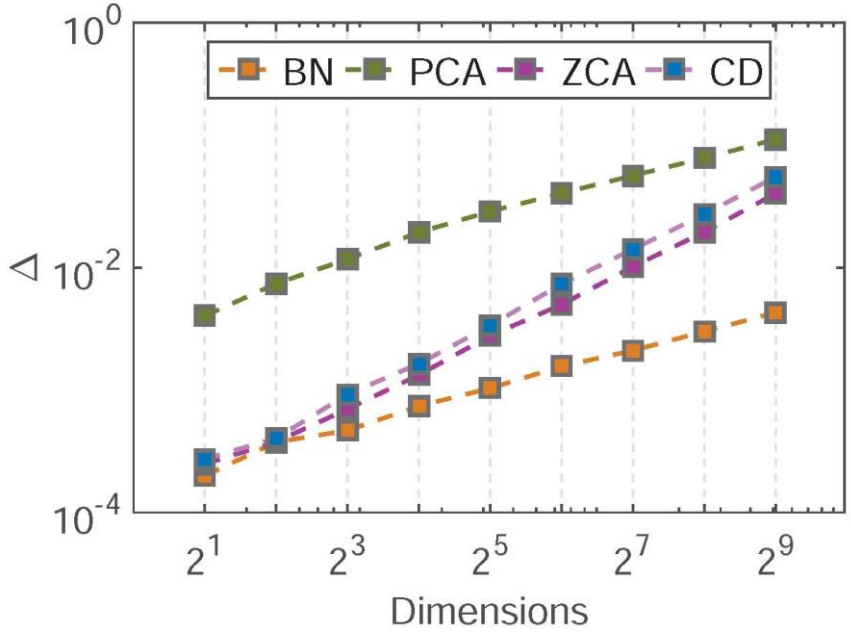
CD

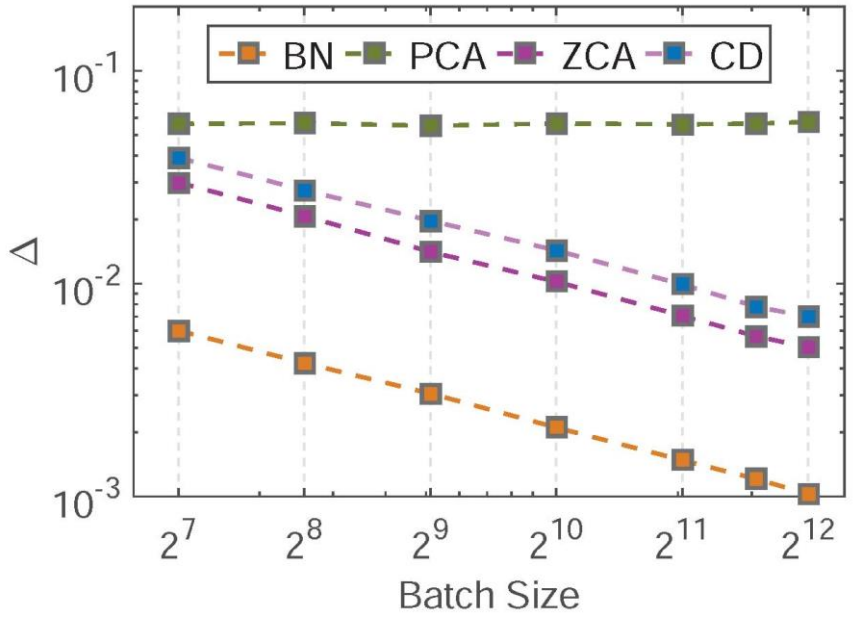


An Investigation into the Stochasticity of Batch Whitening [Huang et al, CVPR 2020]

Investigation into the Stochasticity

- Stochasticity related to batch size and dimension
  - Increased SND with increasing dimension and decreasing batch size
  - SND: PCA>CD>ZCA>BN
  - PCA has nearly same large SND among batches

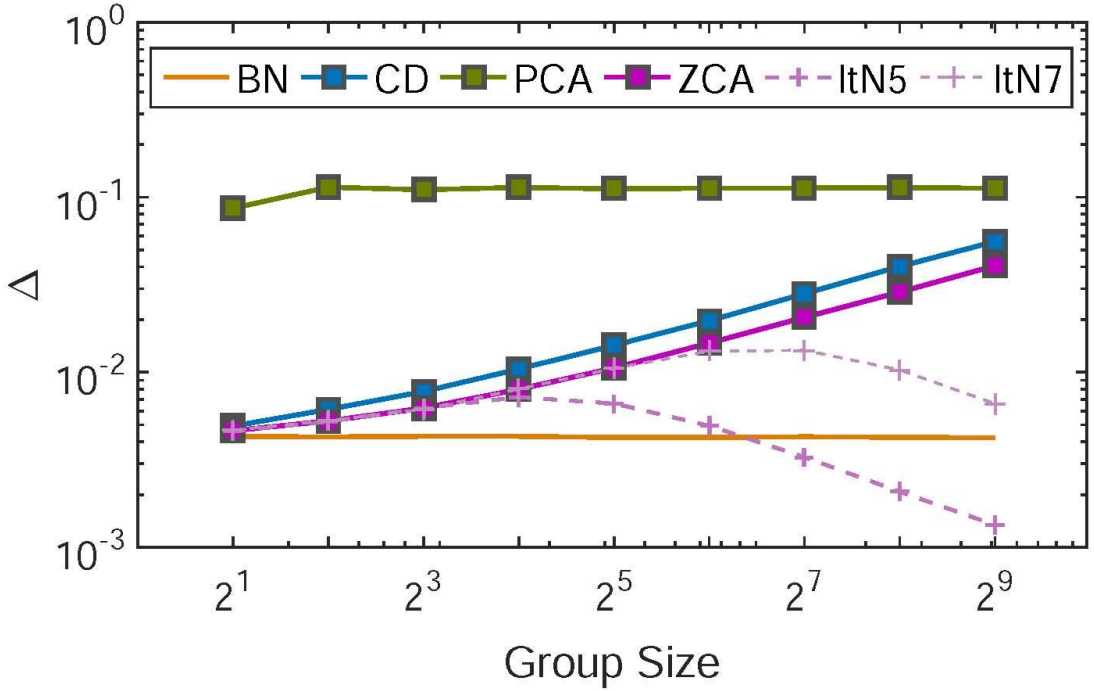


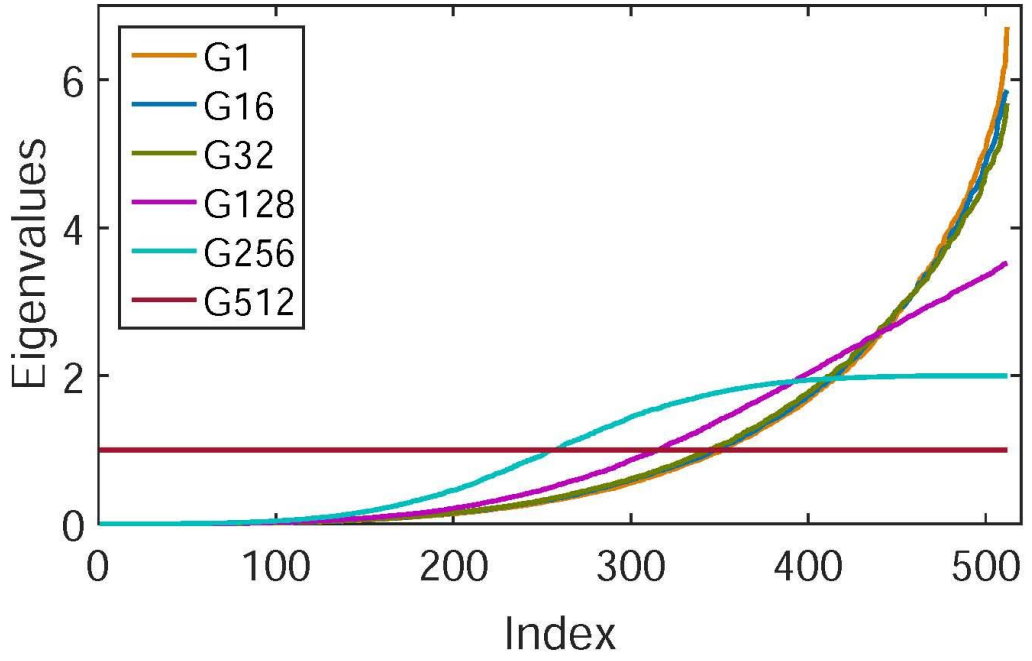


An Investigation into the Stochasticity of Batch Whitening [Huang et al, CVPR 2020]

Investigation into the Stochasticity

- Controlling the stochasticity of BW by groups





An Investigation into the Stochasticity of Batch Whitening [Huang et al, CVPR 2020]

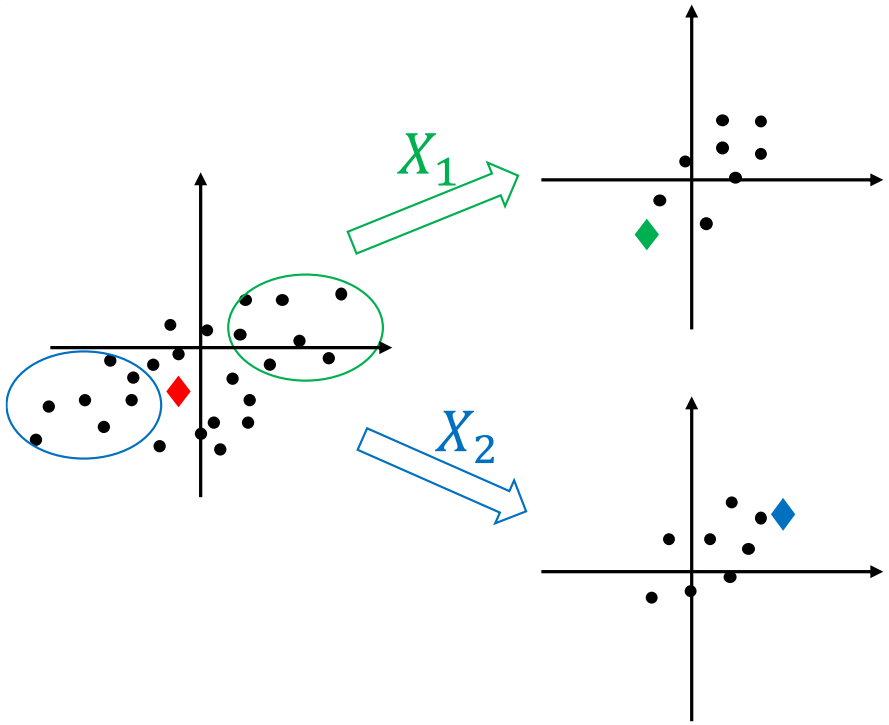
Theoretical Model

- **Assumption**: X is approximately normal distribution with population statistics $(\hat{\mu}, \hat{\sigma})$, X_m is i.i.d sampled

Consider an example x,
mini-batch output:

$$\frac{x-\mu_{X_m}}{\sigma_{X_m}} = \left(\frac{x-\hat{\mu}}{\hat{\sigma}} + \frac{\hat{\mu}-\mu_{X_m}}{\hat{\sigma}}\right)\frac{\hat{\sigma}}{\sigma_{X_m}}$$

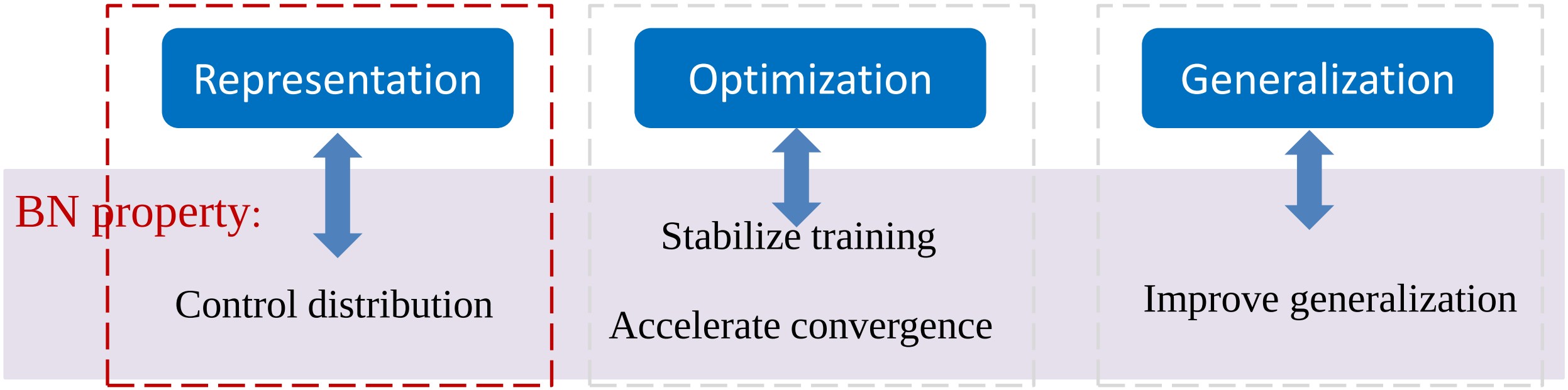
$$\frac{\hat{\mu}-\mu_{X_m}}{\hat{\sigma}} \sim \frac{1}{\sqrt{m}}\mathcal{N}(0,1) \qquad \frac{\hat{\sigma}}{\sigma_{X_m}} \sim \sqrt{m}\chi^{-1}_{m-1}$$



[Teye et al, ICML 2018; Shekhovtsov and Flach, ACCV 2018; Yong et al, ECCV 2020]

Analyses of Normalization

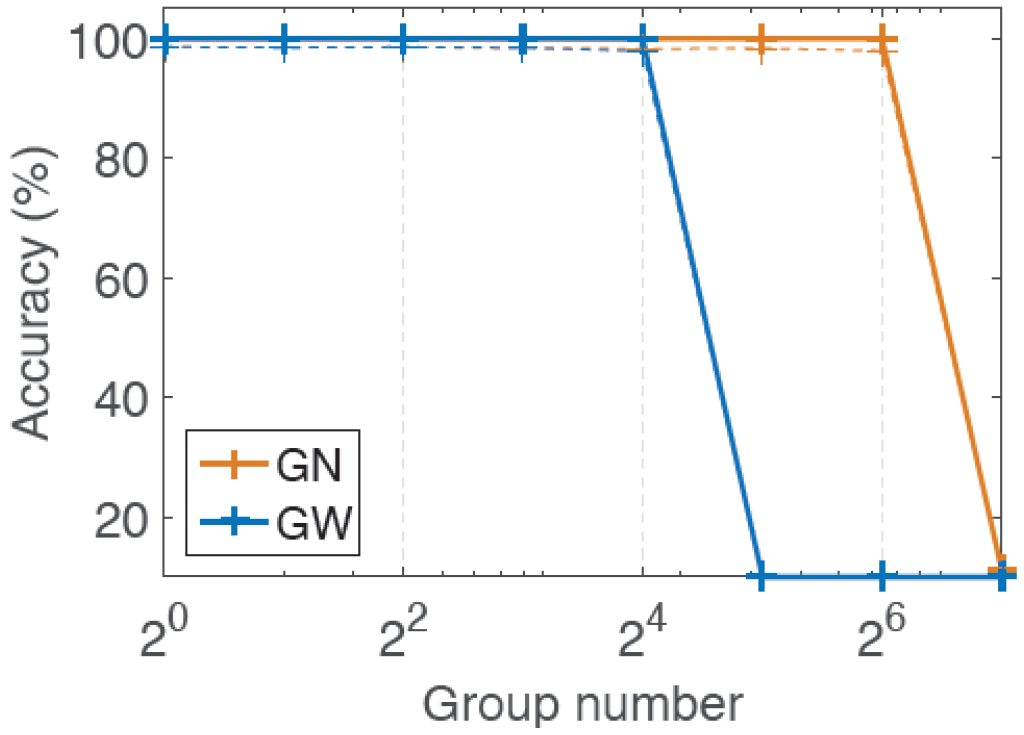
- Topics of deep learning theory



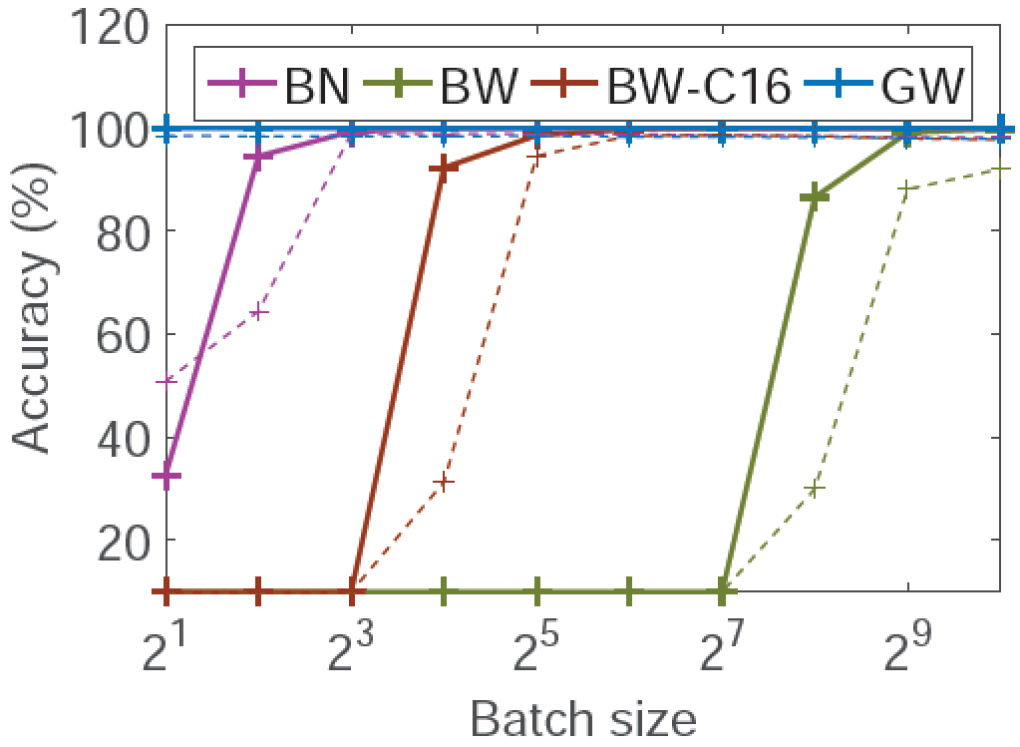
- Disclaims in this tutorials:
  - Not theoretic analysis in generalization bound or optimization theory
  - But empirical analysis with high-level theoretic insights.

Revisiting the Constraint of Normalization

- The large group problem of GN



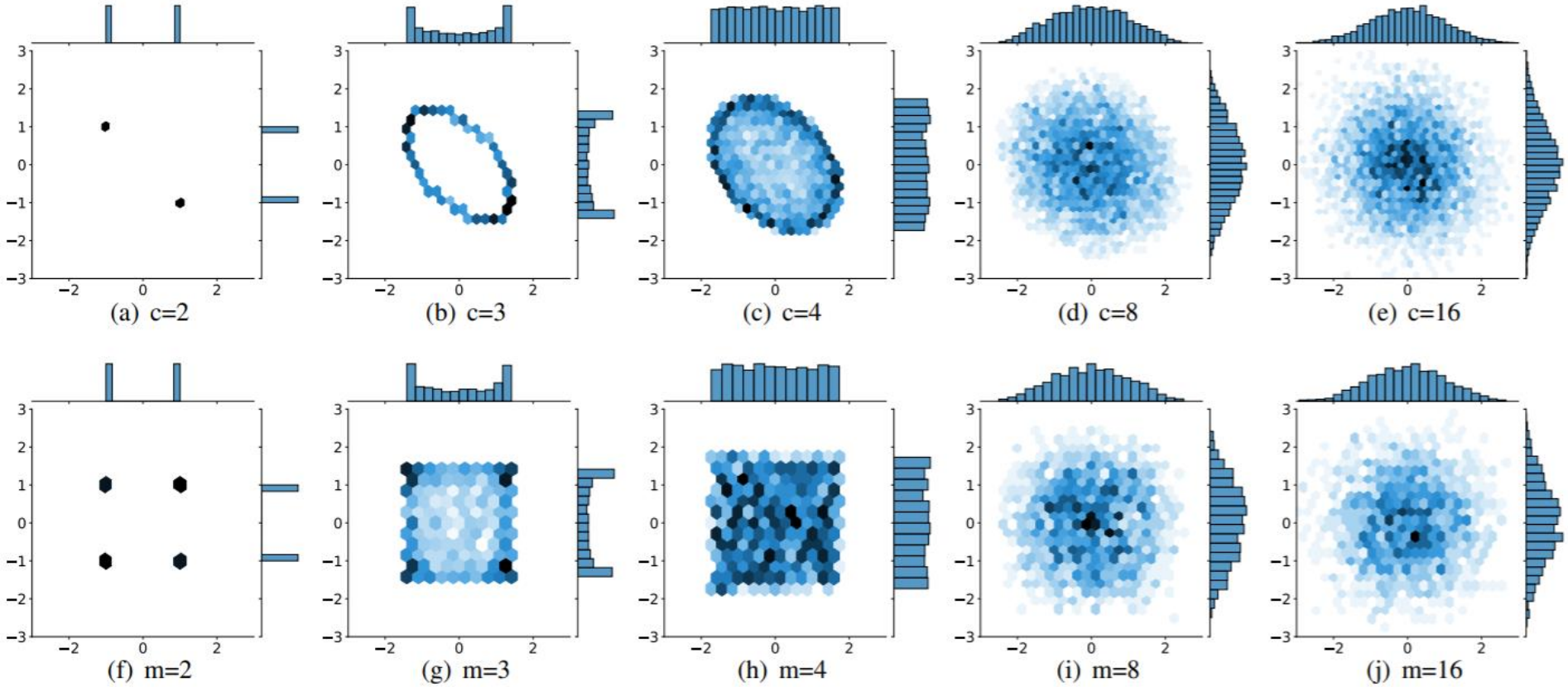
- The small-batch problem of BN/BW



Group Whitening: Balancing Learning Efficiency and Representational Capacity [Huang et al, CVPR 2021]

Revisiting the Constraint of Normalization

- Feature representation in a layer



Group Whitening: Balancing Learning Efficiency and Representational Capacity [Huang et al, CVPR 2021]

Revisiting the Constraint of Normalization

- Constraints as a system of equations: $\Upsilon_\phi(\hat{X})$

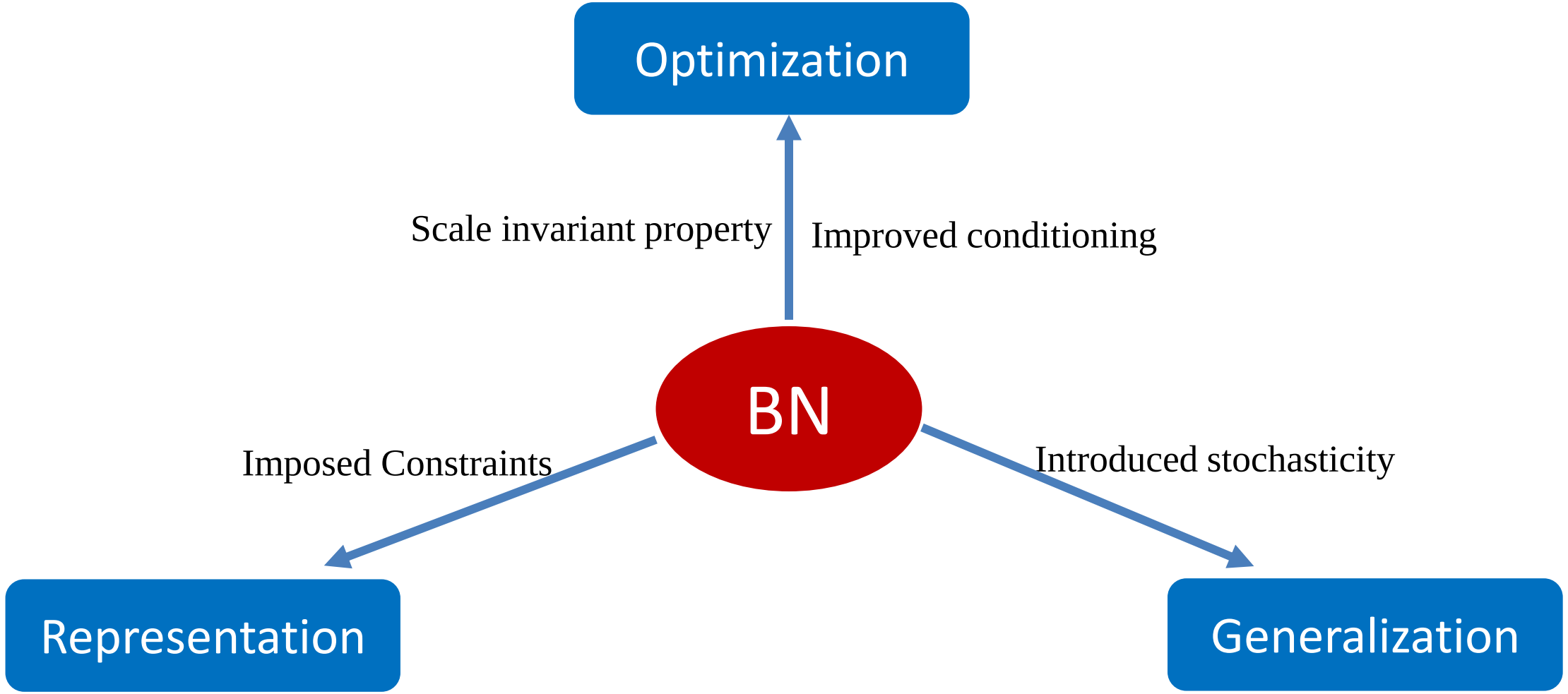
$$\Upsilon_{\phi_{BN}}(\hat{X}): \sum_{j=1}^{m} \hat{X}_{ij} = 0 \; and \; \sum_{j=1}^{m} \hat{X}_{ij}{}^2 - m = 0$$

- Constrain number $\zeta(\phi; X)$: the number of equations in $\Upsilon_\phi(\hat{X})$

	Normalization along a batch		Normalization along a group of neurons	
	BN	BW	GN	GW
$\zeta(\phi; \mathbf{X})$	$2d$	$\frac{d(d+3)}{2}$	$2gm$	$\frac{mg(g+3)}{2}$
$\zeta(\phi; \mathbb{D})$	$\frac{2Nd}{m}$	$\frac{Nd(d+3)}{2m}$	$2gN$	$\frac{Ng(g+3)}{2}$
Ranges of m/g	$m \geq 2$	$m \geq \frac{d+3}{2}$	$g \leq \frac{d}{2}$	$g \leq \frac{\sqrt{8d+9}-3}{2}$

Group Whitening: Balancing Learning Efficiency and Representational Capacity [Huang et al, CVPR 2021]

Summary


Optimization
Scale invariant property
Improved conditioning
BN
Imposed Constraints
Introduced stochasticity
Representation
Generalization

Challenges and Opportunities

- Theoretic analysis of DNN itself is difficult
- Normalizing activations make theoretic analysis of DNN challenging
  - Ruin the assumption of representation analysis tools
  - Data dependent
- Normalizing weight attribute to theoretic analysis of DNN
  - Pros: Lipschitz constant can be controlled/bounded during training by normalizing the weight
    - Certificated defense against adversarial attacks [Tsuzuku et al, NeurIPS 2018]
    - Theoretically analyzing DNNs generalization [Bartlett et al, NeurIPS 2017]]
  - Challenges: normalizing weights is still not as effective as normalizing activations in practice [Huang et al, CVPR 2020; Brock et al, ICLR 2021]