PAC-Bayes Information Bottleneck

Zifeng Wang[1,2], Shao-Lun Huang[1], Ercan E Kuruoglu[1], Jimeng Sun[2], Xi Chen[3], Yefeng Zheng[3]

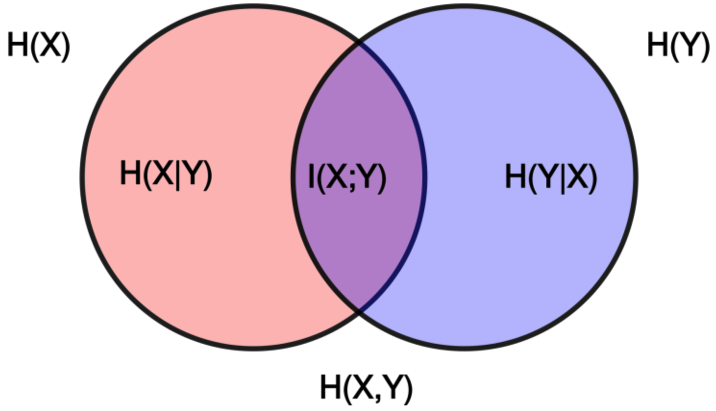
[1]UIUC, [2]Tsinghua University, [3]Tencent

ICLR 2022

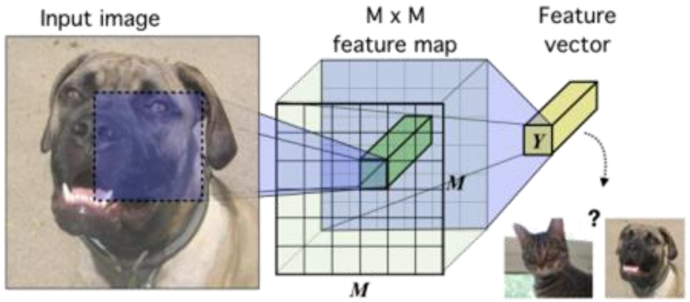
Background: Information in neural networks?

Mutual information:

$$I(X;Y) = \iint p(X)p(Y|X)\log\frac{p(X,Y)}{p(X)p(Y)}dX\,dY$$

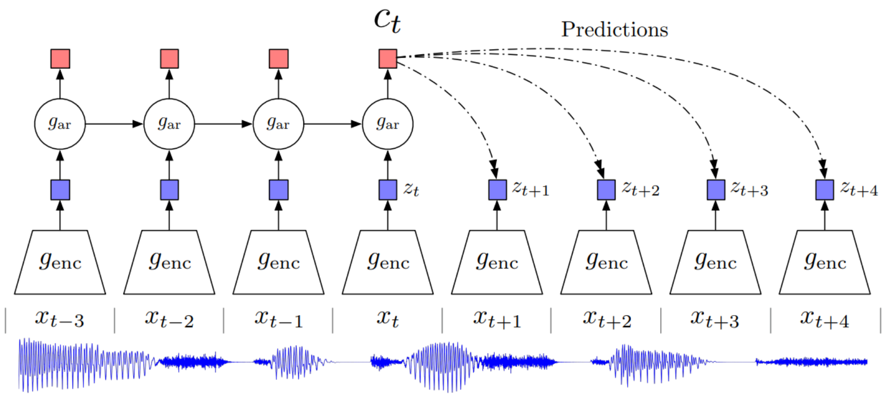


Deep InfoMax [Hjelm, 2019]:



X: input images
Y: encoded representations

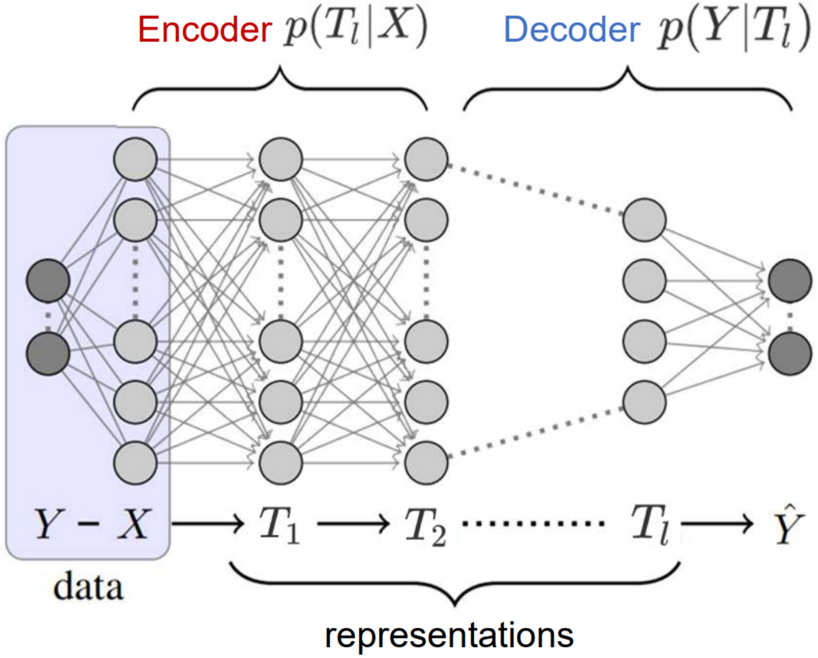
Contrastive Predictive Coding (CPC) [Van den Oord, 2018]:



InfoNCE loss:

$$\mathcal{L}_{\mathrm{N}} = -\mathop{\mathbb{E}}_{X}\left[\log\frac{f_k(x_{t+k},c_t)}{\sum_{x_j\in X} f_k(x_j,c_t)}\right]$$

Representation Information bottleneck (R-IB)



$I(T;X)$ Input information

$I(T;Y)$ Label information

A Markov chain description of DNN: $Y \leftrightarrow X \leftrightarrow T$

Minimal sufficient statistic

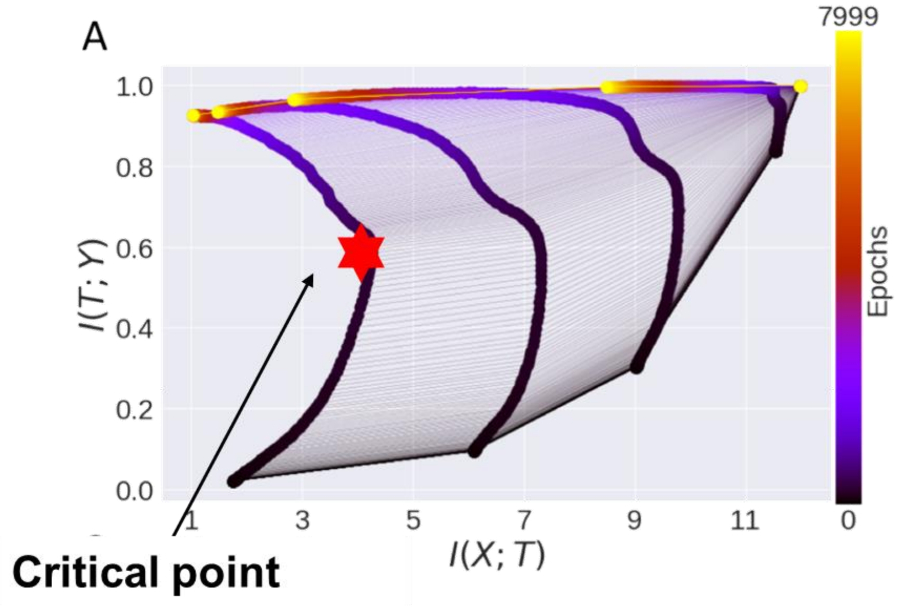
$$\min_{T} I(T;X), \text{ subject to } I(T;Y) = I(X;Y)$$

Information bottleneck (IB)

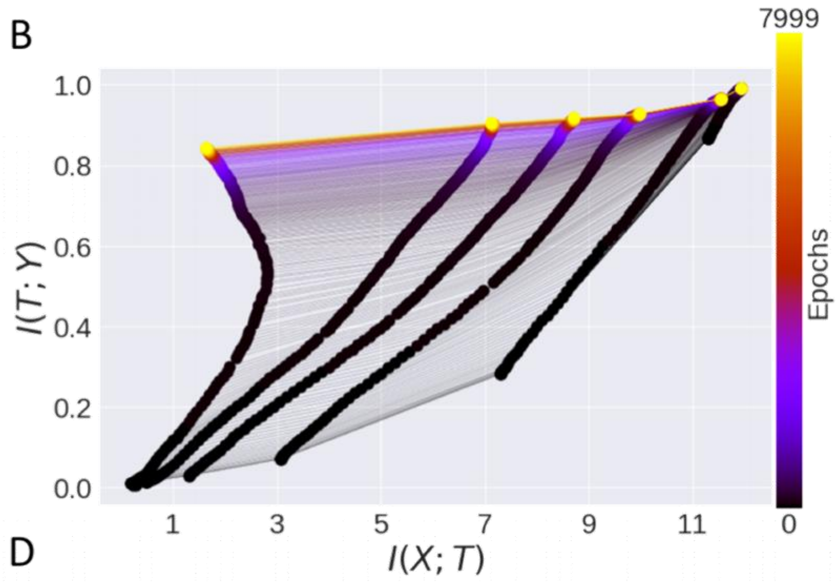
Lagrange relaxation
[Tishby, 1999]

$$\min_{p(t|x)} I(T;X) - \beta((I;Y) - I(X;Y))$$

Caveats of R Information bottleneck



Two-phase transition of **tanh** NNs trained by SGD (Shwartz-ziv, 2017)



Two-phase transition of **ReLU** NNs trained by SGD (Saxe, 2019)

Objective function:

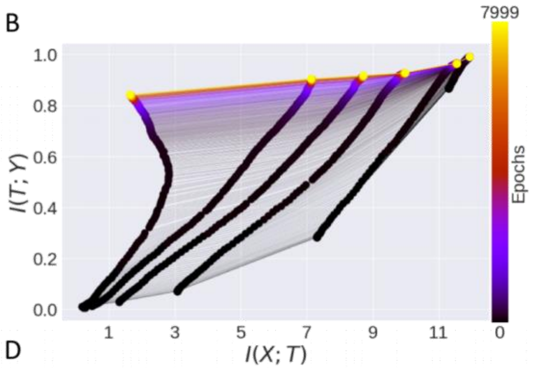
$$L_{xent}(x, y, w) = -\sum_{y \in \mathcal{Y}} p(y|x) \log p(y|x, w)$$

$$\min_{p(t|x)} I(T;X) - \beta(I;Y)$$

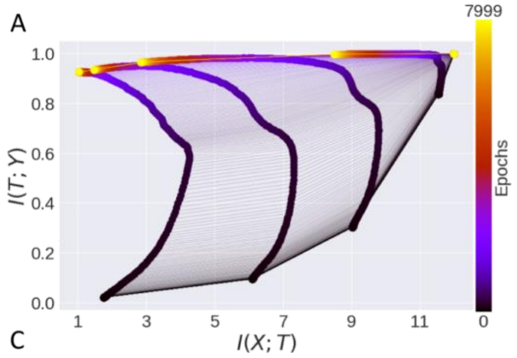
Minimality term

Sufficiency term

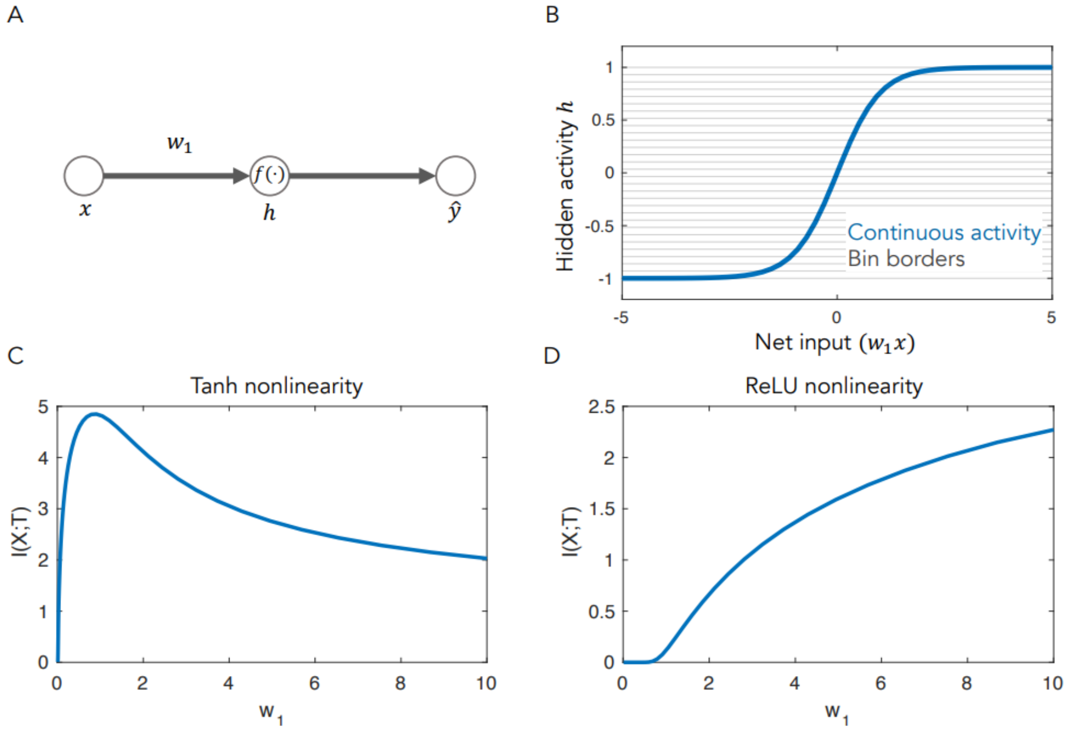
Caveats of R Information bottleneck



With ReLU non-linearities (Saxe, 2019)



With Tanh non-linearities (Shwartz-ziv, 2017)



The mutual information term $I(X;T)$ is *amortized*, i.e., it is directly influenced by inputs, so different activation functions will yield different distributions of representation T.

Question: Is conciseness of representations $I(X;T)$ necessarily connected to generalization of DNN?

On generalization error of neural networks

- Conventional generalization error definition

Empirical risk $L_S(w) \triangleq \frac{1}{n}\sum_{i=1}^{n} \ell(w, Z_i),$

True risk $L_{\text{test}}(w) \triangleq \mathbb{E}_{p(z)}[\ell(w, Z)] = \int p(z)\ell(w, z)dz.$

→

Generalization risk

$$\Delta L \triangleq L_{\text{test}}(w) - L_S(w).$$

- An effective generalization measure should consider the dataset (Nakkiran 2019).

Sampled dataset

$$S = (Z_1, Z_2, \ldots, Z_n) \sim p(Z)^{\otimes n}$$

Stochastic algorithm

$$\mathcal{A} : p(w|S)$$

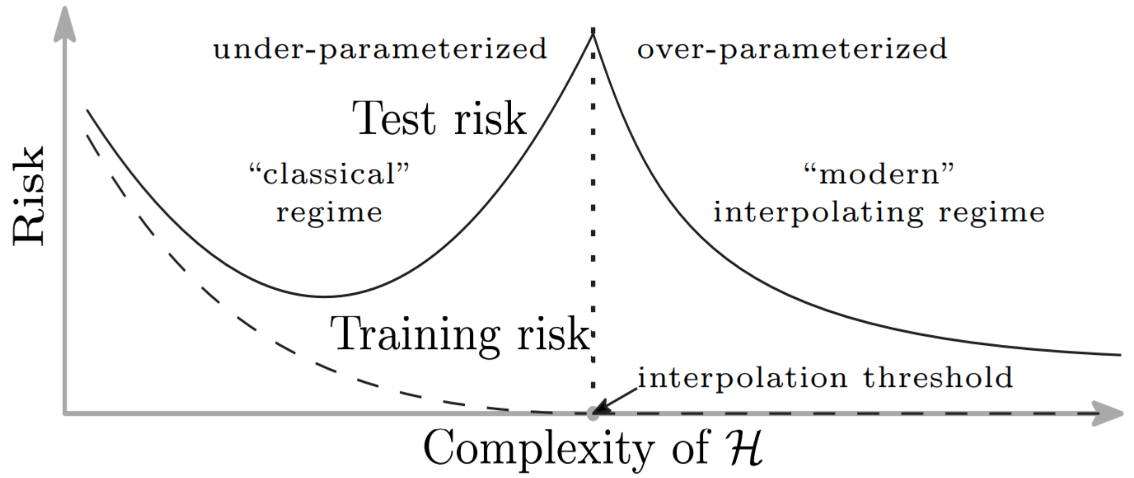
→

Data-dependent generalization risk

$$\mathbb{E}_{p(w,S)}[\Delta L] = \mathbb{E}_{p(w,S)}[L_{\text{test}}(w) - L_S(w)].$$

On generalization error of neural networks

Double descent (Belkin, 2018)



Linear PAC-Bayes bound (McAllester, 2013)

Empirical risk

Generalization risk

$$L_{\text{test}} \leq \frac{1}{1-\frac{1}{2\beta}}\left(L_S(w) + \beta \mathrm{KL}(p(w|S) \parallel p(w))\right)$$

Data-dependent PAC-Bayes bound (Dziugaite, 2020)

$$\mathbb{E}_{p(w,S)}[L_{\text{test}} - L_S(w)] \leq \gamma \inf_{p(w)} \mathbb{E}_{p(S)}[\mathrm{KL}(p(w|S) \parallel p(w))]$$

"Oracle prior" $p^*(w) = \mathbb{E}_S[p(w|S)]$

$$\mathbb{E}_S[\mathrm{KL}(p(w|S) \parallel p^*(w))] = I(W;S)$$

IIW: Information in weights

Information complexity of learning algorithms (Xu, 2017)

$$\mathbb{E}_{p(w,S)}[\Delta L] \leq \sqrt{\frac{2\sigma^2}{n} I(S;W)},$$

Data-dependent PAC-Bayes bound (Dziugaite, 2020)

$$\mathbb{E}_{p(w,S)}[L_{\text{test}} - L_S(w)] \leq \gamma \inf_{p(w)} \mathbb{E}_{p(S)}[\text{KL}(p(w|S) \parallel p(w))]$$

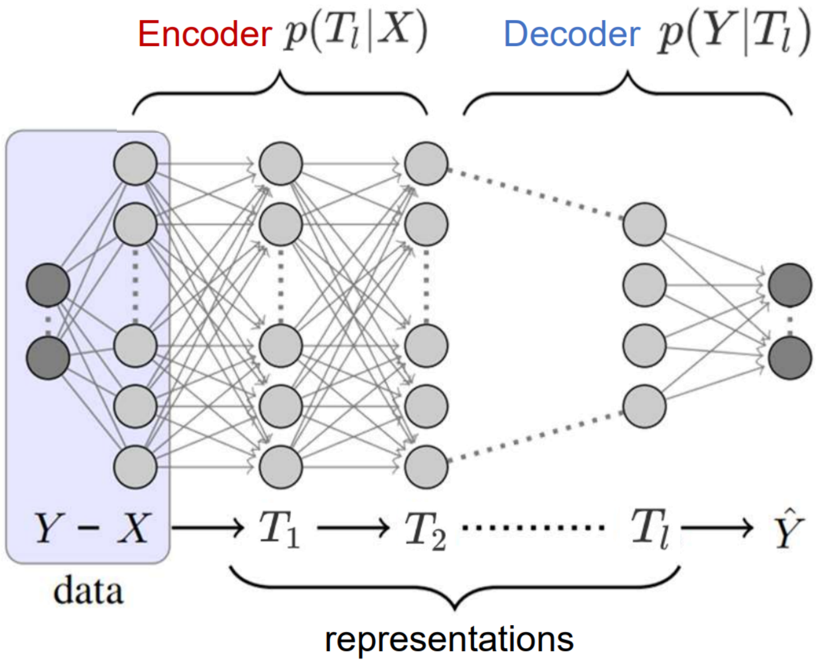
"Oracle prior" $p^*(w) = \mathbb{E}_S[p(w|S)]$

$$\mathbb{E}_S[\text{KL}(p(w|S) \parallel p^*(w))] = I(W;S)$$

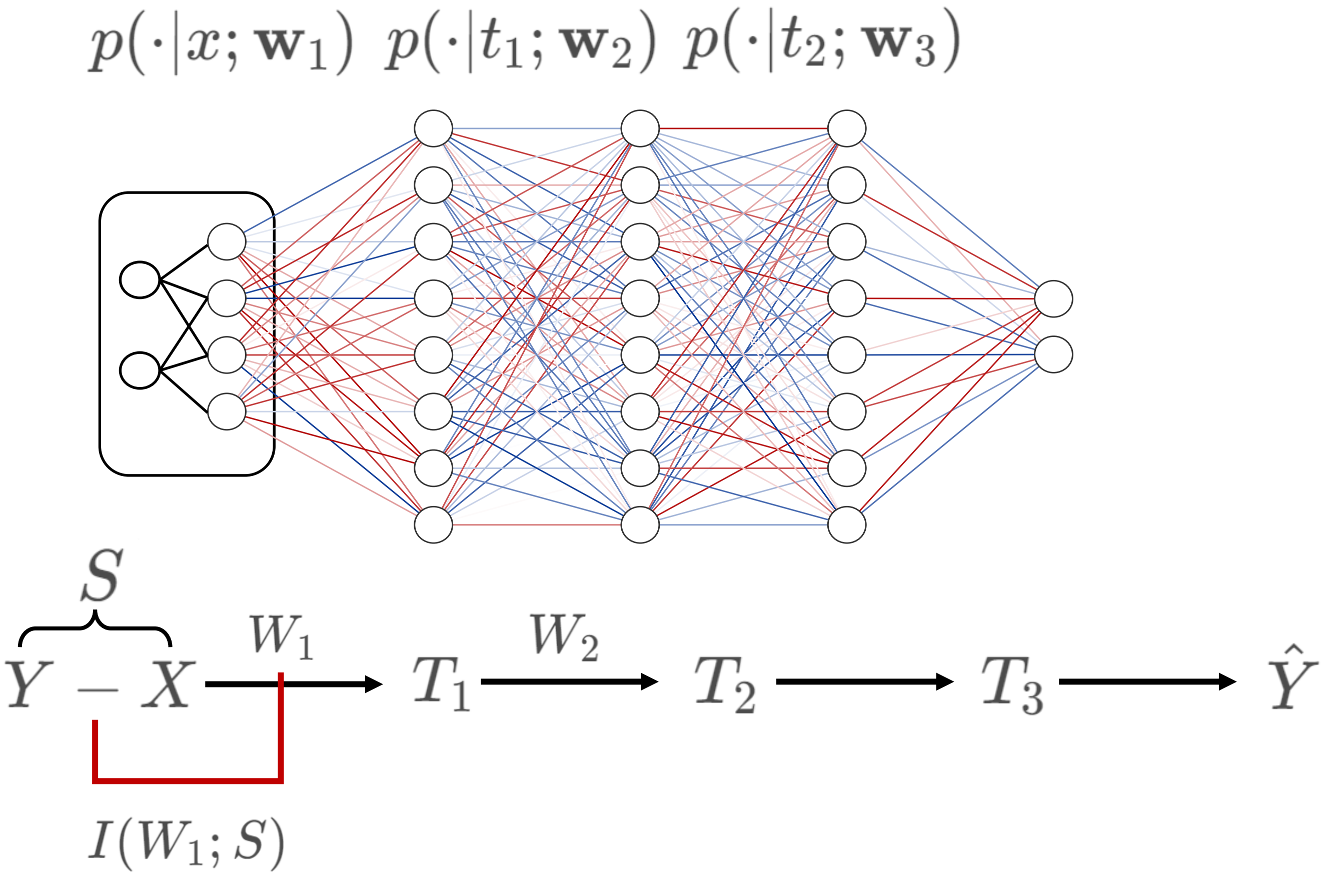
Findings:

- The oracle prior that achieves the sharpest PAC-Bayes bound is aligned with the information-theoretic algorithm complexity!
- Both are based on information stored in weights (IIW)

Information in weights or Information in representation?



V.S.



PIB: PAC-Bayes information bottleneck

maximize $I(W_1; Y|X, S)$

$$\overbrace{Y - X}^{S} \xrightarrow{W_1} T_1 \xrightarrow{W_2} T_2 \longrightarrow T_3 \longrightarrow \hat{Y}$$

minimize $I(W_1; S)$

Sufficiency term

Minimality term

$$\max_{p(\mathbf{w}|S)} I(\mathbf{w}; Y|X, S) - \beta I(\mathbf{w}; S),$$

$$\min_{p(\mathbf{w}|S)} \mathcal{L}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S).$$

$$\min_{p(t|x)} I(T; X) - \beta(I; Y)$$

Minimality term

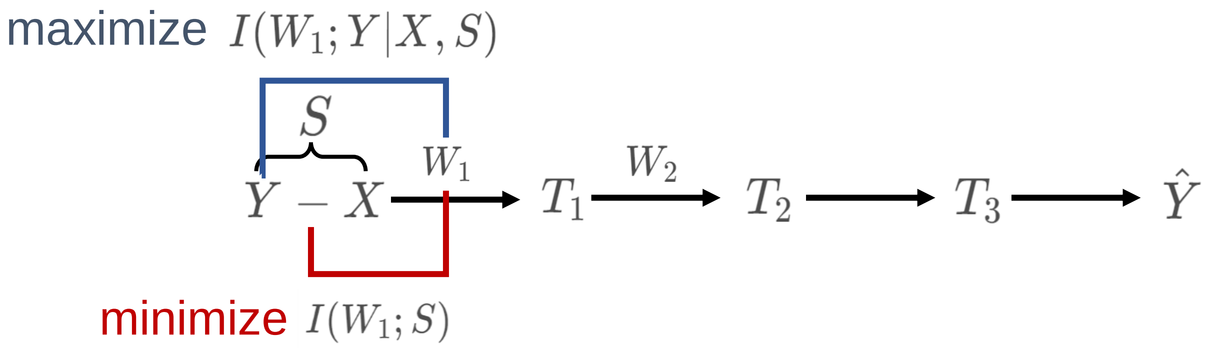
Sufficiency term

Approximate IIW

PIB objective:

$$\min_{p(\mathbf{w}|S)} \mathcal{L}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S).$$

$$I(\mathbf{w}; S) = \mathbb{E}_{p(S)}[\text{KL}(p(\mathbf{w}|S) \parallel p(\mathbf{w}))]$$



$$\text{KL}(p(\mathbf{w}|S) \parallel p(\mathbf{w})) = \frac{1}{2}\left[\log \frac{\det \Sigma_S}{\det \Sigma_0} - D + (\boldsymbol{\theta}_S - \boldsymbol{\theta}_0)^\top \Sigma_0^{-1}(\boldsymbol{\theta}_S - \boldsymbol{\theta}_0) + \text{tr}\left(\Sigma_0^{-1}\Sigma_S\right)\right]$$

Assume affinity $\Sigma_0 = A\Sigma_S$

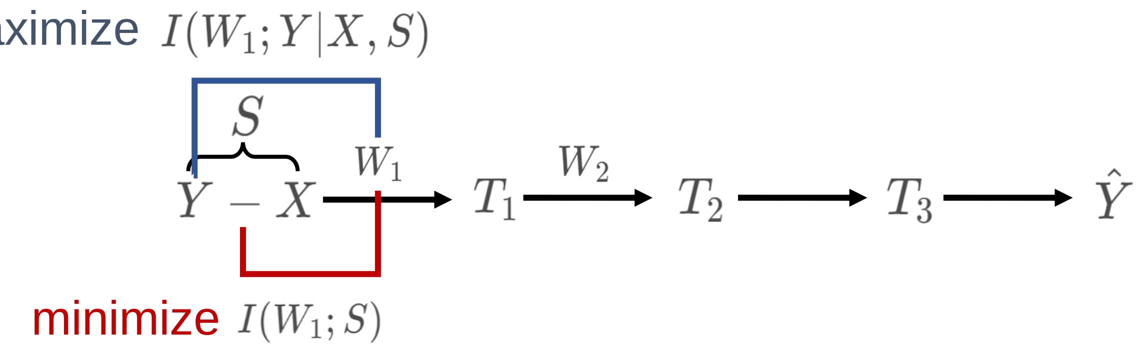
$$\text{KL}(p(w|S) \parallel p(w)) \propto (\theta_S - \theta_0)^\top \Sigma_0^{-1}(\theta_S - \theta_0)$$

Approximate IIW

PIB objective:

$$\min_{p(\mathbf{w}|S)} \mathcal{L}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S).$$

$$I(\mathbf{w}; S) = \mathbb{E}_{p(S)}[\text{KL}(p(\mathbf{w}|S) \parallel p(\mathbf{w}))]$$



$$\mathbb{E}_{p(S)}[\text{KL}(p(w|S) \parallel p(w))] \propto \mathbb{E}_{p(S)}[(\theta_S - \theta_0)^\top \Sigma_0^{-1} (\theta_S - \theta_0)]$$

"Oracle prior"

$$p^*(w) = \mathbb{E}_S[p(w|S)]$$

Q. How do we generate many samples following $p(S)$ without truly sampling from $p(z)^{\otimes n}$?

A. Bootstrapping for prior covariance:

$$\Sigma_0 = \mathbb{E}_{p(S)}\left[(\boldsymbol{\theta}_S - \boldsymbol{\theta}_0)(\boldsymbol{\theta}_S - \boldsymbol{\theta}_0)^\top\right] \simeq \frac{1}{K} \sum_k (\boldsymbol{\theta}_{S_k} - \boldsymbol{\theta}_S)(\boldsymbol{\theta}_{S_k} - \boldsymbol{\theta}_S)^\top, \quad S_k \sim p(S)$$

Approximate IIW

Lemma 2 (Approximation of Oracle Prior Covariance). *Given the definition of influence functions (Lemma 1) and Poisson bootstrapping (Lemma A.2), the covariance matrix of the oracle prior can be approximated by*

$$\Sigma_0 = \mathbb{E}_{p(S)}\left[(\boldsymbol{\theta}_S - \boldsymbol{\theta}_0)(\boldsymbol{\theta}_S - \boldsymbol{\theta}_0)^\top\right] \simeq \frac{1}{K}\sum_{k=1}^{K}\left(\hat{\boldsymbol{\theta}}_{\boldsymbol{\xi}^k} - \hat{\boldsymbol{\theta}}\right)\left(\hat{\boldsymbol{\theta}}_{\boldsymbol{\xi}^k} - \hat{\boldsymbol{\theta}}\right)^\top \simeq \frac{1}{n}\mathbf{H}_{\hat{\boldsymbol{\theta}}}^{-1}\mathbf{F}_{\hat{\boldsymbol{\theta}}}\mathbf{H}_{\hat{\boldsymbol{\theta}}}^{-1} \simeq \frac{1}{n}\mathbf{F}_{\hat{\boldsymbol{\theta}}}^{-1}, \tag{13}$$

where $\mathbf{F}_{\hat{\boldsymbol{\theta}}}$ is Fisher information matrix (FIM); we omit the subscript S of $\hat{\boldsymbol{\theta}}_S$ and $\hat{\boldsymbol{\theta}}_{S,\boldsymbol{\xi}}$ for notation conciseness, and $\boldsymbol{\xi}^k$ is the bootstrap resampling weight in the k-th experiment.

$$\hat{\boldsymbol{\theta}}_S \triangleq \operatorname*{argmin}_{\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n}\ell_i(\boldsymbol{\theta}),$$

$$\hat{\boldsymbol{\theta}}_{S,\boldsymbol{\xi}} \triangleq \operatorname*{argmin}_{\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n}\xi_i\ell_i(\boldsymbol{\theta}),$$

Influence function ψ

$$\hat{\boldsymbol{\theta}}_{S,\boldsymbol{\xi}} - \hat{\boldsymbol{\theta}}_S \simeq \frac{1}{n}\sum_{i=1}(\xi_i - 1)\psi_i = \frac{1}{n}\Psi^\top(\boldsymbol{\xi} - \mathbf{1}).$$

Estimate IIW in acceptable time

$$I(\mathbf{w};S) \propto n\mathbb{E}_{p(S)}\left[(\boldsymbol{\theta}_S-\boldsymbol{\theta}_0)^\top \mathbf{F}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{\theta}_S-\boldsymbol{\theta}_0)\right] \simeq n(\bar{\boldsymbol{\theta}}_S-\boldsymbol{\theta}_0)^\top \mathbf{F}_{\hat{\boldsymbol{\theta}}}(\bar{\boldsymbol{\theta}}_S-\boldsymbol{\theta}_0) = \tilde{I}(\mathbf{w};S)$$

$$\bar{\boldsymbol{\theta}}_S = \sqrt{\frac{1}{K}\sum_{k=1}^K \hat{\boldsymbol{\theta}}_k^2} = \left(\sqrt{\frac{1}{K}\sum_{k=1}^K \hat{\theta}_{1,k}^2},\dots,\sqrt{\frac{1}{K}\sum_{k=1}^K \hat{\theta}_{D,k}^2}\right)^\top$$

$$\Delta\boldsymbol{\theta} = \bar{\boldsymbol{\theta}}_S - \boldsymbol{\theta}_0 \in \mathbb{R}^D \qquad F_{\hat{\theta}} = \frac{1}{T}\sum_{t=1^T} \nabla_\theta \ell_t(\hat{\theta})\nabla_\theta \ell_t^\top(\hat{\theta}) \in \mathbb{R}^{D\times D}$$

$$\tilde{I}(\mathbf{w};S) = n\Delta\boldsymbol{\theta}^\top\left[\frac{1}{T}\sum_{t=1}^T \nabla_{\boldsymbol{\theta}}\ell_t(\hat{\boldsymbol{\theta}})\nabla_{\boldsymbol{\theta}}\ell_t^\top(\hat{\boldsymbol{\theta}})\right]\Delta\boldsymbol{\theta} = \frac{n}{T}\sum_{t=1}^T\Big[\underbrace{\Delta\boldsymbol{\theta}^\top\nabla_{\boldsymbol{\theta}}\ell_t(\hat{\boldsymbol{\theta}})}_{\text{Vector product}}\Big]^2,$$

Estimate IIW in acceptable time

Algorithm 1: Efficient approximate information estimation of $I(\mathbf{w}; S)$

Pretrain the model by vanilla SGD to obtain the prior mean $\boldsymbol{\theta}_0$;

for t=1:T_0 **do**

  $\nabla L_t \leftarrow \nabla_{\boldsymbol{\theta}} \frac{1}{B} \sum_b \ell_b(\hat{\boldsymbol{\theta}}_{t-1}),\ \hat{\boldsymbol{\theta}}_t \leftarrow \hat{\boldsymbol{\theta}}_{t-1} - \eta \nabla L_t$; `/* Vanilla SGD */`

  $\nabla \mathcal{L} \leftarrow \nabla \mathcal{L} \bigcup \{\nabla L_t\}$; `/* Store gradients */`

  $\bar{\boldsymbol{\theta}}_t \leftarrow \sqrt{\rho \bar{\boldsymbol{\theta}}_{t-1}^2 + \frac{1-\rho}{K} \sum_{k=0}^{K-1} \hat{\boldsymbol{\theta}}_{t-k}^2}$; `/* Moving average */`

end

$\Delta\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}_{T_0} - \boldsymbol{\theta}_0,\ \ \Delta\mathbf{F}_0 \leftarrow 0$;

for t=1:T_1 **do**

  $\Delta\mathbf{F}_t \leftarrow \Delta\mathbf{F}_{t-1} + (\Delta\boldsymbol{\theta}^\top \nabla L_t)^2$; `/* Storage-friendly computation */`

end

$\widetilde{I}(\mathbf{w}; S) \leftarrow \frac{n}{T_1} \Delta\mathbf{F}_{T_1}$;

Optimal posterior on PIB

PIB objective:

$$\min_{p(\mathbf{w}|S)} \mathcal{L}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S).$$

IIW approximation and oracle prior covariance

$$\tilde{I}(\mathbf{w}; S) = n\Delta\boldsymbol{\theta}^\top \left[\frac{1}{T}\sum_{t=1}^{T} \nabla_{\boldsymbol{\theta}}\ell_t(\hat{\boldsymbol{\theta}})\nabla_{\boldsymbol{\theta}}\ell_t^\top(\hat{\boldsymbol{\theta}})\right]\Delta\boldsymbol{\theta} = \frac{n}{T}\sum_{t=1}^{T}\left[\Delta\boldsymbol{\theta}^\top\nabla_{\boldsymbol{\theta}}\ell_t(\hat{\boldsymbol{\theta}})\right]^2,$$

$$\Sigma_0 \simeq \frac{1}{n}H^{-1}FH^{-1} \simeq \frac{1}{n}F^{-1}$$

Q. How do we train a model that optimizes on PIB directly?

$$\min_{p(\mathbf{w}|S)} \mathcal{L}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S), \quad \text{s.t.} \int p(\mathbf{w}|S)d\mathbf{w} = 1.$$

Build the Lagrangian

$$\min_{p(\mathbf{w}|S)} \tilde{\mathcal{L}}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S) + \int \alpha_S \int \left(p(\mathbf{w}|S) - 1\right) d\mathbf{w}dS$$

$$\nabla_{p(\mathbf{w}|S^*)}\tilde{\mathcal{L}}_{\text{PIB}} = 0$$

$$p(\mathbf{w}|S^*) = \frac{1}{Z(S)}p(\mathbf{w})\exp\left\{-\frac{1}{\beta}\hat{L}_{S^*}(\mathbf{w})\right\}$$

Optimal posterior on PIB

Lemma 3 (Optimal Posterior for PAC-Bayes Information Bottleneck). *Given an observed dataset S^*, the optimal posterior $p(\mathbf{w}|S^*)$ of PAC-Bayes IB in Eq. (5) should satisfy the following form that*

$$p(\mathbf{w}|S^*) = \frac{1}{Z(S)} p(\mathbf{w}) \exp\left\{-\frac{1}{\beta}\hat{L}_{S^*}(\mathbf{w})\right\} = \frac{1}{Z(S)} \exp\left\{-\frac{1}{\beta}U_{S^*}(\mathbf{w})\right\}, \tag{16}$$

where $U_{S^*}(\mathbf{w})$ is the energy function defined as $U_{S^*}(\mathbf{w}) = \hat{L}_{S^*}(\mathbf{w}) - \beta \log p(\mathbf{w})$, and $Z(S)$ is the normalizing constant.

Q. How do we design algorithm that enables us to sample from the optimal posterior?

A. Using stochastic gradient Langevin dynamics (SGLD) (Welling, 2011)

$$\ldots w_{t-1} \xrightarrow{\Delta w_t + \epsilon_t} w_t \xrightarrow{\Delta w_{t+1} + \epsilon_{t+1}} w_{t+1} \;\cdots$$

ϵ_t is a zero-mean, isotropic Gaussian noise.

Optimal posterior on PIB

Energy function $U(\mathbf{w})$

Gradient $g_k = \nabla U(w)$

SGLD update process

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta_k \mathbf{g}_k + \sqrt{2\eta_k \beta}\varepsilon_k,$$

w_k converge to $\pi(w)$

$$\pi(\mathbf{w}) \propto \exp(-\tfrac{1}{\beta}U(\mathbf{w}))$$

Optimal PIB posterior

$$p(\mathbf{w}|S^*) = \frac{1}{Z(S)} p(\mathbf{w}) \exp\left\{-\frac{1}{\beta}\hat{L}_{S^*}(\mathbf{w})\right\} = \frac{1}{Z(S)} \exp\left\{-\frac{1}{\beta}U_{S^*}(\mathbf{w})\right\},$$

PIB energy function

$$U_{S^*}(\mathbf{w}) = \hat{L}_{S^*}(\mathbf{w}) - \beta \log p(\mathbf{w})$$

Optimal posterior on PIB

Algorithm 2: Optimal Gibbs posterior inference by SGLD.

Data: Total number of samples n, batch size B, learning rate η, temperature β

Result: A sequence of weights $\{\mathbf{w}_t\}_{t \geq \hat{k}}$ following $p(\mathbf{w}|S^*)$

repeat

/* Mini-batch gradient of energy function */

Compute energy function gradient

$\nabla \widetilde{U}_{S^*}(\mathbf{w}_{t-1}) \leftarrow \nabla \left(\boxed{-\frac{B}{n} \sum_b \log p(Y_b|X_b, \mathbf{w}_{t-1}) - \beta_{t-1} \log p(\mathbf{w}_{t-1})} \right)$;

/* SGLD by gradient plus isotropic Gaussian noise */

$\boxed{\varepsilon_t \leftarrow \mathcal{N}(\varepsilon|\mathbf{0}, \mathbf{I}_D), \mathbf{w}_t \leftarrow \mathbf{w}_{t-1} - \eta_{t-1} \nabla \widetilde{U}_{S^*}(\mathbf{w}_{t-1}) + \sqrt{2\eta_{t-1}\beta_{t-1}}\varepsilon_t}$;

/* Learning rate & temperature decay */

Energy function gradient descent w/ noise

$\eta_t \leftarrow \phi_\eta(\eta_{t-1}), \beta_t \leftarrow \phi_\beta(\beta_{t-1}), t \leftarrow t + 1$;

until *the weight sequence* $\{\mathbf{w}_t\}_{t \geq \hat{k}}$ *becomes stable*;

What we have done

- Build information bottleneck on information in weights (IIW)

$$\min_{p(\mathbf{w}|S)} \mathcal{L}_{\text{PIB}} = L_S(\mathbf{w}) + \beta I(\mathbf{w}; S).$$

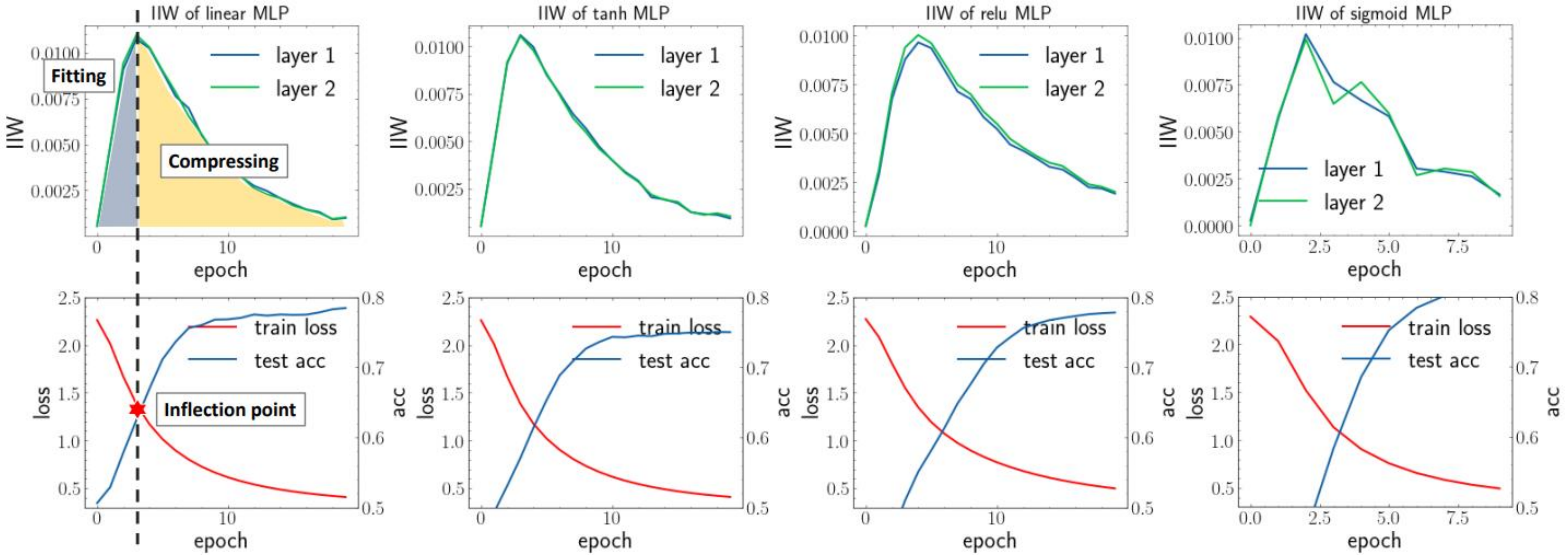
- Propose an algorithm for estimating IIW

$$\widetilde{I}(\mathbf{w}; S) = n\Delta\boldsymbol{\theta}^\top \left[\frac{1}{T} \sum_{t=1}^{T} \nabla_{\boldsymbol{\theta}} \ell_t(\hat{\boldsymbol{\theta}}) \nabla_{\boldsymbol{\theta}} \ell_t^\top(\hat{\boldsymbol{\theta}}) \right] \Delta\boldsymbol{\theta} = \frac{n}{T} \sum_{t=1}^{T} \left[\Delta\boldsymbol{\theta}^\top \nabla_{\boldsymbol{\theta}} \ell_t(\hat{\boldsymbol{\theta}}) \right]^2,$$

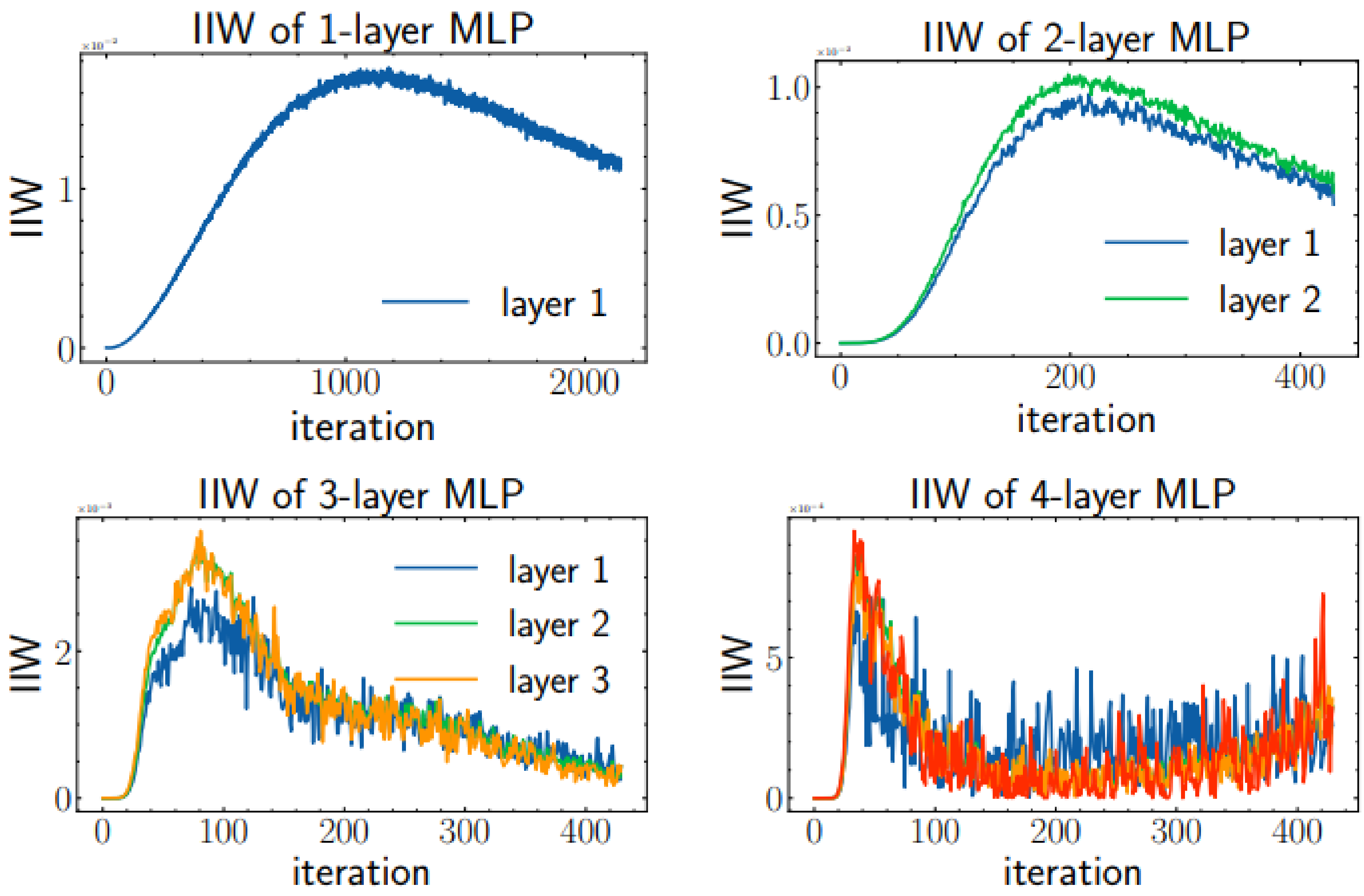
- Propose an algorithm to sample optimal posterior of PIB

$$\varepsilon_t \leftarrow \mathcal{N}(\varepsilon | \mathbf{0}, \mathbf{I}_D),\ \mathbf{w}_t \leftarrow \mathbf{w}_{t-1} - \eta_{t-1} \nabla \widetilde{U}_{S^*}(\mathbf{w}_{t-1}) + \sqrt{2\eta_{t-1}\beta_{t-1}}\varepsilon_t\ ;$$

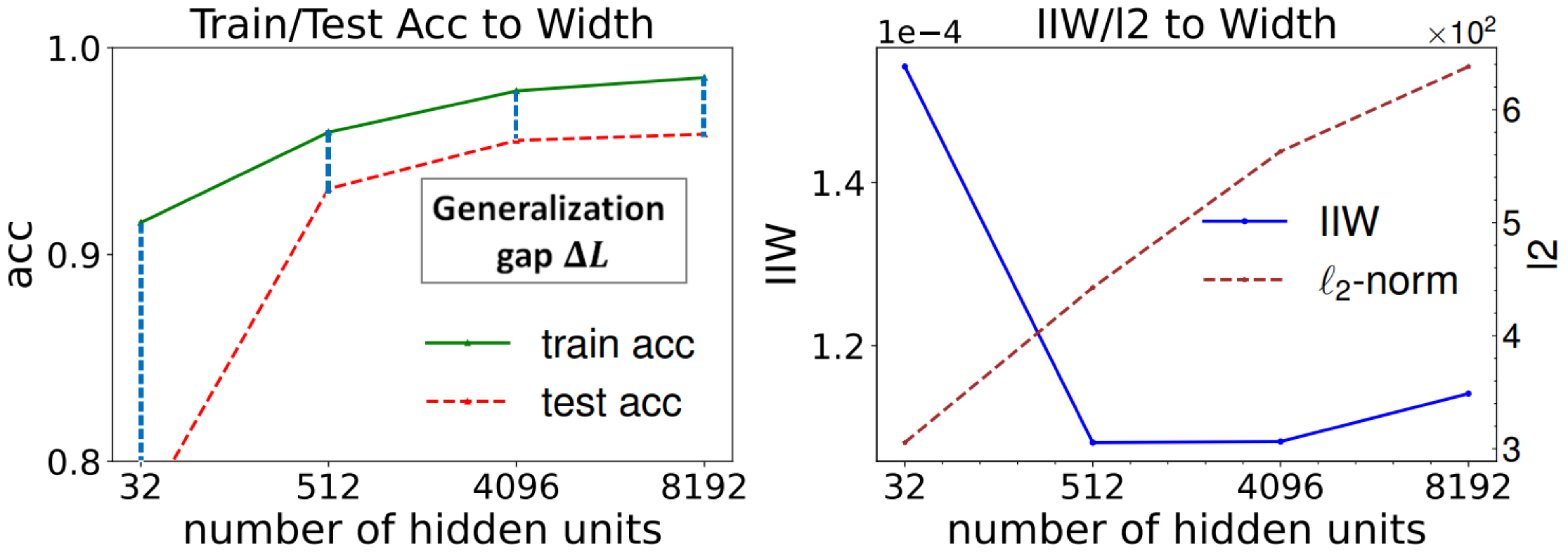
IIW w.r.t. activation functions



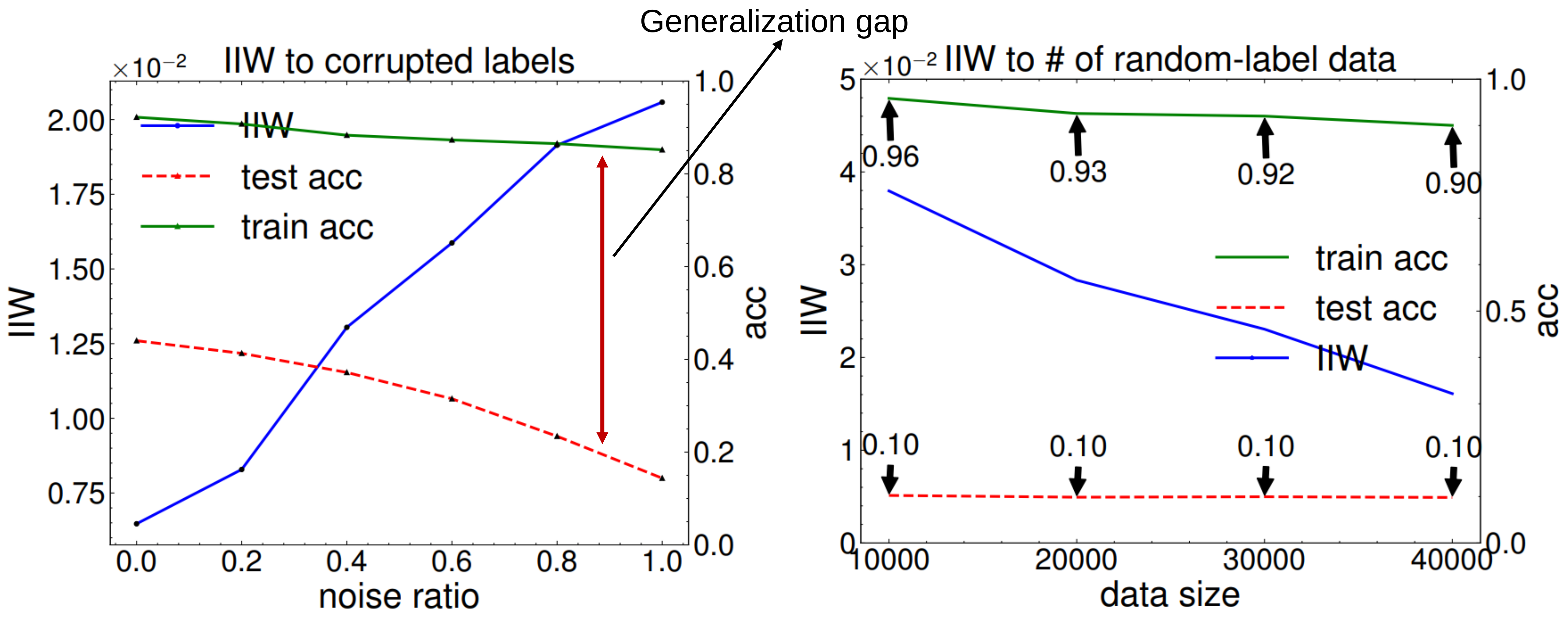
IIW w.r.t. number of layers



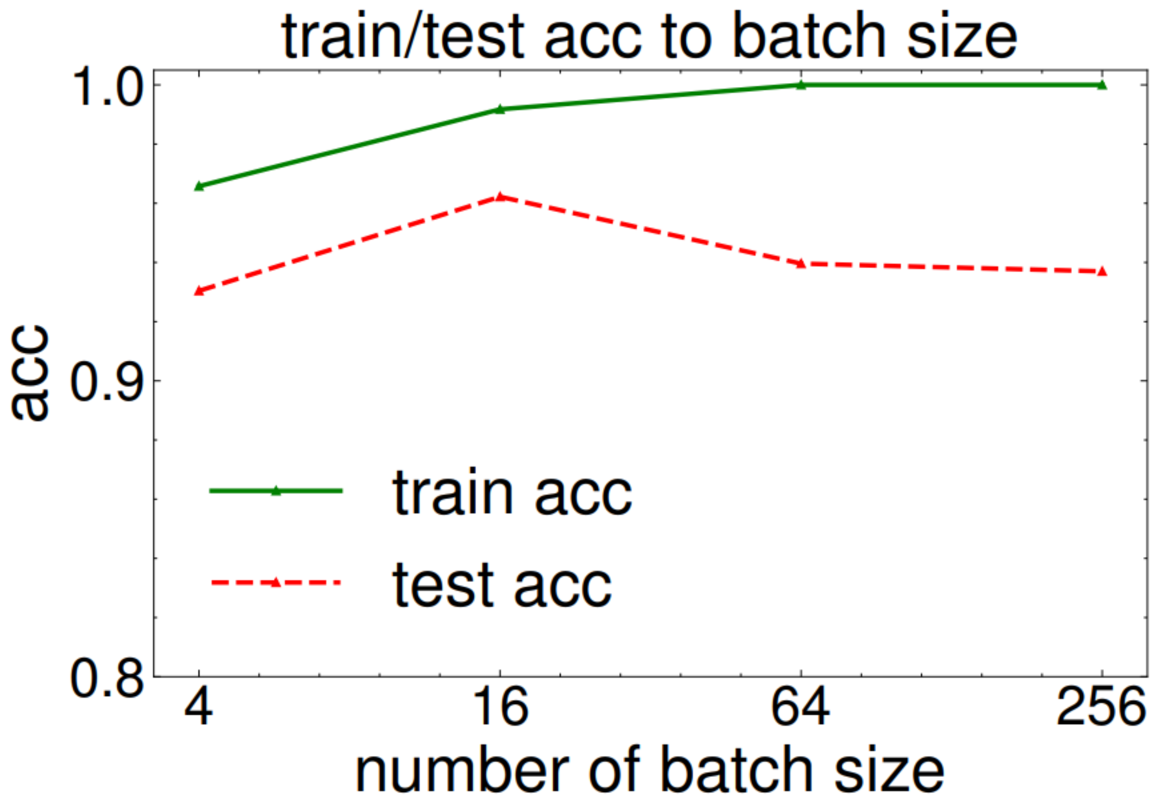
IIW w.r.t. width

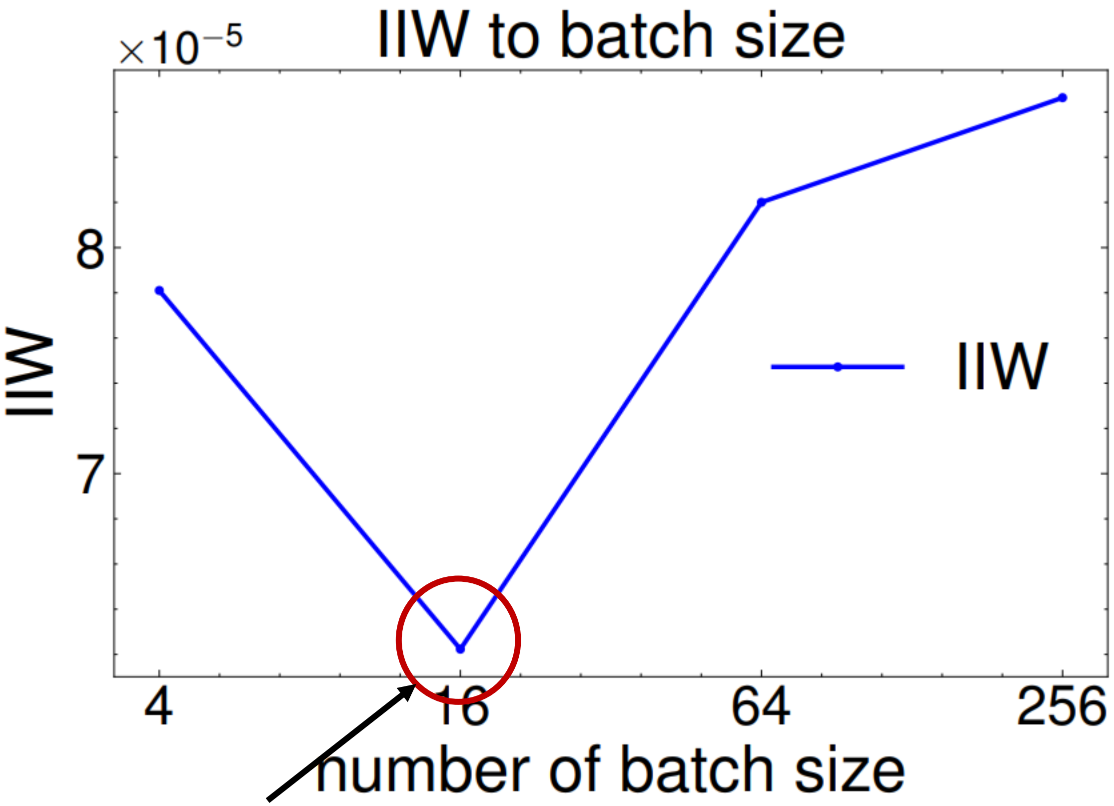


IIW w.r.t. noise ratio & sample size



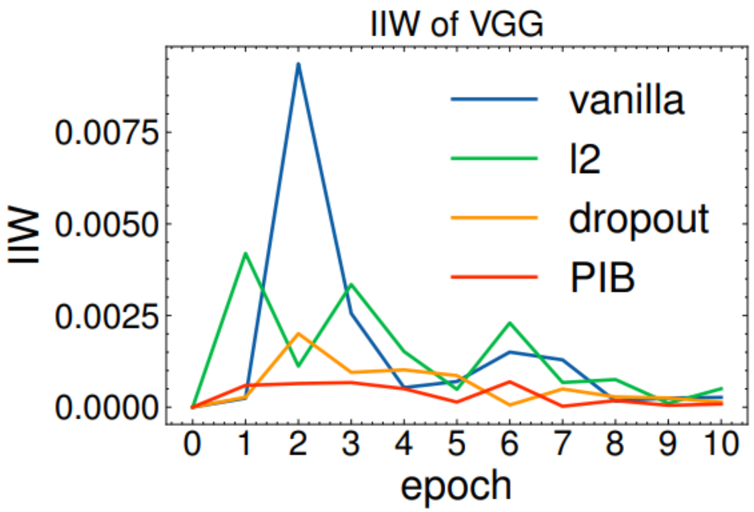
IIW w.r.t. batch size





We get to know this is the optimal even when we do not have the test set!

IIW in deep nets: VGGNet



Test ACC (%)	CIFAR10	CIFAR100	STL10	SVHN
vanilla SGD	77.03(0.57)	52.07(0.44)	54.31(0.65)	93.57(0.67)
SGD+ℓ_2-norm	77.13(0.53)	50.84(0.71)	55.30(0.68)	93.60(0.68)
SGD+dropout	78.95(0.60)	52.34(0.66)	56.35(0.78)	93.61(0.76)
SGD+PIB	**80.19(0.42)**	**56.47(0.62)**	**58.83(0.75)**	**93.88(0.88)**

Takeaway

- Measure **representation** v.s. **weight** information: weight is more essential
- PAC-Bayes information bottleneck: identify memorize-forget phase
- Information in weights: specify NNs generalization in broad cases
- PIB: introduce a new training strategy explicitly regularizing IIW

References

1. Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., & Bengio, Y. (2018, September). Learning deep representations by mutual information estimation and maximization. In ICLR.
2. Van den Oord, A., Li, Y., & Vinyals, O. (2018). Representation learning with contrastive predictive coding. arXiv e-prints, arXiv-1807.
3. Tishby, N., Pereira, F. C., & Bialek, W. (2000). The information bottleneck method. arXiv preprint physics/0004057.
4. Tishby N, Zaslavsky N. Deep learning and the information bottleneck principle[C]// IEEE Information Theory Workshop (ITW). IEEE, 2015: 1-5.
5. Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., & Sutskever, I. (2019). Deep double descent: Where bigger models and more data hurt. *arXiv preprint arXiv:1912.02292*.
6. Saxe A M, Bansal Y, Dapello J, et al. On the information bottleneck theory of deep learning[J]. Journal of Statistical Mechanics: Theory and Experiment, 2019, 2019(12): 124020.
7. Shwartz-Ziv R, Tishby N. Opening the black box of deep neural networks via information[J]. arXiv preprint arXiv:1703.00810, 2017.
8. M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine learning and the bias-variance trade-off. arXiv preprint arXiv:1812.11118, 2018.
9. Aolin Xu and Maxim Raginsky. Information-theoretic analysis of generalization capability of learning algorithms. In Advances in Neural Information Processing Systems, pp. 2521–2530, 2017.
10. Gintare Karolina Dziugaite, Kyle Hsu, Waseem Gharbieh, Gabriel Arpino, and Daniel Roy. On the role of data in PAC-Bayes. In International Conference on Artificial Intelligence and Statistics, pp. 604–612. PMLR, 2021.
11. Max Welling and Yee W Teh. Bayesian learning via stochastic gradient Langevin dynamics. In International Conference on Machine Learning, pp. 681–688, 2011.

PAC-Bayes Information Bottleneck

Zifeng Wang[1,2], Shao-Lun Huang[1], Ercan E Kuruoglu[1], Jimeng Sun[2], Xi Chen[3], Yefeng Zheng[3]

[1]UIUC, [2]Tsinghua University, [3]Tencent

ICLR 2022