MVIP2022

Shahid Chamran University of Ahvaz

Recent Advances in Understanding and Interpreting the DNNs

Erfan Loweimi* and Samira Loveymi†

* Research Associate, King's College London (KCL)
† Adjunct Lecturer, Shahid Chamran University of Ahvaz

Motivation ...

- Why is understanding DNNs important?

Motivation …

- Why is understanding DNNs important?
  - Reliable validation → Safer practice
    - E.g., self-driving car ... no margin for error
  - Extract new insights → Better practice
    - E.g., more efficient training … with less data

Outlines

- Information Bottleneck
- Over-parameterisation and Generalisation
- Interpretation/Visualisation of Filters/Activations

Outlines

- Information Bottleneck
- Over-parameterisation and Generalisation
- Interpretation/Visualisation of Filters/Activations

Why do DNNs generalise well?

MVIP2022

Shahid Chamran University of Ahvaz

Outlines (Part I)

- Information Bottleneck
- Over-parameterisation and Generalisation
- Interpretation/Visualisation of Filters/Activations

Information – Definition

- Information ≡ Average Surprise
- Information ... ≥ 0, $\propto$ *1/P*, additive for independent RV*s

* RV: random variable

Information – Definition

- Information ≡ Average Surprise
- Information ... ≥ 0, ∝ *1/P*, additive for independent RV*s

$$H(X) = \mathbb{E}\left[\log \frac{1}{P(x)}\right] = \sum_{x \in \mathcal{X}} P(x) \log \frac{1}{P(x)}$$

* RV: random variable

Information – Definition

- Information ≡ Average Surprise
- Information ... ≥ 0, ∝ *1/P*, additive for independent RV*s
- Quantitatively measured by **Entropy**

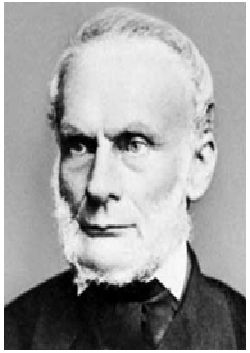
$$H(X) = \mathbb{E}\left[\log \frac{1}{P(x)}\right] = \sum_{x \in \mathcal{X}} P(x) \log \frac{1}{P(x)}$$

Entropy

* RV: random variable

Entropy over Time

R. Clausius



$$dS = \frac{dQ}{T}$$

1865

L. Boltzmann

$$S = k_B \log W$$

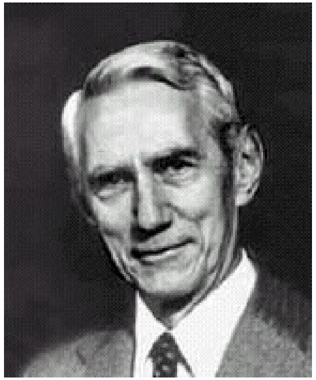
1870

J. Gibbs

$$S = -k_B \sum_i p_i \log p_i$$

1876

C. Shannon

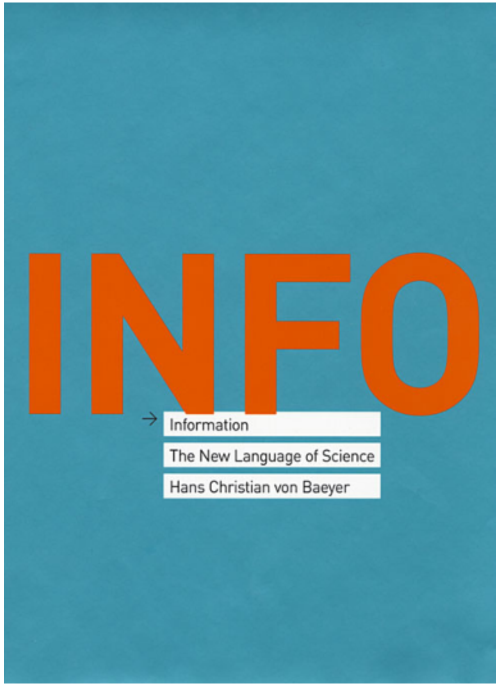


$$H = -\sum_i p_i \log_2 p_i$$

1948

MVIP2022

Shahid Chamran University of Ahvaz

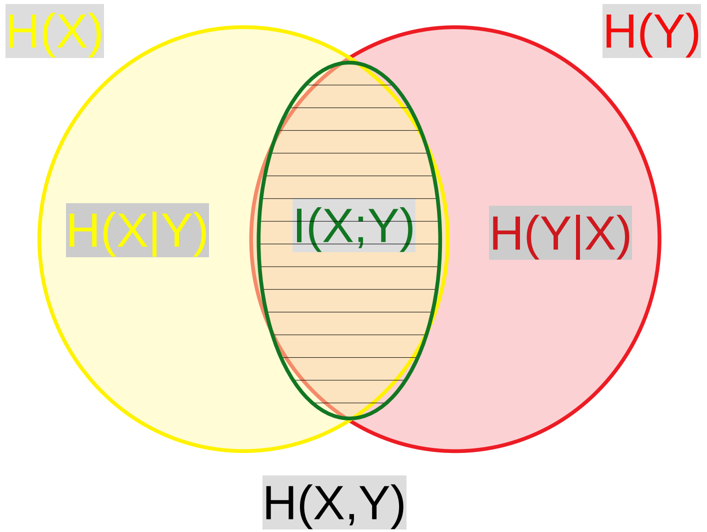


Claude Shannon, the founder of information theory, invented a way to measure 'the amount of information' in a message without defining the word 'information' itself, nor even addressing the question of the meaning of the message.

Information, The New Language of Science, Ch. 4, p. 28

Mutual Information (MI) … Idea

- A measure for Information X gives about Y (or vice verse)

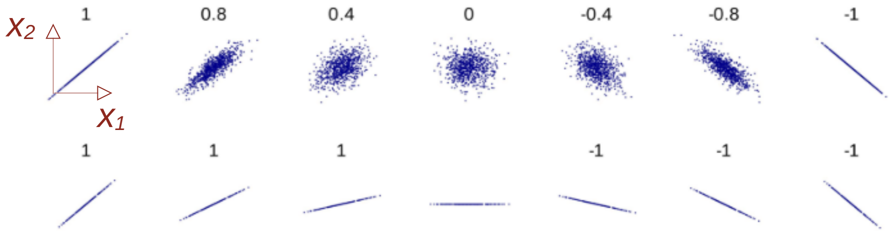


* I(X;Y): Mutual Information
* H(X): Entropy
* H(X|Y): Conditional entropy
* H(X,Y): Joint entropy

Mutual Information (MI) … Idea

- Think of cross-correlation …

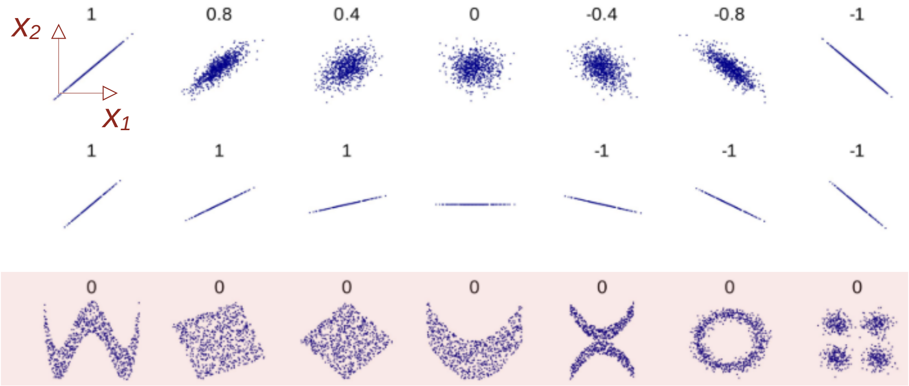
Cross-correlation (CC)



Mutual Information (MI) … Idea

- Think of cross-correlation … but *non-linear*

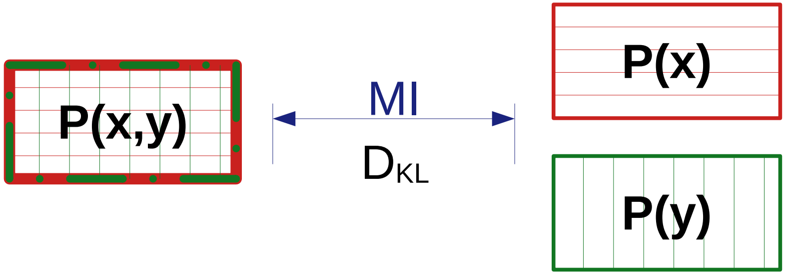
Cross-correlation (CC)



CC = 0 … but … MI != 0

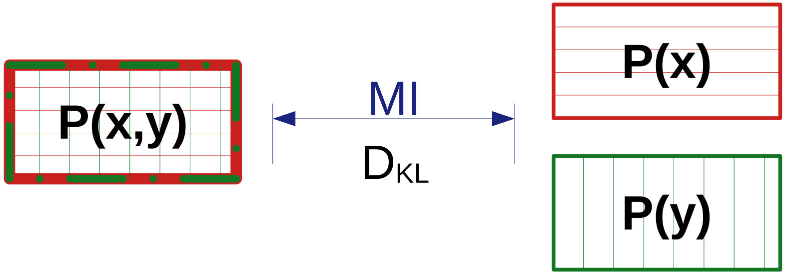
MI ... Definition

$$I(X;Y) = D_{KL}(P(x,y)||P(x)P(y))$$



MI ... Definition

$$I(X;Y) = D_{KL}(P(x,y)||P(x)P(y))$$



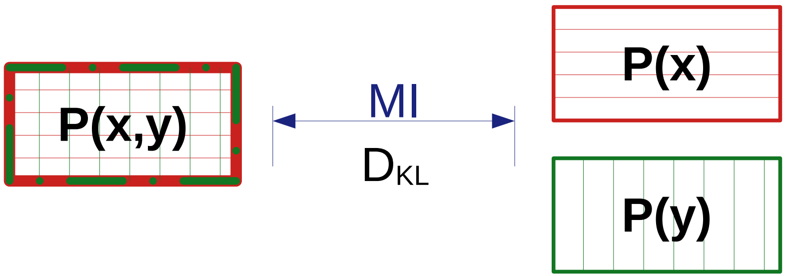
$$D_{KL}(P||Q) = -\sum_{x\in X} P(x)\ \log\frac{Q(x)}{P(x)} = H(P,Q) - H(P)$$

Cross-entropy Entropy

* D_{KL} : Kullback-Leibler Divergence

MI … Definition

$$I(X;Y) = D_{KL}(P(x,y)||P(x)P(y))$$



If X ⊥ Y => I(X,Y) = 0

$$D_{KL}(P||Q) = -\sum_{x \in X} P(x)\ \log \frac{Q(x)}{P(x)} = H(P,Q) - H(P)$$

Cross-entropy Entropy

* D_{KL} : Kullback-Leibler Divergence

MI … Properties

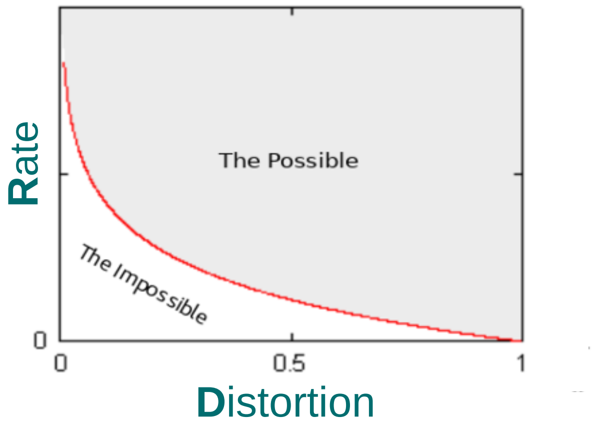
- **Data Processing Inequality (DPI)**
  - *… Post-processing cannot increase information …*
  - Markov Chain: $X \rightarrow T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow \ldots$
    - $I(X;T_1) \geq I(X;T_2)$; $I(T_1;T_2) \geq I(X;T_2)$
- **Transformation Invariance**
  - $I(X;Y) = I(f(X); g(Y))$ where f & g are *invertible* functions

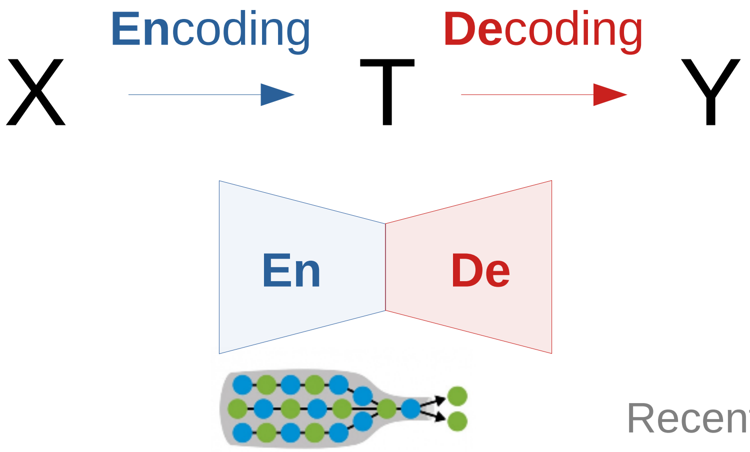
MVIP2022

Shahid Chamran University of Ahvaz

Rate-Distortion Theory

- Encode X by T ...
  - Obj. Minimal Rate
  - s.t. Distortion ≤ D_{max}





X: Observation
Y: Variable of interest
T: Representation of X

Information Bottleneck (IB)

- Turn finding T to a learning problem using MI ...

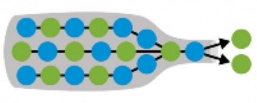
Compression/
Minimality/Complexity

Fidelity/
Sufficiency/Accuracy

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

Encoding   Decoding

X → T → Y



Information Bottleneck (IB)

- Turn finding T to a learning problem using MI ...

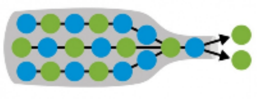
Compression/
Minimality/Complexity

Fidelity/
Sufficiency/Accuracy

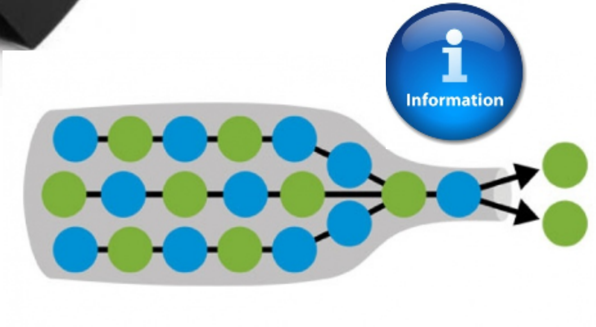
$$\min_{q(t|x)} \{I(T;X) \; - \; \beta I(T;Y)\}$$

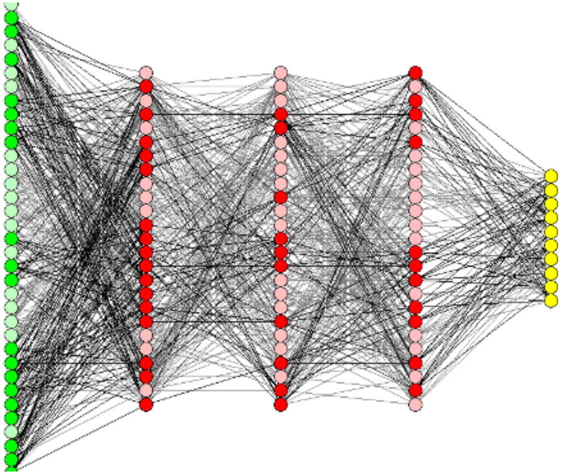
IDEALLY … in coding ...
- $I(T;X)$ ↔ as LOW as possible (min Rate)
- $I(T;Y)$ ↔ as HIGH as possible (min Distortion)

X —Encoding→ T —Decoding→ Y



Opening the Black Box of DNNs via Information Bottleneck



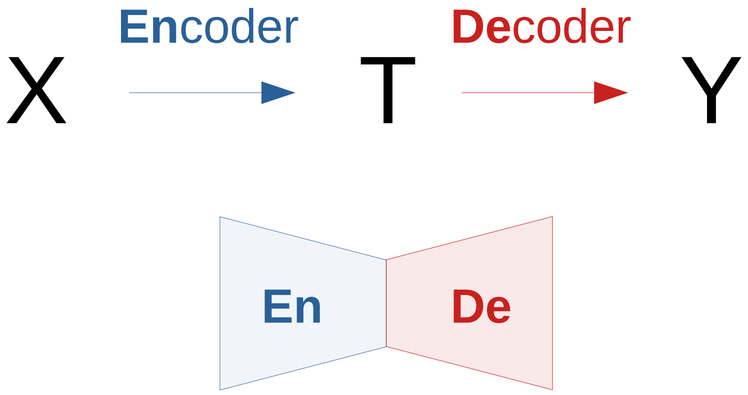


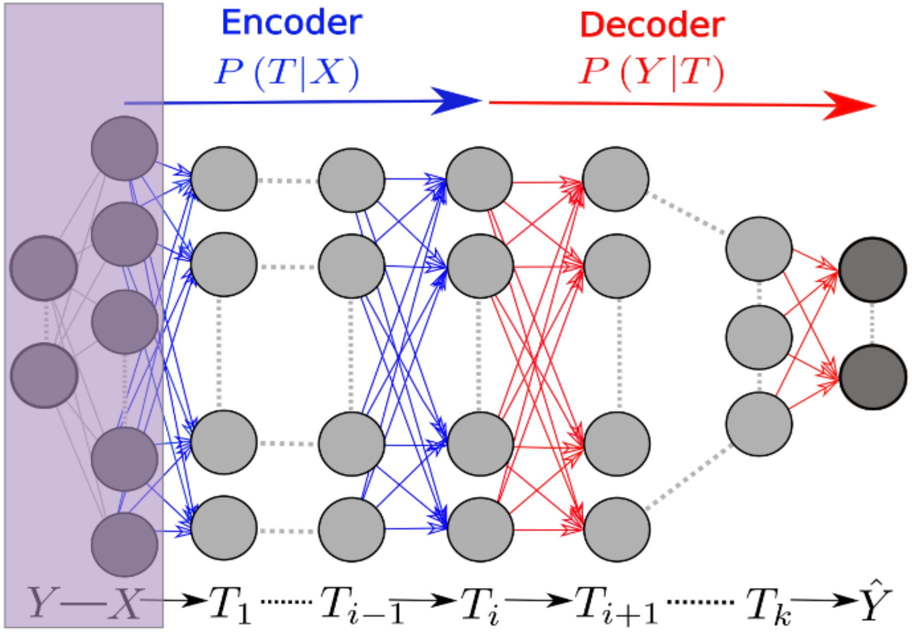
Recent advances ...

MVIP2022

Shahid Chamran University of Ahvaz

Opening the black box ...

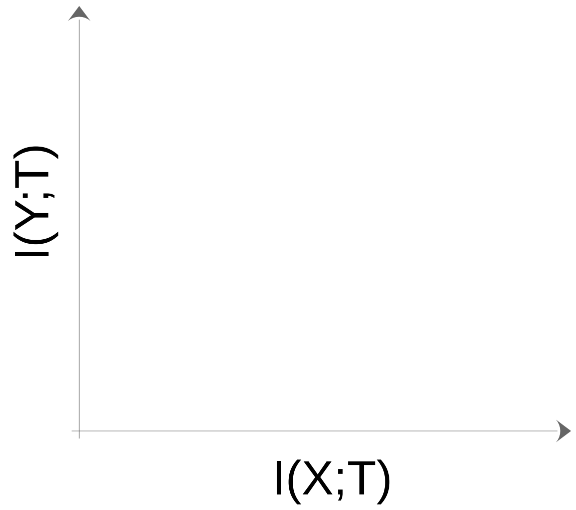


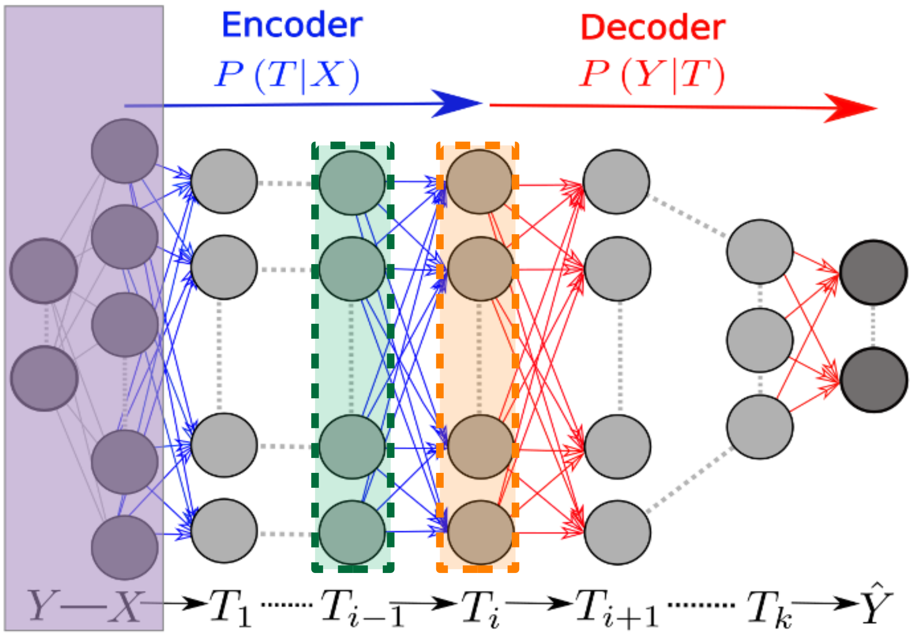


$$\text{Markov Chain}: Y \leftrightarrow X \rightarrow T \rightarrow \hat{Y}$$

$$\text{Data}: \{(x_i, y_i)\}_{i=1}^{N} \sim p(x, y)$$

Information Plane

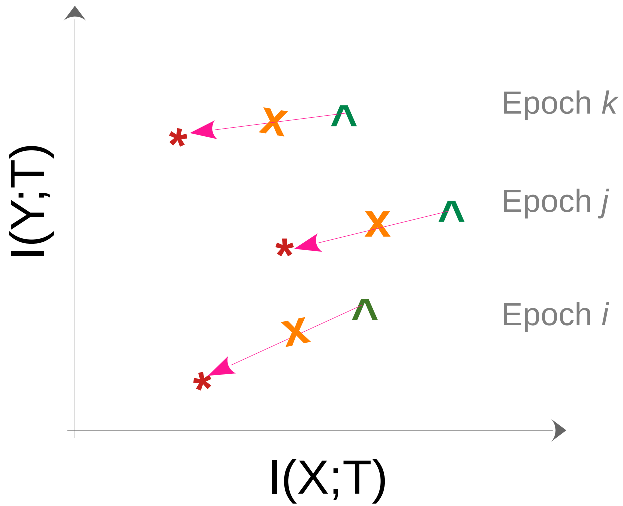


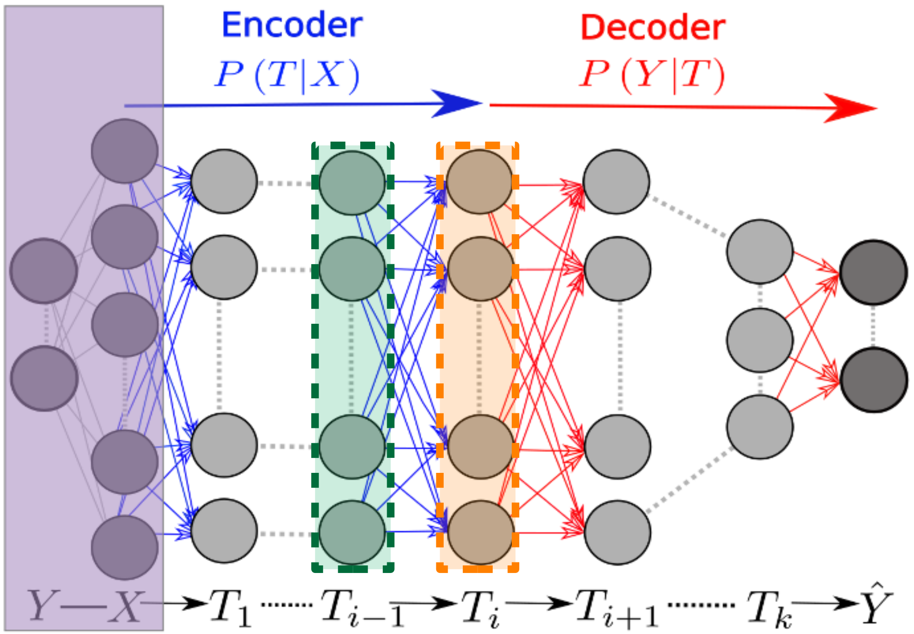


$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$

Information Plane

$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$



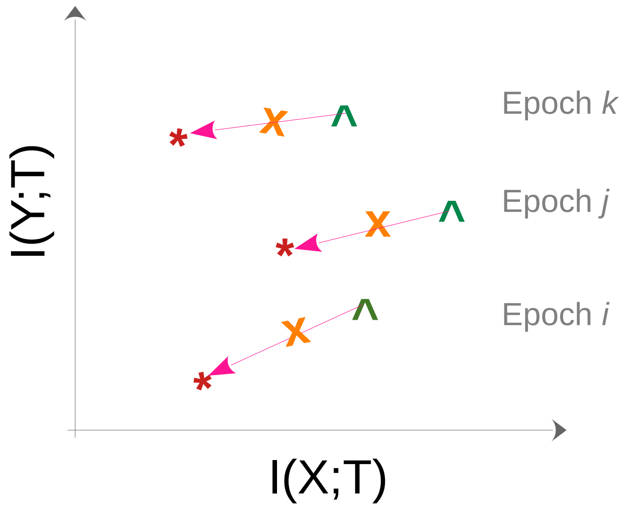


A point for each epoch and T_i ...

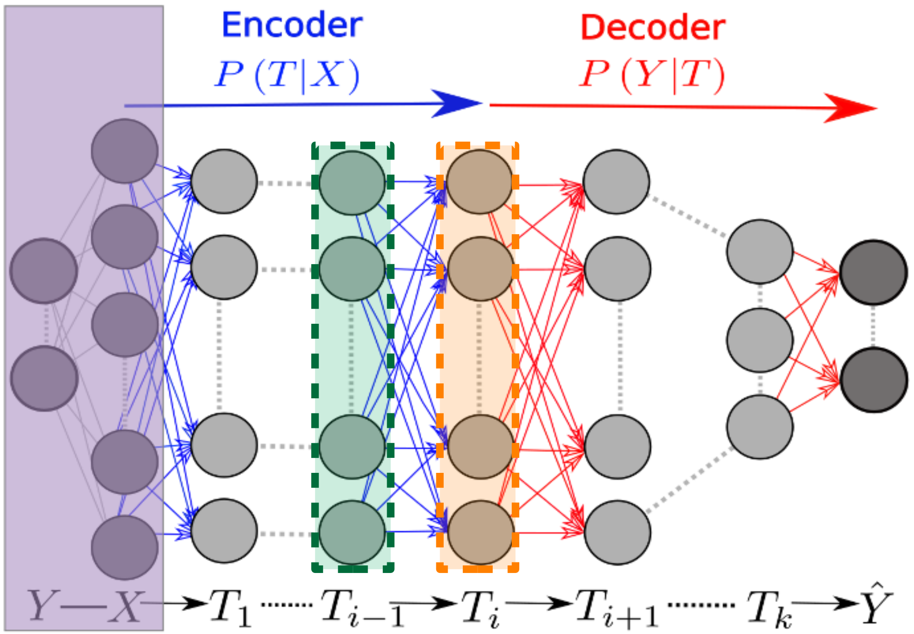
$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$

Information Plane

$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$



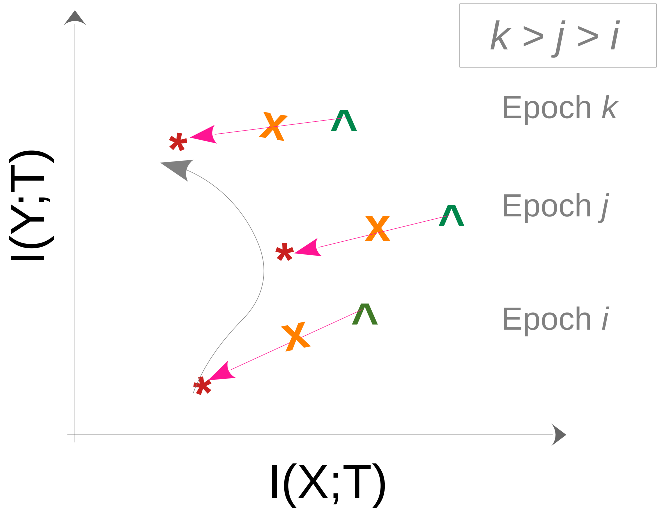
$$I(X; T_{i-1}) \geq I(X; T_i) \geq I(X; T_{i+1})$$
$$I(Y; T_{i-1}) \geq I(Y; T_i) \geq I(Y; T_{i+1})$$

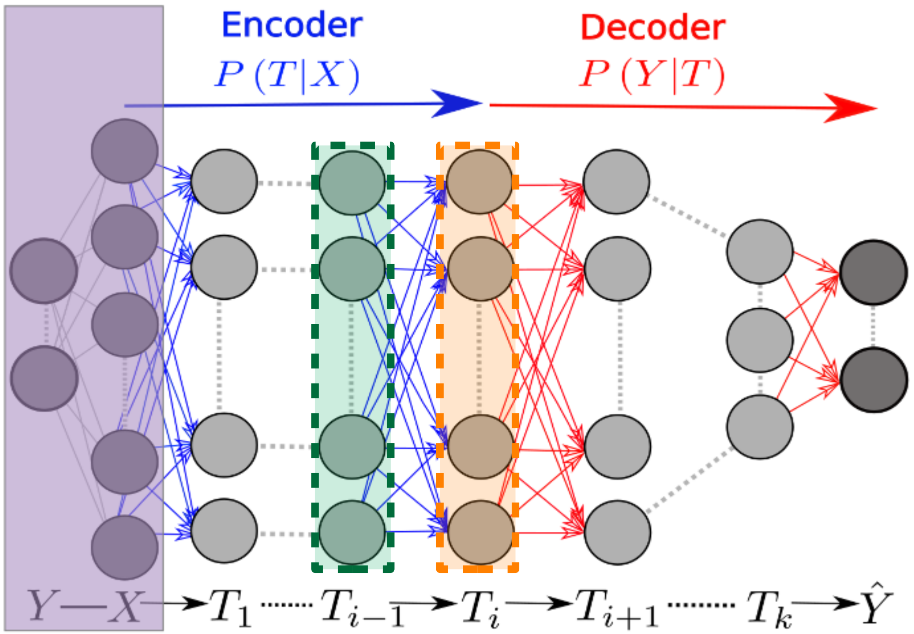


$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$

Information Plane

$$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$$



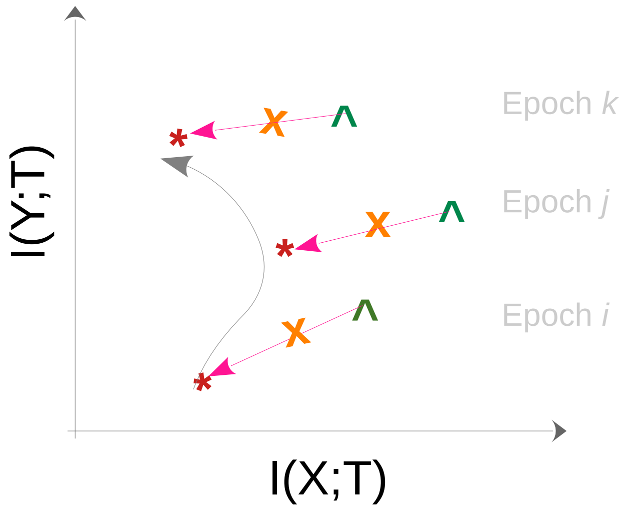


$$I(X;\ T_{i-1}) \geq I(X;\ T_i) \geq I(X;\ T_{i+1})$$
$$I(Y;\ T_{i-1}) \geq I(Y;\ T_i) \geq I(Y;\ T_{i+1})$$

$$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$$

Information Plane

$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$

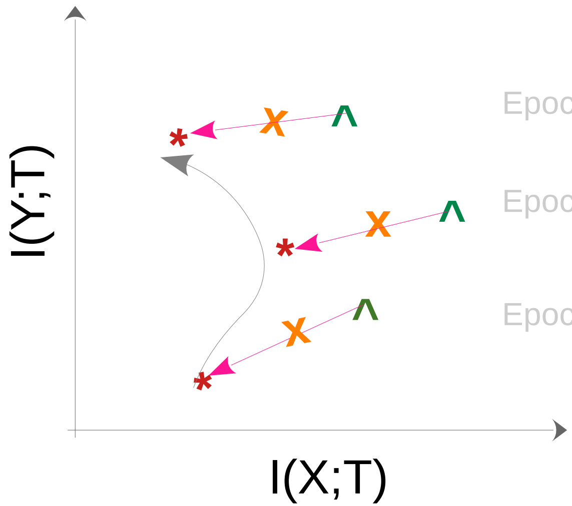


$I(X; T_{i-1}) \geq I(X; T_i) \geq I(X; T_{i+1})$
$I(Y; T_{i-1}) \geq I(Y; T_i) \geq I(Y; T_{i+1})$

IDEALLY … in coding ...
- $I(T;X)$ ↔ as LOW as possible (min Rate)
- $I(T;Y)$ ↔ as HIGH as possible (min Distortion)

Information Plane

$Y \rightarrow X \rightarrow \ldots \rightarrow T_{i-1} \rightarrow T_i \rightarrow T_{i+1} \rightarrow \ldots \rightarrow \hat{Y}$



$I(X; T_{i-1}) \geq I(X; T_i) \geq I(X; T_{i+1})$
$I(Y; T_{i-1}) \geq I(Y; T_i) \geq I(Y; T_{i+1})$

IDEALLY … in coding ...

- $I(T;X)$ ↔ as LOW as possible (min Rate)
- $I(T;Y)$ ↔ as HIGH as possible (min Distortion)

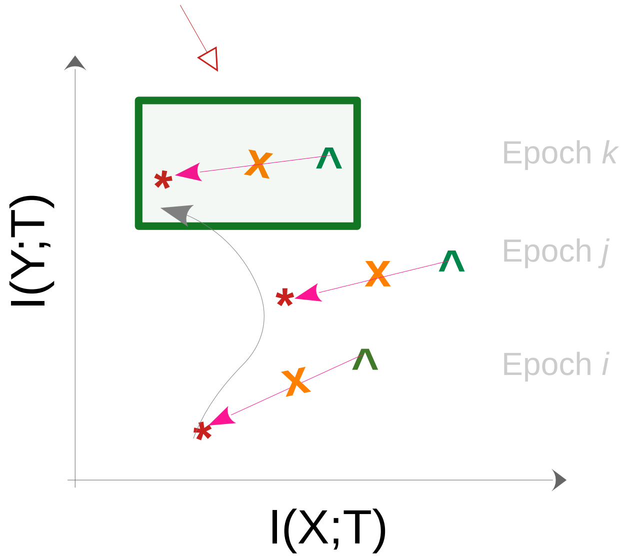
IDEALLY … in learning ...

- $I(T;X)$ ↔ as LOW as possible (discard irrelevant info)
- $I(T;Y)$ ↔ as HIGH as possible (keep relevant info)

MVIP2022
Shahid Chamran University of Ahvaz

Information Plane

Ideal solution

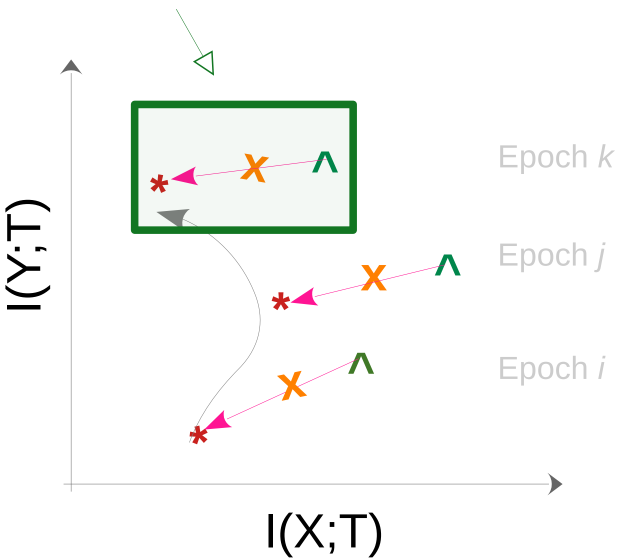


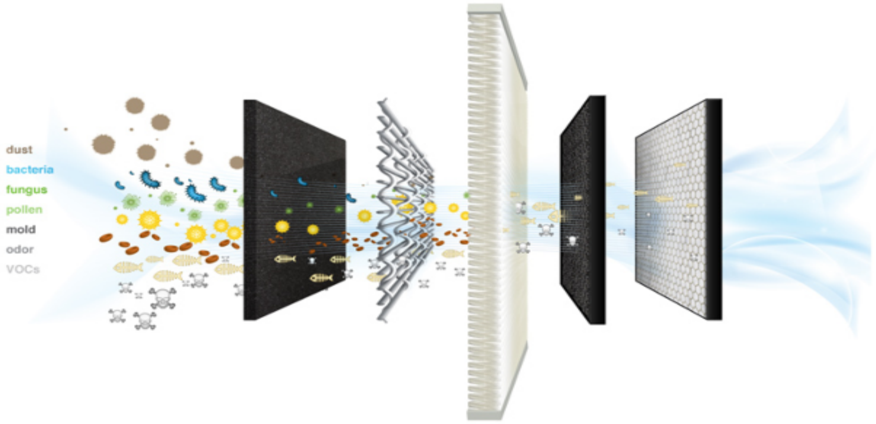
IDEALLY … in coding ...
- $I(T;X)$ ↔ as LOW as possible (min Rate)
- $I(T;Y)$ ↔ as HIGH as possible (min Distortion)

IDEALLY … in learning ...
- $I(T;X)$ ↔ as LOW as possible (discard irrelevant info)
- $I(T;Y)$ ↔ as HIGH as possible (keep relevant info)

Information Plane

Ideal solution

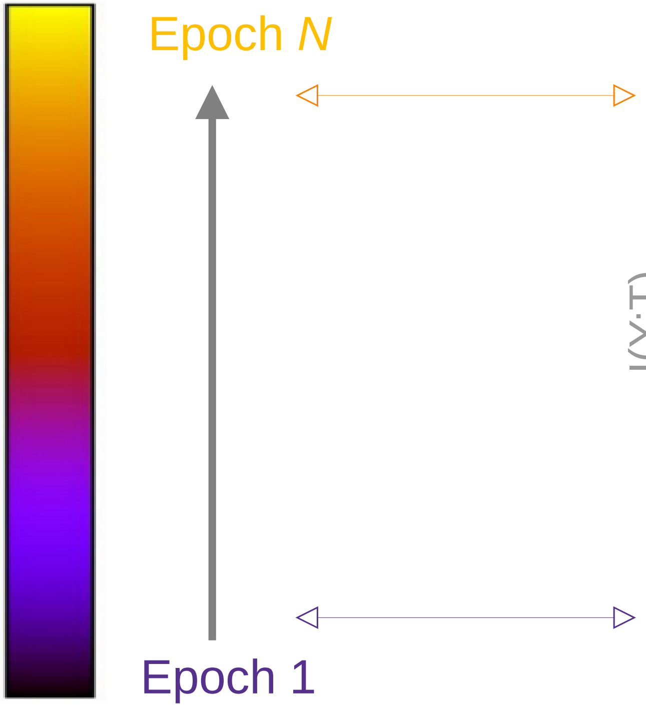


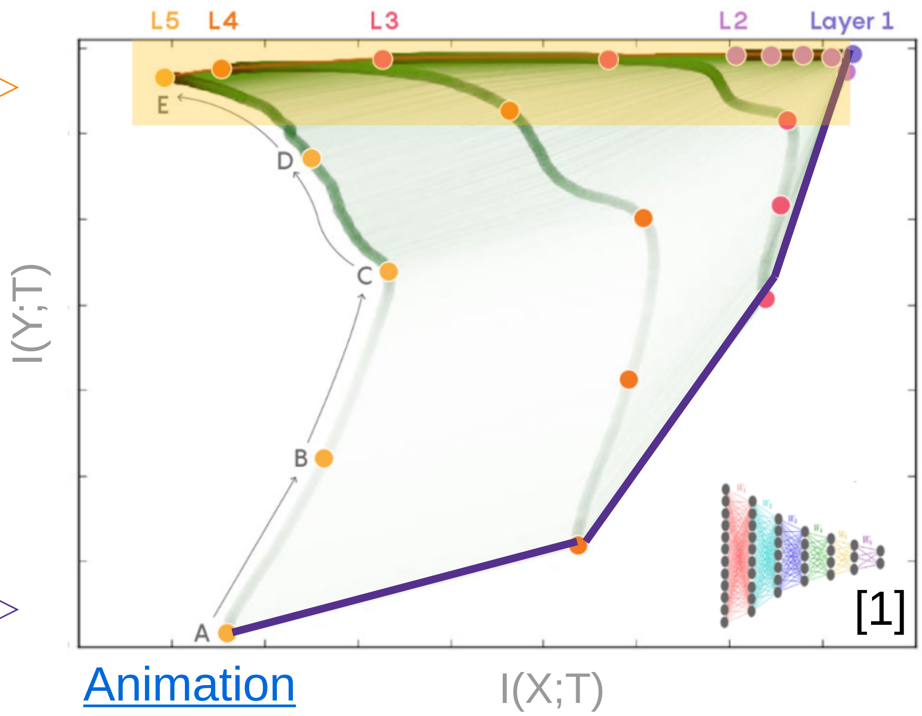


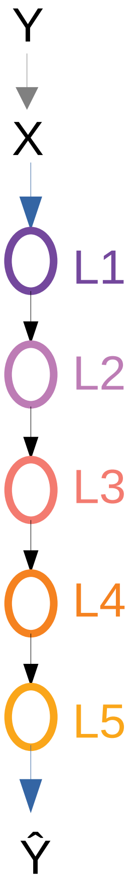
IDEALLY … in learning ...

– $I(T;X)$ ↔ as LOW as possible (discard irrelevant info)
– $I(T;Y)$ ↔ as HIGH as possible (keep relevant info)

Learning from IB view



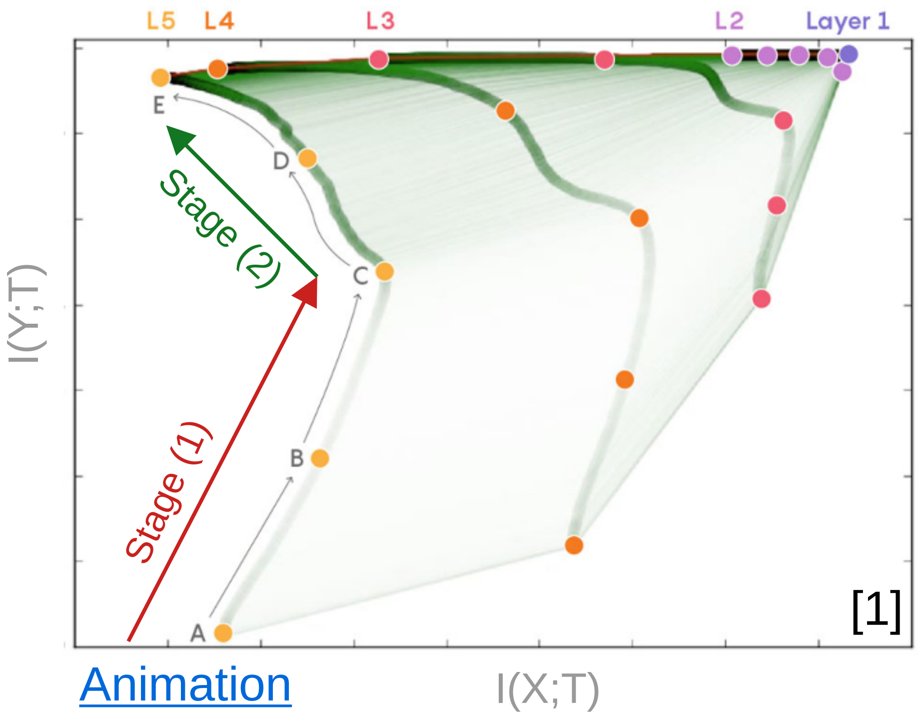




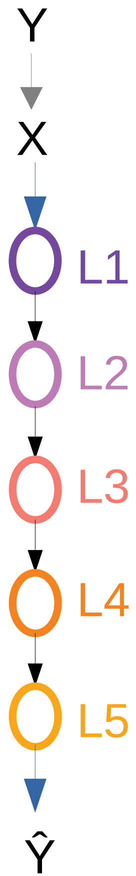
Animation

Learning from IB view

- Two distinct stages ...
  - Stage (1): A → C
  - Stage (2): C → E

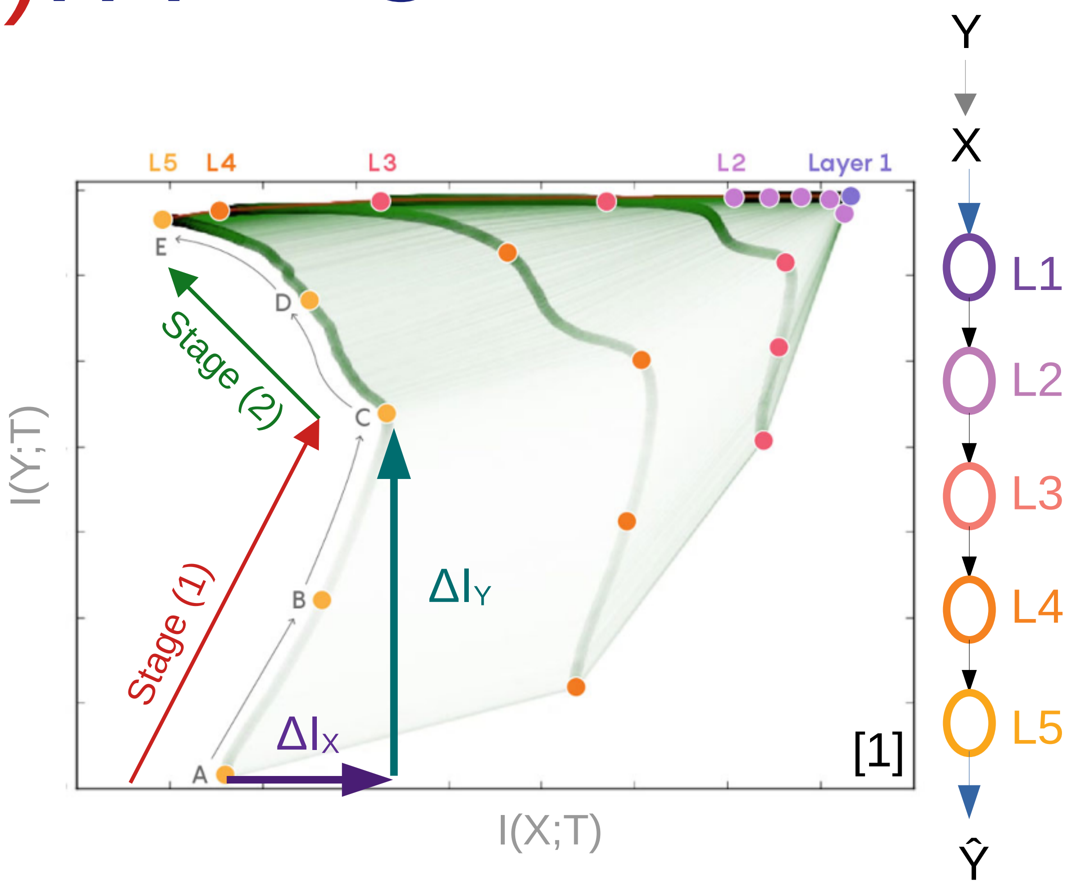


[Animation]



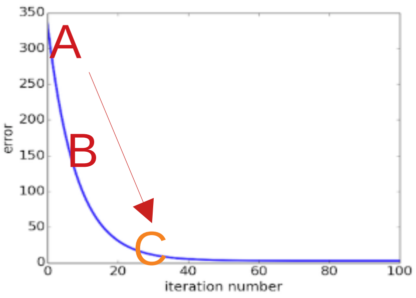
Stage (1): A → C

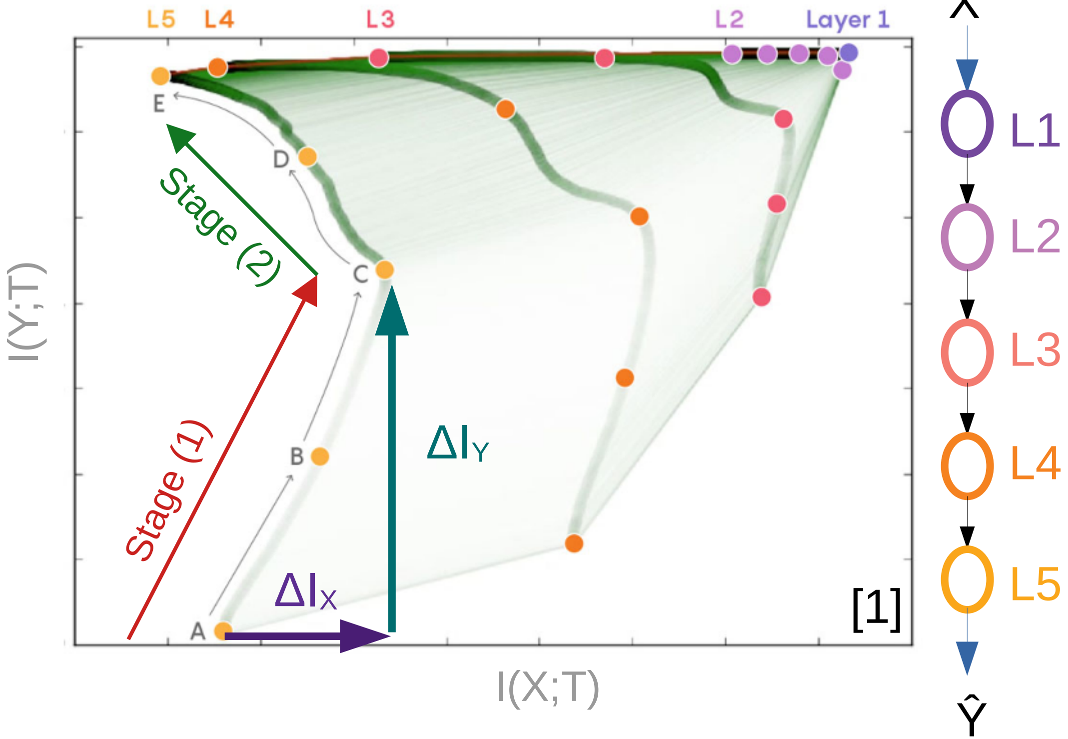
- $\Delta I_Y > 0$ and $\Delta I_X > 0$
  - Fitting
- $\Delta Empirical\_risk \leq 0$
- Fast



Stage (1): A → C

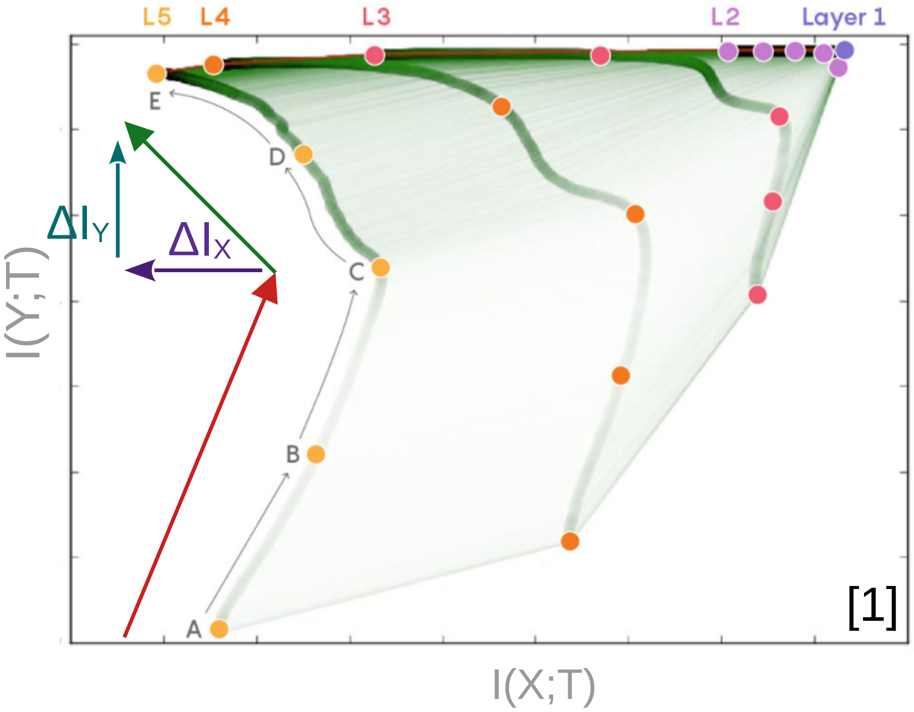
- $\Delta I_Y > 0$ and $\Delta I_X > 0$
  - Fitting
- $\Delta Empirical\_risk \leq 0$
- Fast

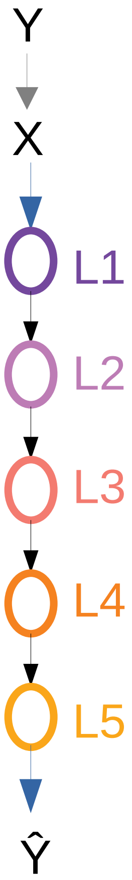




Stage (2): C → E

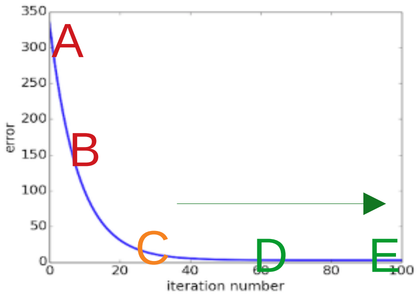
- $\Delta I_Y > 0$ and $\Delta I_X < 0$
  - Compression
  - Forget irrelevant info
- $\Delta Empirical\_risk \approx 0$
- Slow

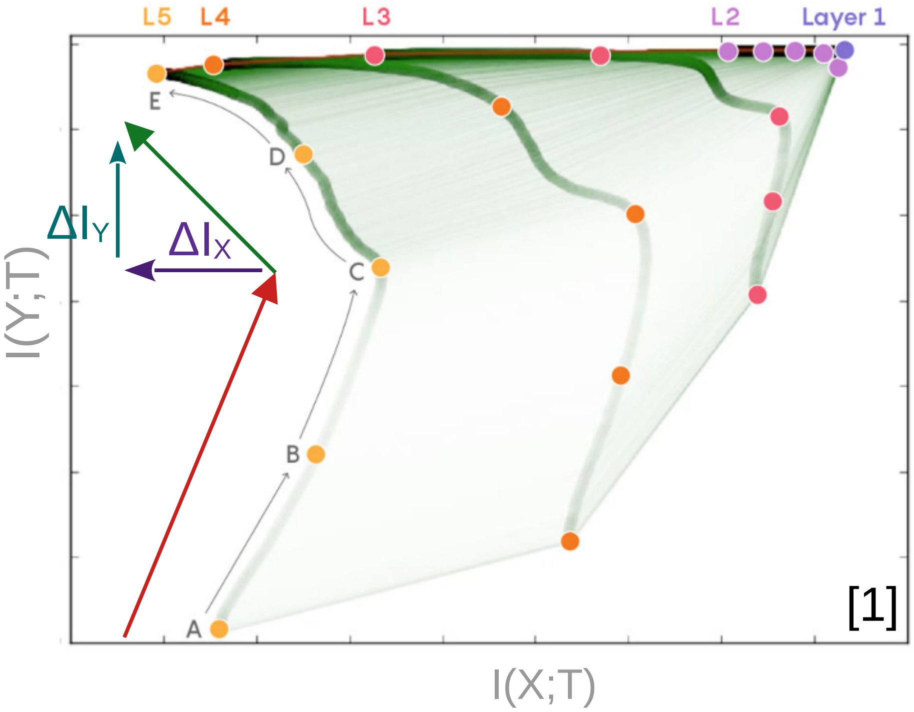


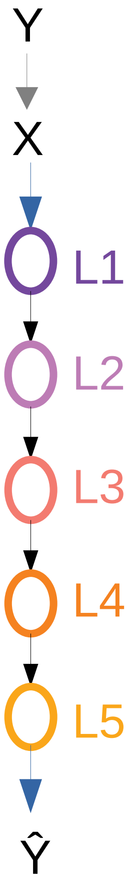


Stage (2): C → E

- $\Delta I_Y > 0$ and $\Delta I_X < 0$
  - Compression
  - Forget irrelevant info
- $\Delta Empirical\_risk \approx 0$
- Slow

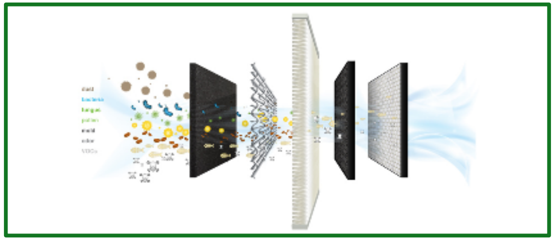


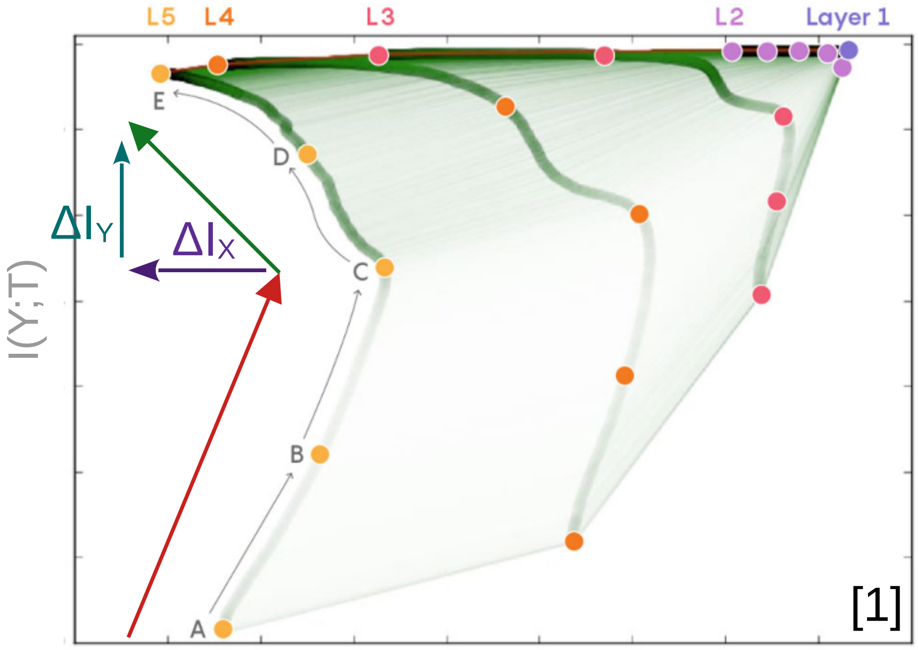


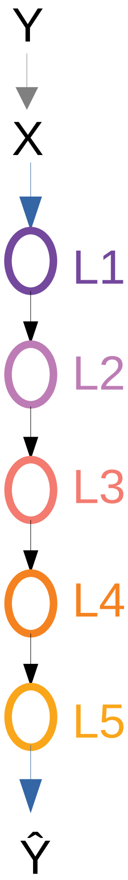


Stage (2): C → E

- $\Delta I_Y > 0$ and $\Delta I_X < 0$
  - Compression
  - Forget irrelevant info
- $\Delta Empirical\_risk \approx 0$
- Slow

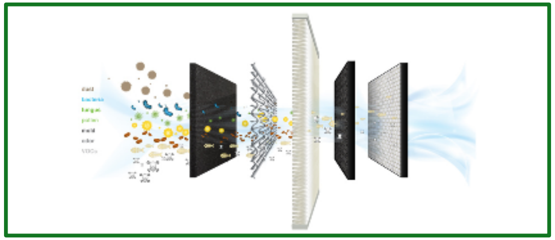




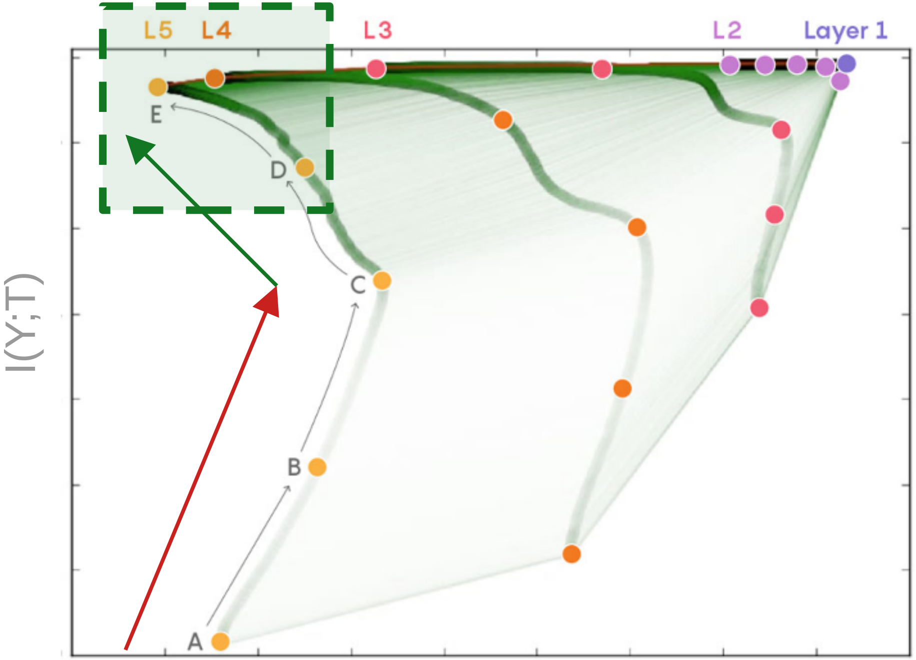


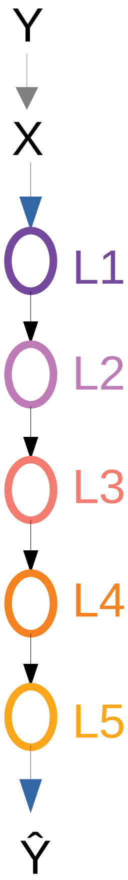
Stage (2): C → E

- $\Delta I_Y > 0$ and $\Delta I_X < 0$
  - Compression
  - Forget irrelevant info
- $\Delta Empirical\_risk \approx 0$
- Slow



Ideal solution ...

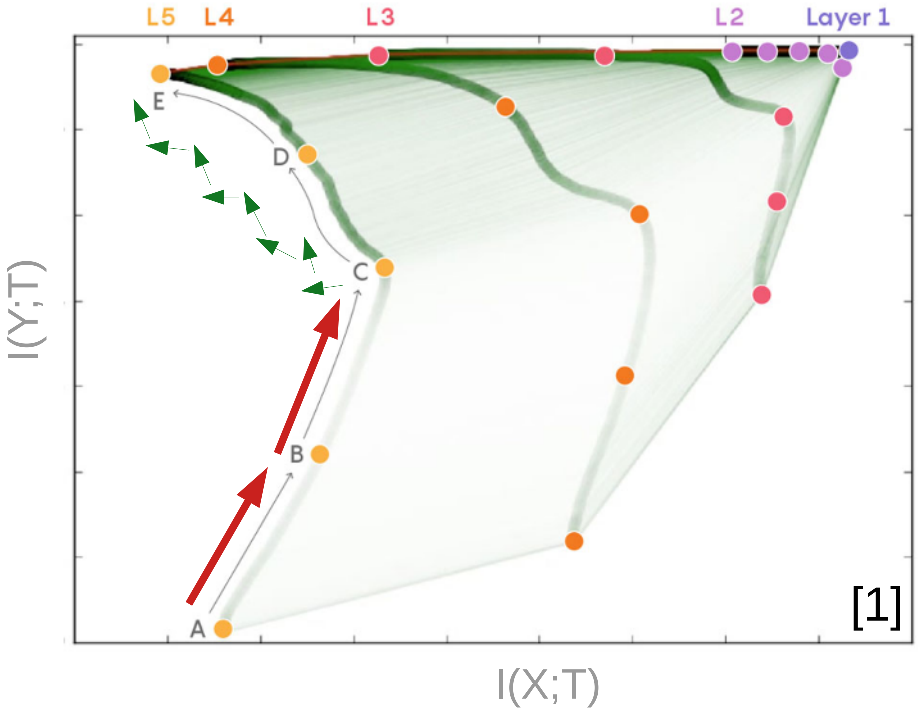


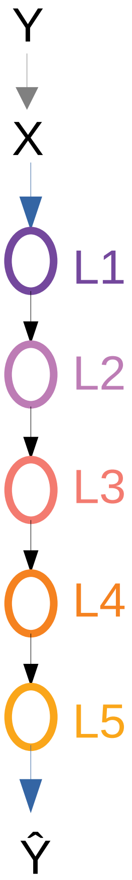


Learning has two stages ...

1) Drift

2) Diffusion



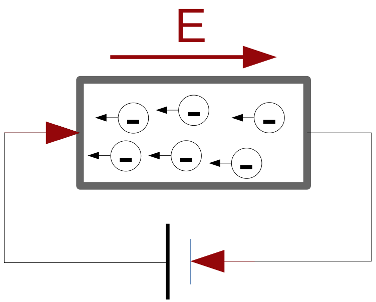


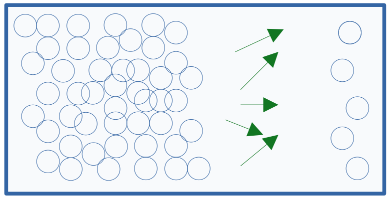
Learning has two stages ...

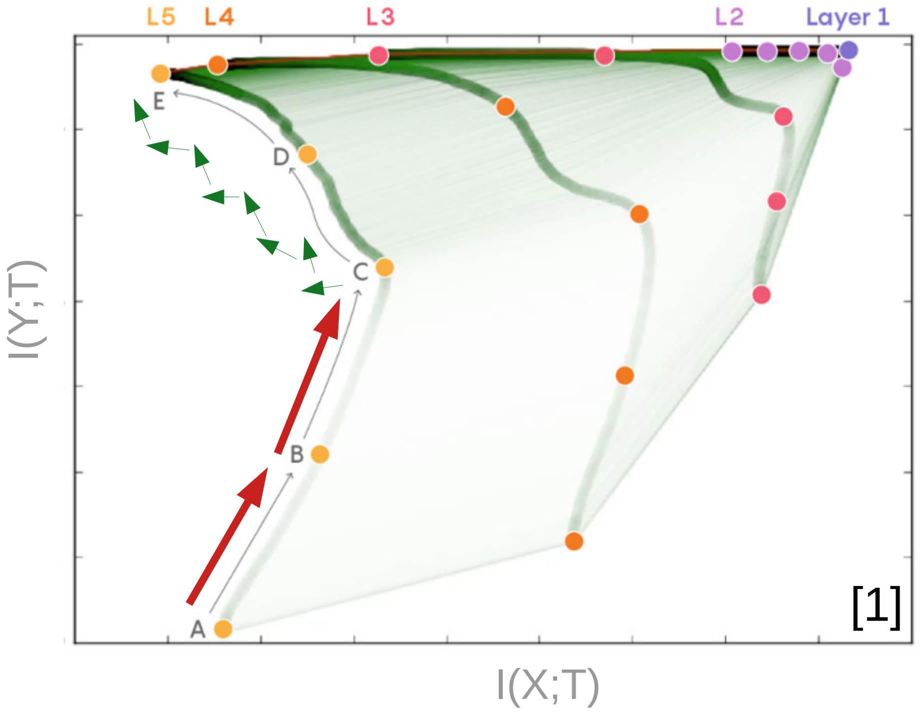
<u>1) Drift</u>

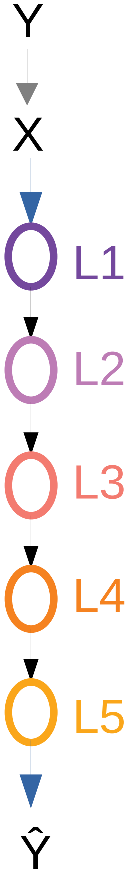
<u>2) Diffusion</u>

Random walk





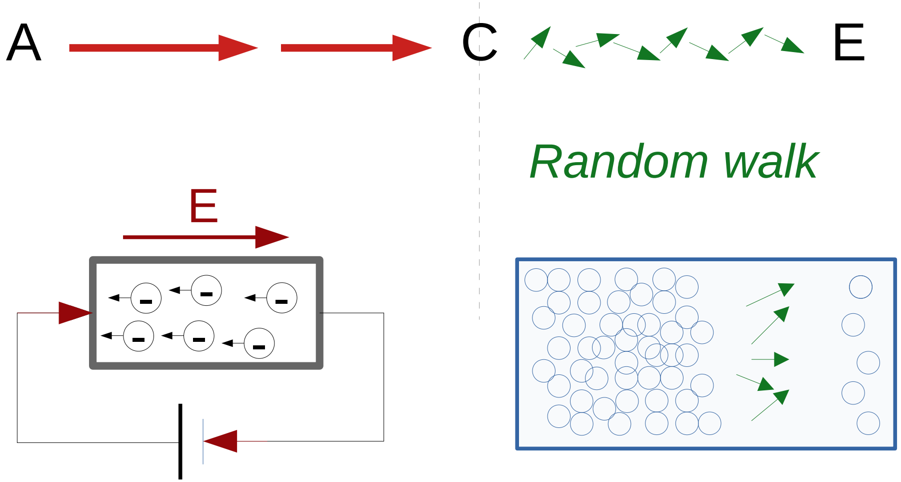


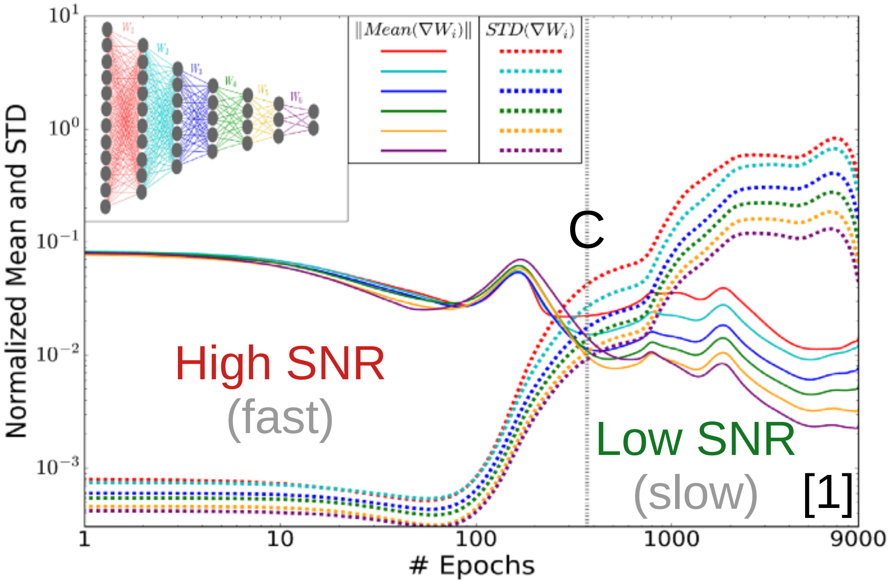


SNR of Gradient

1) Drift

2) Diffusion





$$\mathrm{SNR} \triangleq \frac{Mean(\|\nabla W_l\|)}{STD(\|\nabla W_l\|)}$$

MVIP2022
Shahid Chamran University of Ahvaz

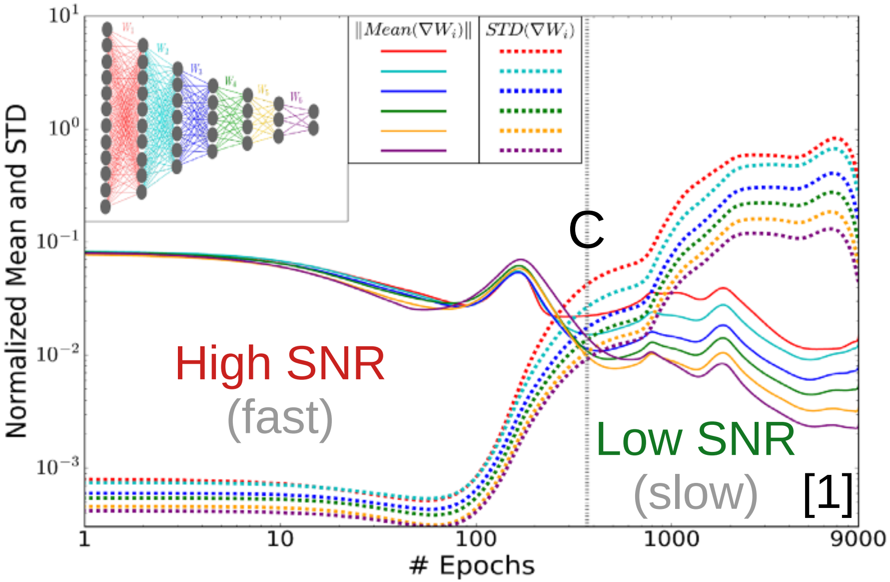
SNR of Gradient

1) Drift

2) Diffusion

Random walk

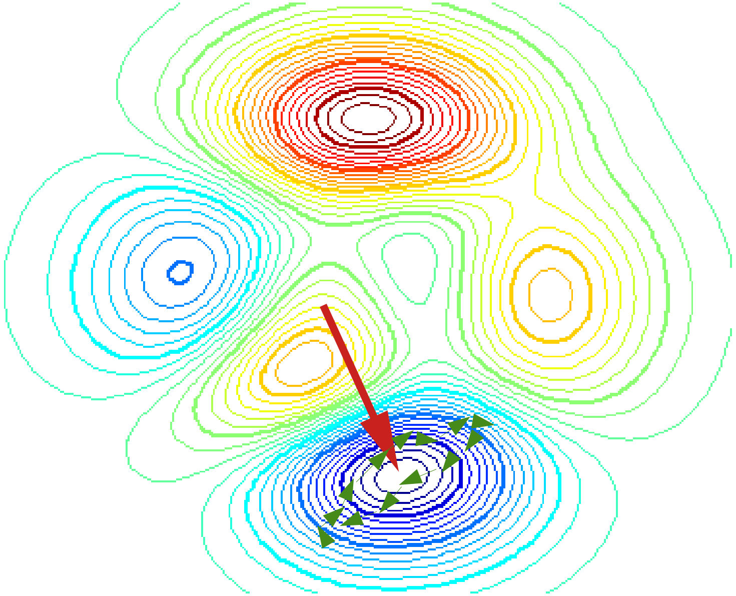
Stochasticity during diffusion is responsible for generalisation ...



$$\mathrm{SNR} \triangleq \frac{Mean(\|\nabla W_l\|)}{STD(\|\nabla W_l\|)}$$

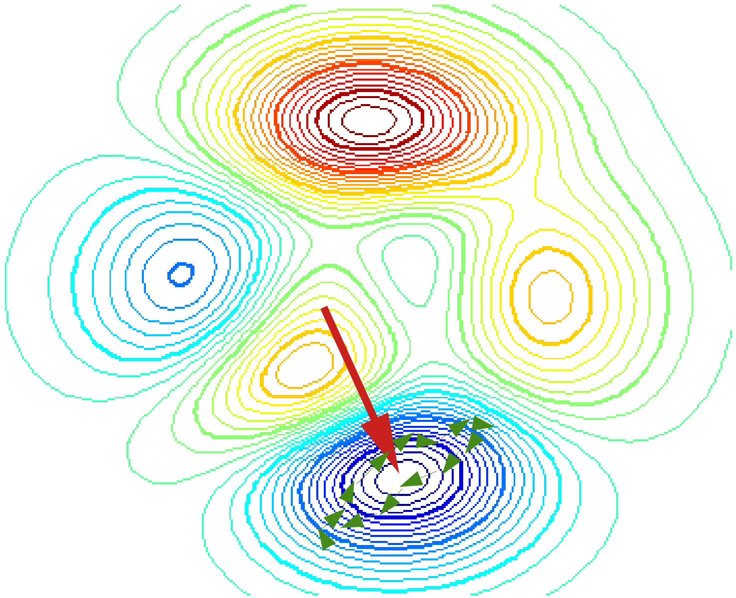
MVIP2022
Shahid Chamran University of Ahvaz

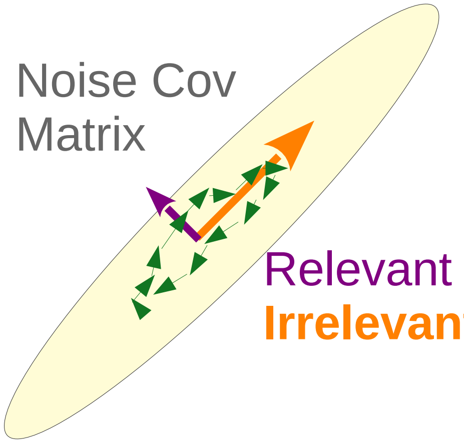
Stochasticity of the Diffusion Improves the Generalisation



Drift (A → C) → High SNR
Diffusion (C → E) → Low SNR

Stochasticity of the Diffusion Improves the Generalisation





Drift (A→C) → High SNR
Diffusion (C→ E) → Low SNR

Diffusion's stochasticity ...

→ Add noise to irrelevant features

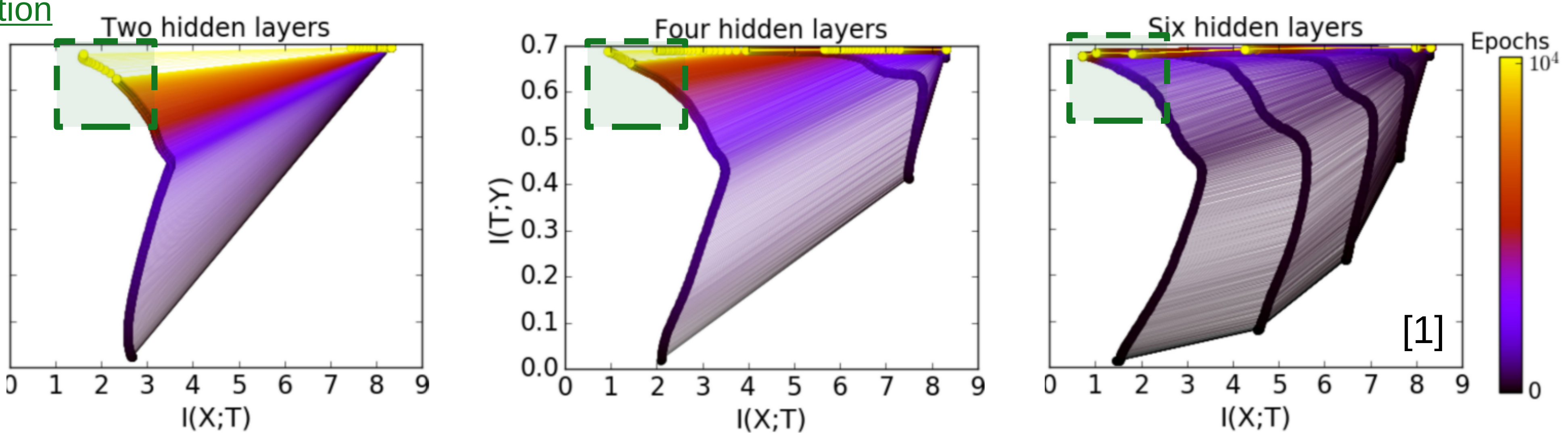
→ Forget irrelevant details

MVIP2022

Shahid Chamran University of Ahvaz

Effect of … Depth

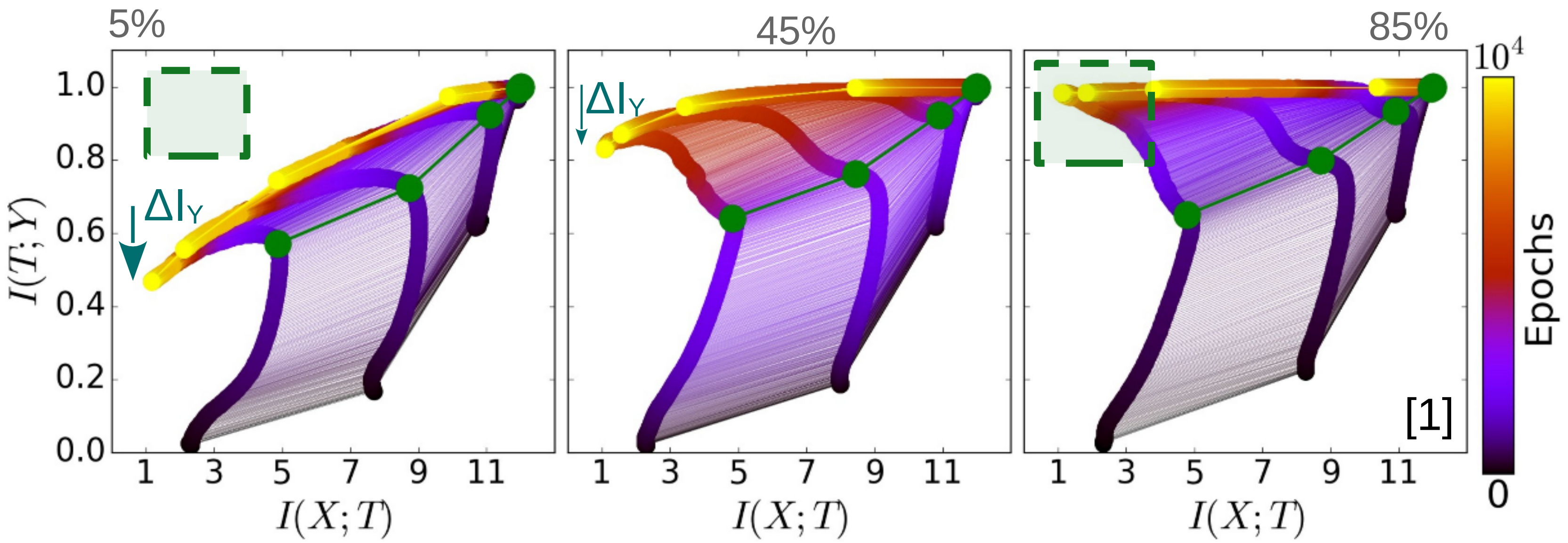
Ideal solution



* Deeper network → Faster training ...
==>> Better generalisation with fewer epochs

Effect of … Training Data Amount (1)

Ideal solution

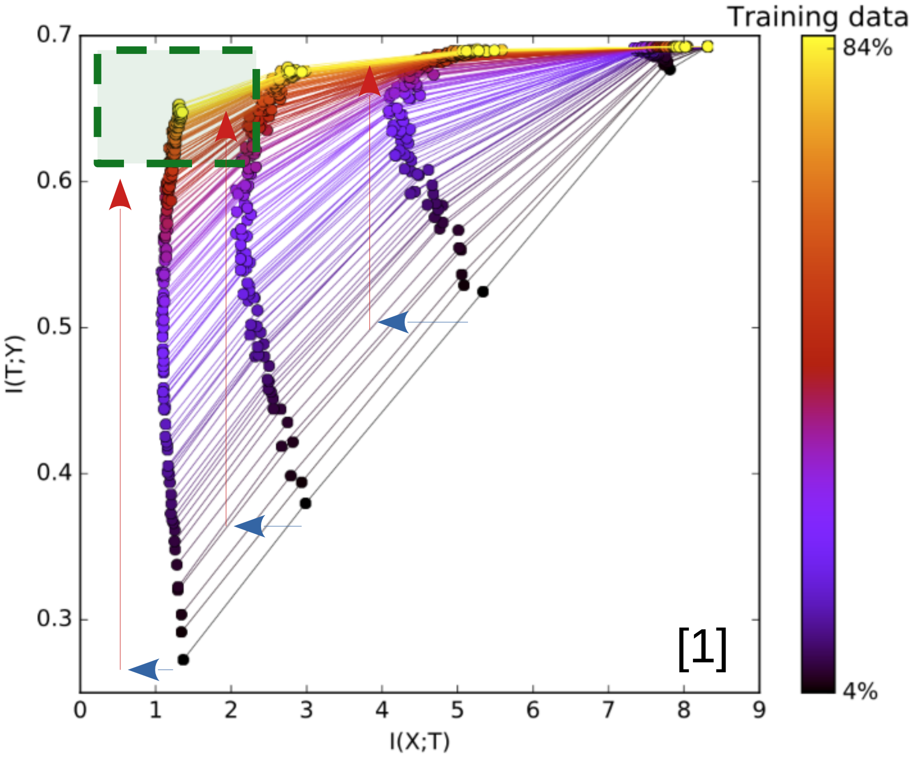


* Less data … may lead to $\Delta I_Y < 0$ & never reaching

MVIP2022
Shahid Chamran University of Ahvaz

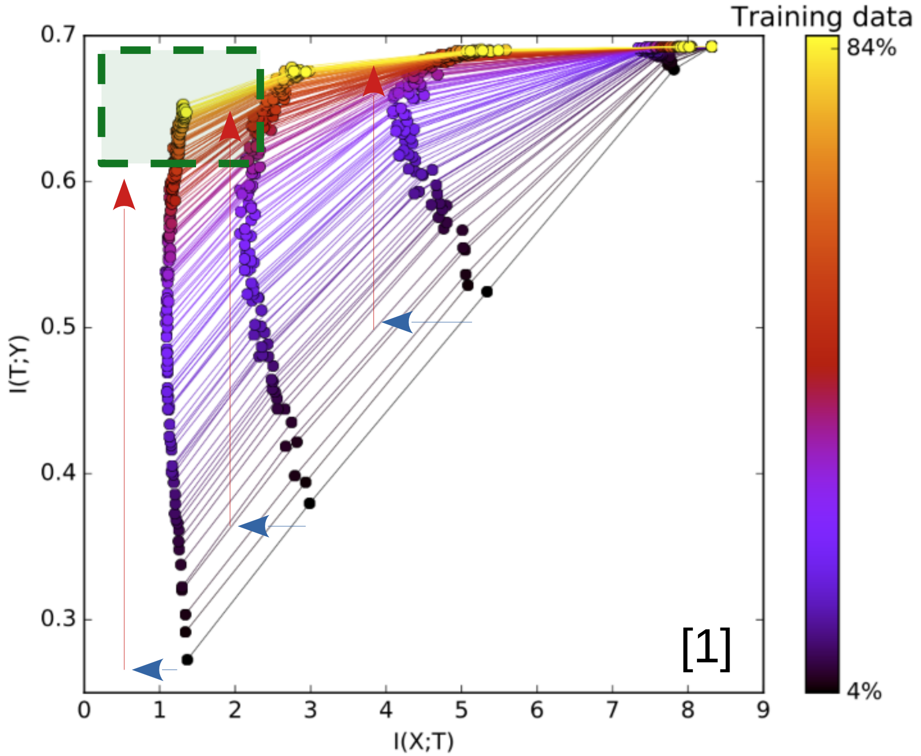
Effect of … Training Data Amount (2)

- More training data …
  - I_X: Minor reduction ↓
  - I_Y: Major increase ↑



Effect of … Training Data Amount (2)

- More training data …
  - I_X: Minor reduction ↓
  - I_Y: Major increase ↑
- Good generalisation
  - I_X: low, I_Y: high



MVIP2022
Shahid Chamran University of Ahvaz

Effect of … Batch Size (BS)

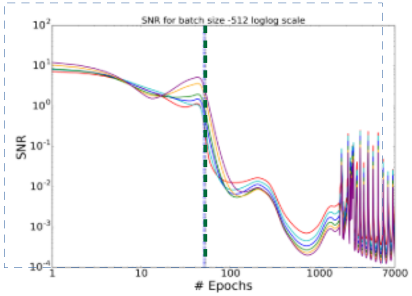
- The smaller the BS, the higher the stochasticity of GD

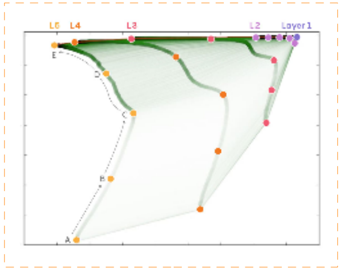
Effect of … Batch Size (BS)

- The smaller the BS, the higher the stochasticity of GD

Drift to *diffusion* transition:

$$\text{argmin}\ \frac{d}{dt}SNR \approx \text{argmax}\ I(X;T)$$



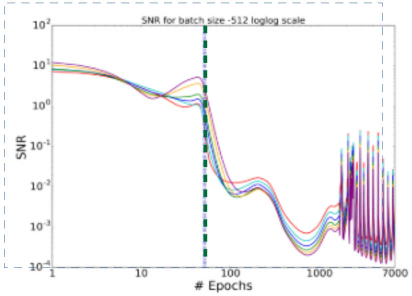


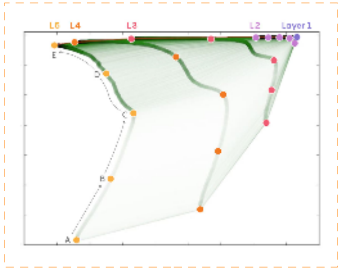
Effect of … Batch Size (BS)

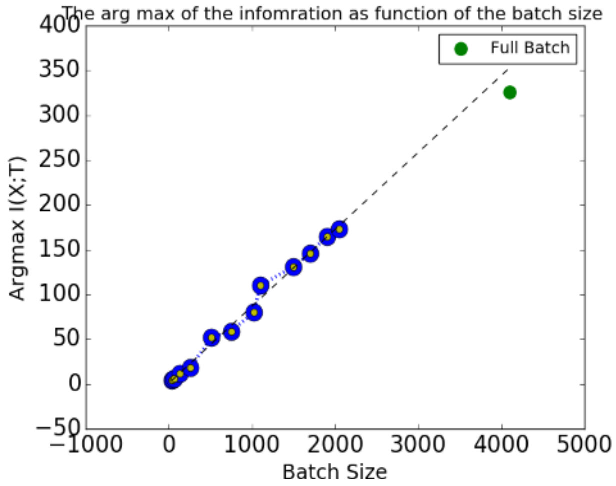
- The smaller the BS, the higher the stochasticity of GD

Drift to *diffusion* transition:

$$\text{argmin}\ \frac{d}{dt}SNR \approx \text{argmax}\ I(X;T)$$



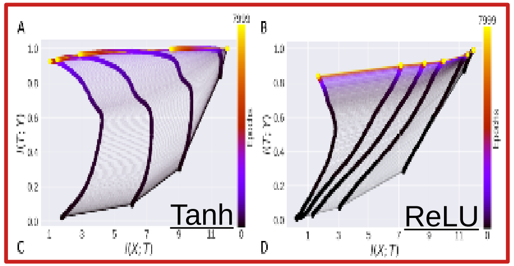




* The smaller the BS, the faster the transition to diffusion ...

Criticisms (1)

- Two-phase process is **NOT** generic [3]!
  - ReLU … Adaptive binning helps [4] …



OpenReview.net Search OpenReview... Login

← Go to ICLR 2019 Conference homepage

Adaptive Estimators Show Information Compression in Deep Neural Networks

Ivan Chelombiev, Conor Houghton, Cian O'Donnell

27 Sept 2018 (modified: 21 Feb 2019) ICLR 2019 Conference Blind Submission Readers: Everyone Show Bibtex Show Revisions

Keywords: deep neural networks, mutual information, information bottleneck, noise, L2 regularization

TL;DR: We developed robust mutual information estimates for DNNs and used them to observe compression in networks with non-saturating activation functions

Abstract: To improve how neural networks function it is crucial to understand their learning process. The information bottleneck theory of deep learning proposes that neural networks achieve good generalization by compressing their representations to disregard information that is not relevant to the task. However, empirical evidence for this theory is conflicting, as compression was only observed when networks used saturating activation functions. In contrast, networks with non-saturating activation functions achieved comparable levels of task performance but did not show compression. In this paper we developed more robust mutual information estimation techniques, that adapt to hidden activity of neural networks and produce more sensitive measurements of activations from all functions, especially unbounded functions. Using these adaptive estimation techniques, we explored compression in networks with a range of different activation functions. With two improved methods of estimation, firstly, we show that saturation of the activation function is not required for compression, and the amount of compression varies between different activation functions. We also find that there is a large amount of variation in compression between different network initializations. Secondary, we see that L2 regularization leads to significantly increased compression, while preventing overfitting. Finally, we show that only compression of the last layer is positively correlated with generalization.

OpenReview.net Search ICLR 2018 Conference Login

← Go to ICLR 2018 Conference homepage

On the Information Bottleneck Theory of Deep Learning

Andrew Michael Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan Daniel Tracey, David Daniel Cox

15 Feb 2018 (modified: 24 Feb 2018) ICLR 2018 Conference Blind Submission Readers: Everyone Show Bibtex Show Revisions

Abstract: The practical successes of deep neural networks have not been matched by theoretical progress that satisfyingly explains their behavior. In this work, we study the information bottleneck (IB) theory of deep learning, which makes three specific claims: first, that deep networks undergo two distinct phases consisting of an initial fitting phase and a subsequent compression phase; second, that the compression phase is causally related to the excellent generalization performance of deep networks; and third, that the compression phase occurs due to the diffusion-like behavior of stochastic gradient descent. Here we show that none of these claims hold true in the general case. Through a combination of analytical results and simulation, we demonstrate that the information plane trajectory is predominantly a function of the neural nonlinearity employed: double-sided saturating nonlinearities like tanh yield a compression phase as neural activations enter the saturation regime, but linear activation functions and single-sided saturating nonlinearities like the widely used ReLU in fact do not. Moreover, we find that there is no evident causal connection between compression and generalization: networks that do not compress are still capable of generalization, and vice versa. Next, we show that the compression phase, when it exists, does not arise from stochasticity in training by demonstrating that we can replicate the IB findings using full batch gradient descent rather than stochastic gradient descent. Finally, we show that when an input domain consists of a subset of task-relevant and task-irrelevant information, hidden representations do compress the task irrelevant information, although the overall information about the input may monotonically increase with training time, and that this compression happens concurrently with the fitting process rather than during a subsequent compression period.

TL;DR: We show that several claims of the information bottleneck theory of deep learning are not true in the general case.

Keywords: information bottleneck, deep learning, deep linear networks

21 Replies

Criticisms (2)

- Two-phase process is **NOT** generic [3]!
  - ReLU … Adaptive binning helps [4] …

- No causal relationship between stochasticity of SGD (compression/forgetting) & generalisation [3]
  - i-RevNet [5] … good gen. w/o forgetting

Criticisms (3)

- Two-phase process is **NOT** generic [3]!
  - ReLU … Adaptive binning helps [4] …

- No causal relationship between stochasticity of SGD (compression/forgetting) & generalisation [3]
  - i-RevNet [5] … good gen. w/o forgetting

- Computing MI is challenging [6] … especially for *random* ***vectors***

Conclusion (Part I)

- Novelty: DNNs from Information Theory's perspective
- $I(X;T_i)$ an $I(Y;T_i)$ plotted in *information plane*
- Learning consists of two stages: 1) Drift, 2) Diffusion

- *Why DNNs generalise well?*
  - *Stochasticity of GD → Diffusion → forgetting irrelevant info*

MVIP2022

Shahid Chamran University of Ahvaz

Outlines (Part II)

- Information Bottleneck
- Over-parameterisation and Generalisation
- Interpretation/Visualisation of Filters/Activations

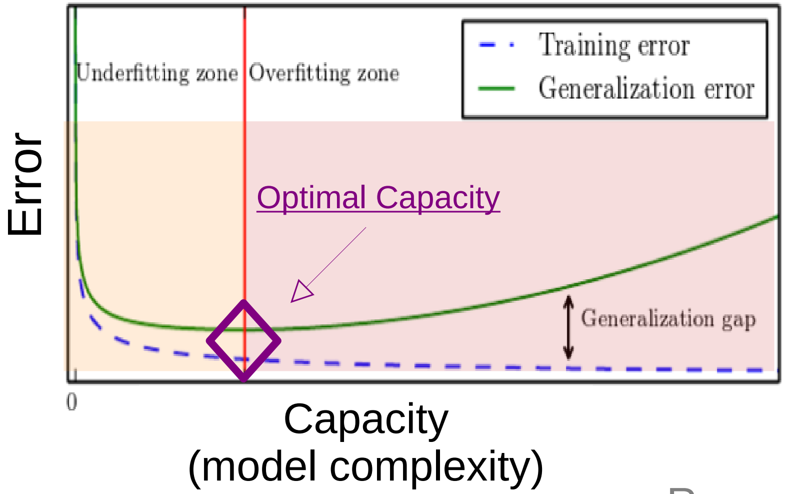
MVIP2022

Shahid Chamran University of Ahvaz

DNNs … Generalisation …

- Why do DNNs generalise well?

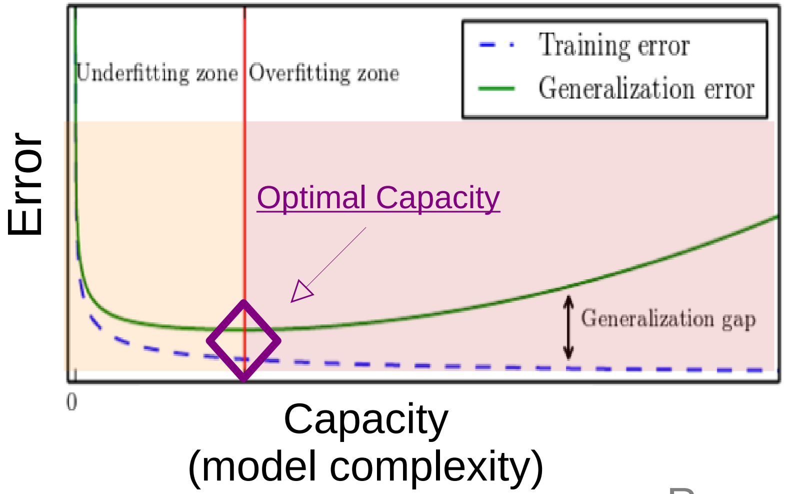
Classic wisdom …



DNNs … Generalisation …

- Why do DNNs generalise well?

Classic wisdom …

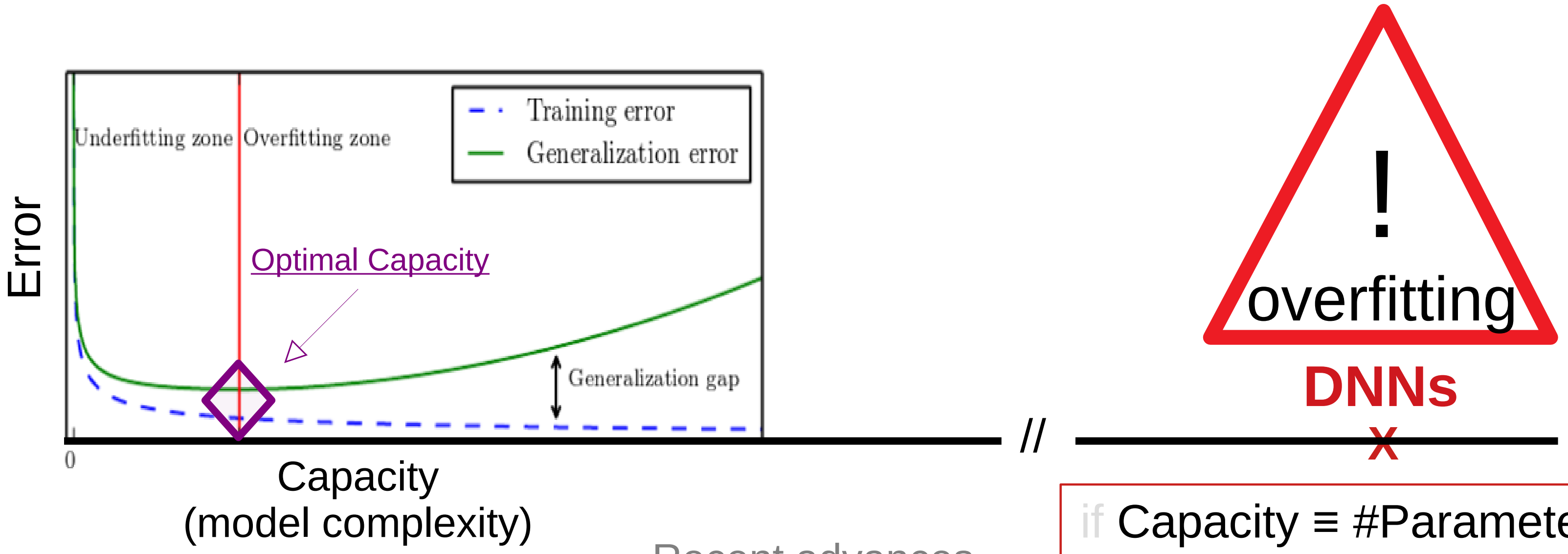


Underfitting: High Bias

Overfitting: High Variance

DNNs … Generalisation …

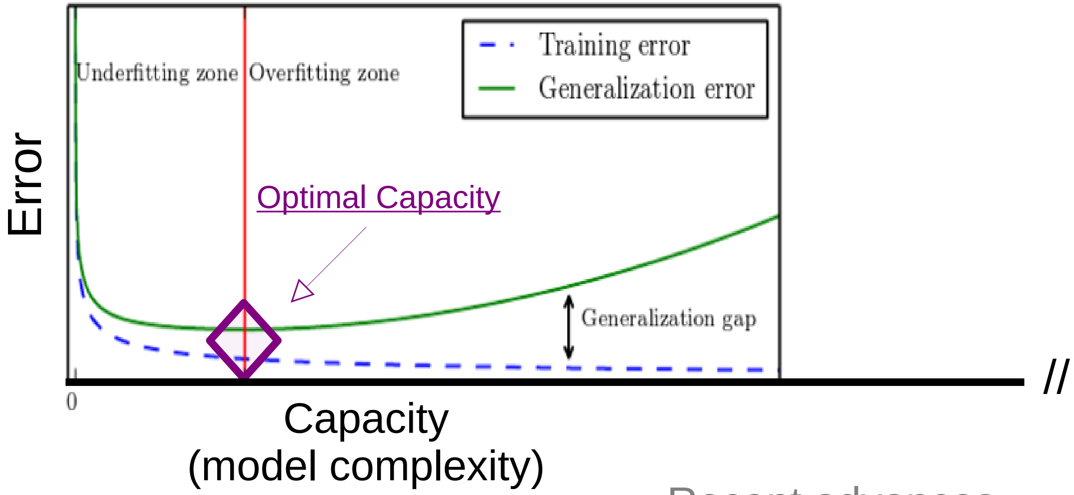
- Why do DNNs generalise well?

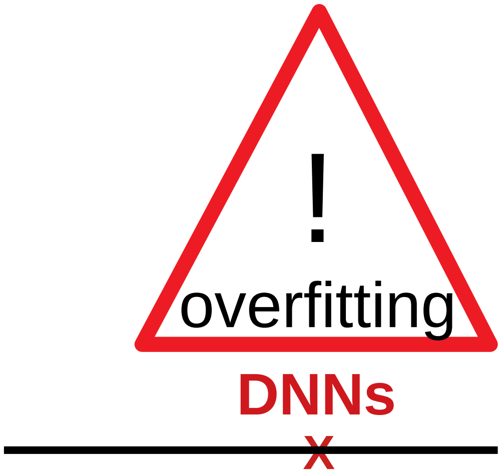


if Capacity ≡ #Parameters

DNNs … Generalisation …

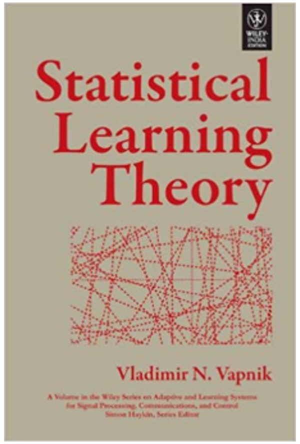
- Why do DNNs generalise well?
  - even when *over-parameterised* → *P/N >> 1*





if Capacity ≡ #Parameters

Generalisation Error



- Classic statistical learning theory ...
  - Upper bound for E_{gen} ↔ Capacity
  - Over-parameterisation (P/N >> 1) is bad!

$$E_{gen} = E_{test} - E_{train} \overset{\leq}{\propto} \frac{f_1(\#parameters)}{f_2(N)} \overset{e.g.}{=} \frac{f_1(VC\text{-}dim)}{f_2(N)}$$

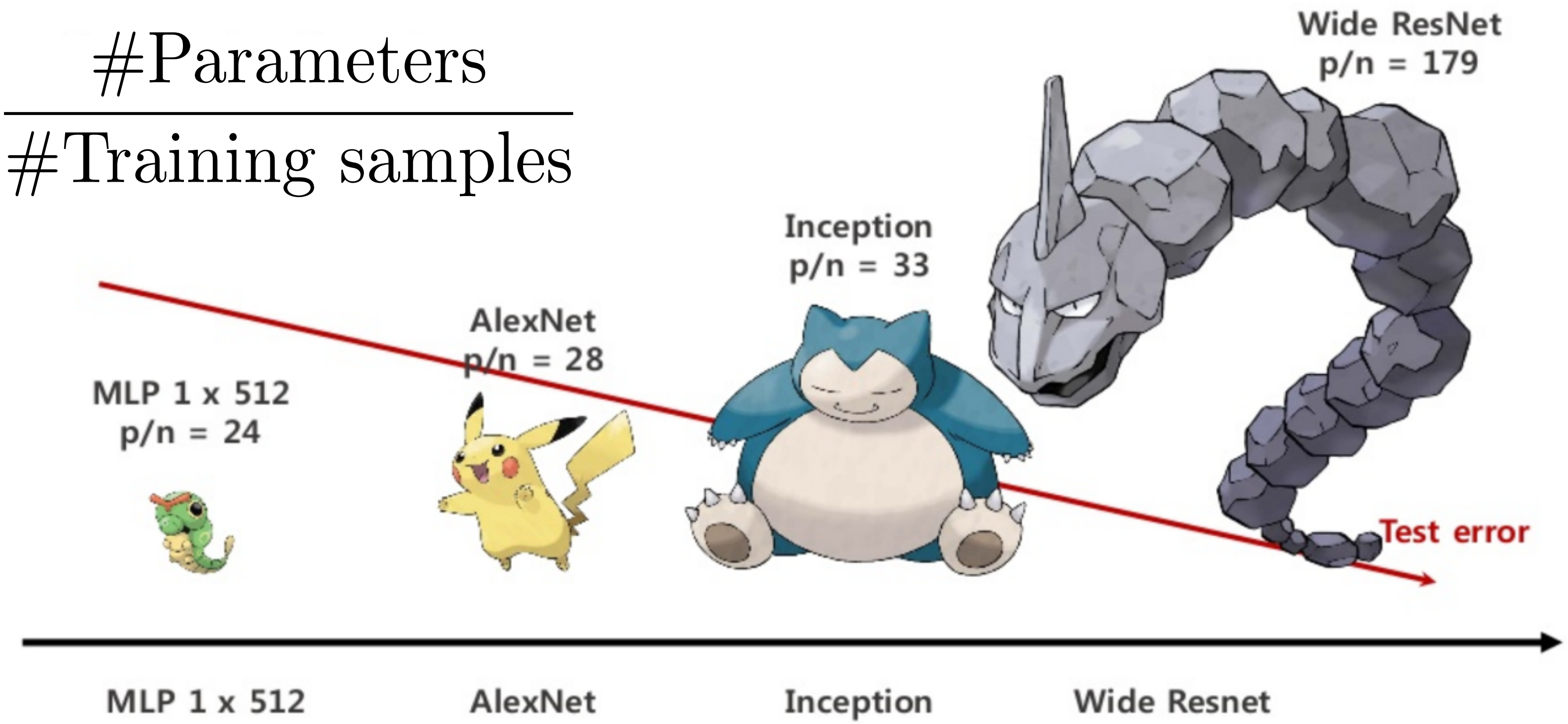
Over-parameterisation is good (1)

CIFAR-10	#train: 50,000	#parameter/#train
Inception	1,649,402	33
AlexNet	1,387,786	28
MLP 1x512	1,209,866	24
ImageNet	#train: 1,200,000	
Inception V3	23,885,392	20
AlexNet	61,100,840	51
ResNet-{18; 152}	11,689,512; 60,192,808	10; 50
VGG-{11;19}	132,863,336; 143,667,240	110; 120

[8]

Over-parameterisation is good (2)

$$p/n = \frac{\#\text{Parameters}}{\#\text{Training samples}}$$



[8]

If over-parametrisation is good ...

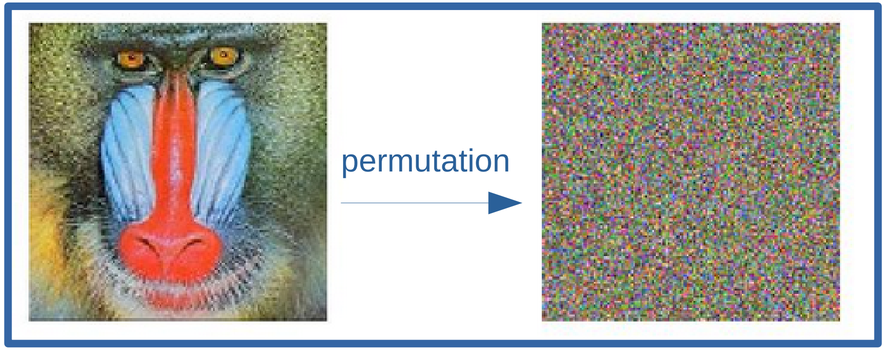
- *#parameters* does **NOT** represent *model complexity*
- *#parameters* does **NOT** upperbound E_{gen}
- Classic views to (*Capacity* ↔ E_{gen}) are **NOT** sufficient [8-12]

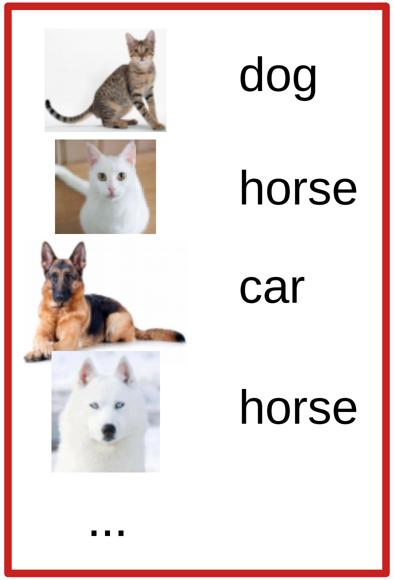
Why DNNs generalise well?

- Classic views … *#P* & *#N* ... insufficient!
- DNNs generalise well because of ...
  - Optimisation?
  - Regularisation?
  - …

Randomisation Test

- Training data: $\{x_i, y_i\}$, $i=1, 2, \ldots, N$
- Break the (x_i,y_i) relationship by randomising $\mathbf{x_i}$ or $\mathbf{y_i}$





[8]

Randomisation Test

- Training data: $\{x_i, y_i\}$, $i=1, 2, \ldots, N$
- Break the (x_i, y_i) relationship by randomising x_i or y_i

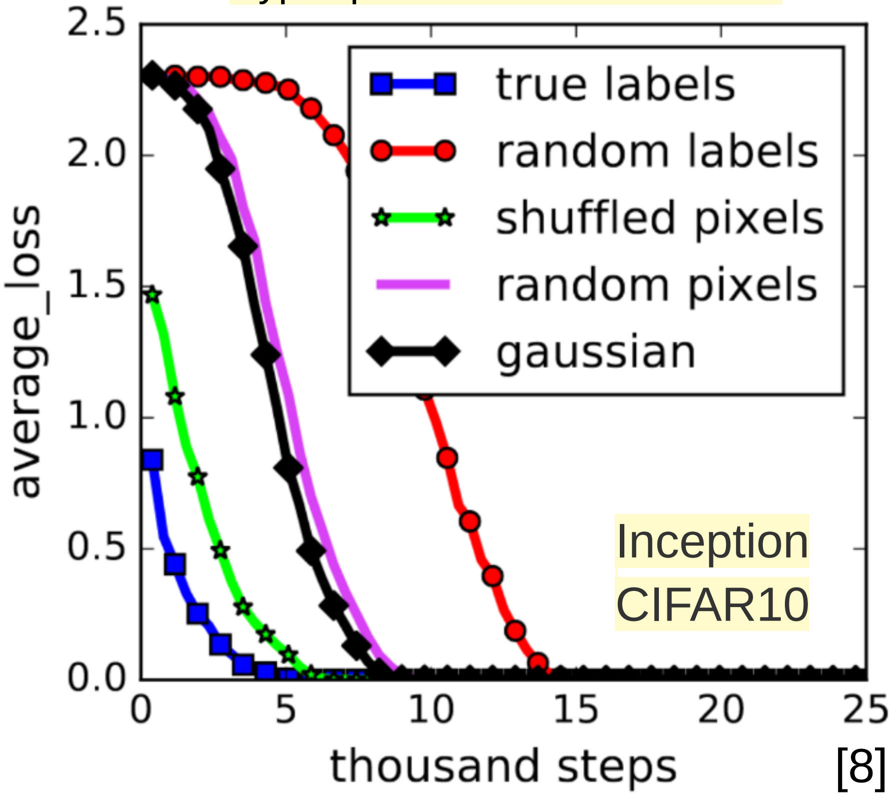
- Learning/Generalisation is IMPOSSIBLE!
- How about optimisation? (IM)Possible?

Randomisation Test – Results (1)

DNN *shatters* *(E_{train}=0)* **training data**, even with random data/labels.

This is ***fitting*** ...
agnostic to quality of **learning**!

Hyper-parameters are identical



[8]

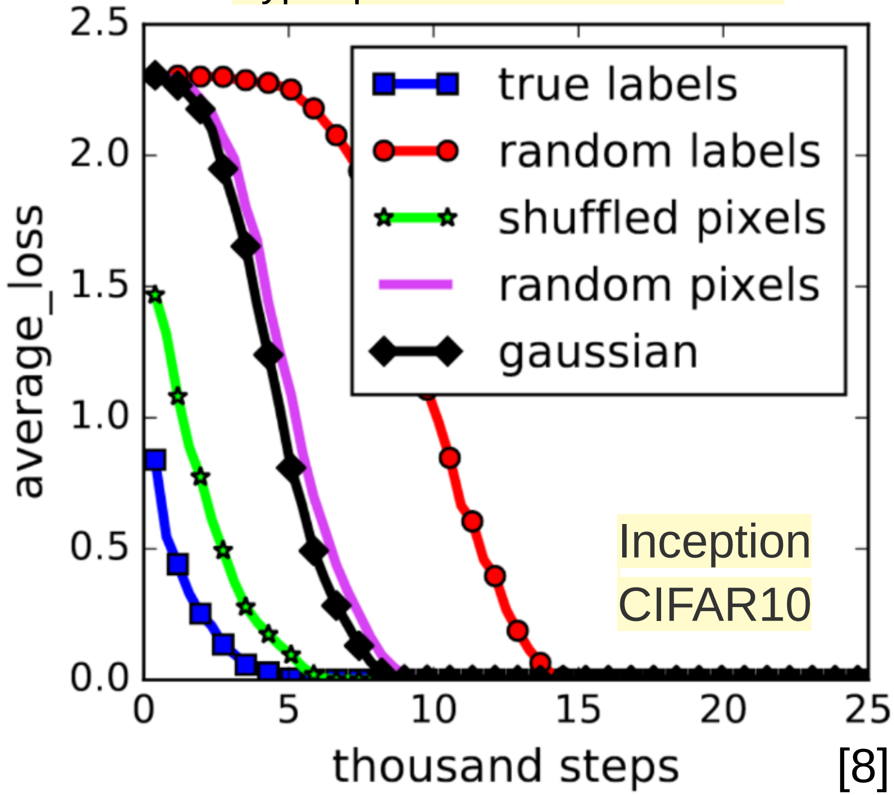
Randomisation Test – Results (2)

$E_{gen} = E_{test} - E_{train} = <15, 90, 90, 90, 90$

E_{gen} is very different even when *N*, *P* and *architecture* are the same!

$$E_{gen}\Big|_{E_{train}=0} \leq O(\frac{VCdim}{N})$$

Hyper-parameters are identical



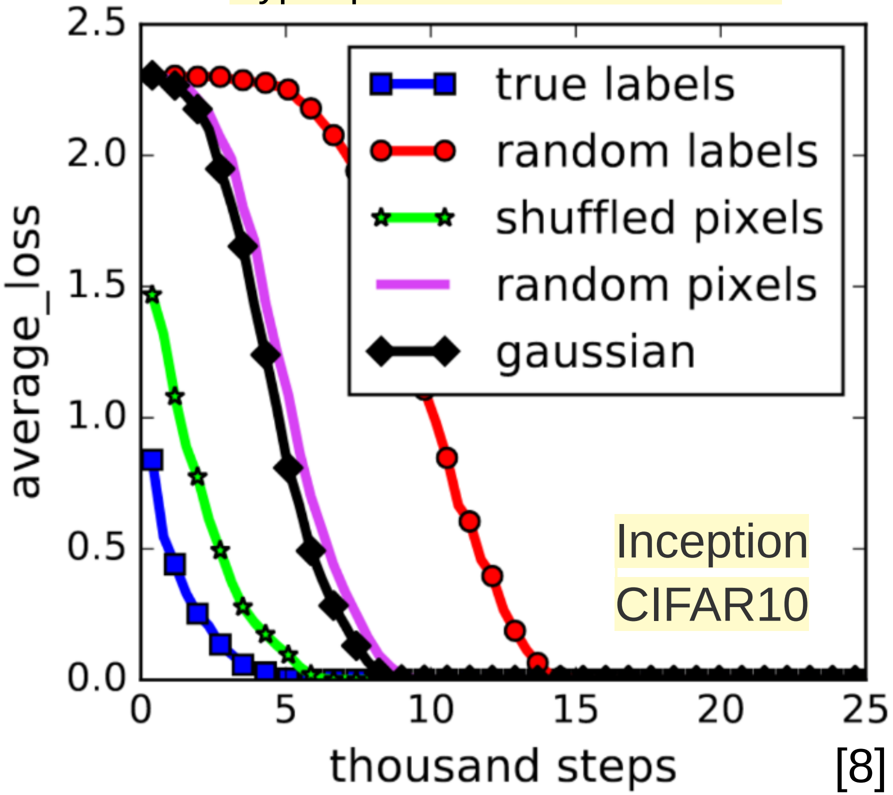
[8]

MVIP2022
Shahid Chamran University of Ahvaz

Randomisation Test – Results (3)

Optimisation remains easy, …
even when learning is impossible!
… Just slows down.

Hyper-parameters are identical



[8]

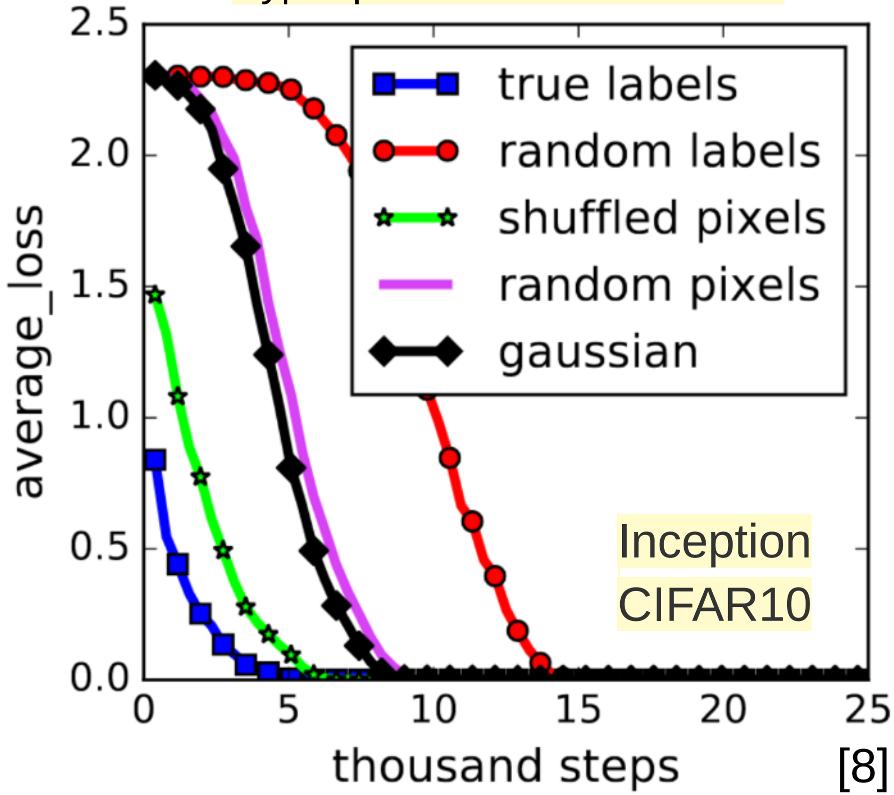
Randomisation Test – Results (3)

Optimisation remains easy, …
even when learning is impossible!
… Just slows down.

Optimisation ↔ Fitting [YES]

Optimisation ↔ Learning [NO]

Hyper-parameters are identical



Inception
CIFAR10

[8]

Local vs Global Optima ...

- Critical points … local/global min/max or saddle
  - Positive/negative/in-definite Hessian → min/max/saddle

Local vs Global Optima ...

- Critical points … local/global min/max or saddle
  - Positive/negative/in-definite Hessian → min/max/saddle
- In high dimensional spaces …
  - Most of the critical points are **saddle** point [13]
  - **Local** minima are likely to be as good as **global** minima [14,15]

Local vs Global Optima ...

- Critical points … local/global min/max or saddle
  - Positive/negative/in-definite Hessian → min/max/saddle
- In high dimensional spaces …
  - Most of the critical points are **saddle** point [13]
  - **Local** minima are likely to be as good as **global** minima [14,15]
    - ✓ *"… struggling to find the global minimum … is not useful in practice and may lead to overfitting … [15]"*

Explicit **Reg**ularisation Effect

Max Performance Improvement ...
- By **Reg.: +3.56** (85.75 → 89.31)
- By **Arch.: +35.24** (50.51 → 85.75)

CIFAR-10 — **W/ Reg.** — **W/O Reg.**

model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
(fitting random labels)		no	no	100.0	9.78
Inception w/o BatchNorm	1,649,402	no	yes	100.0	83.00
		no	no	100.0	82.00
(fitting random labels)		no	no	100.0	10.12
Alexnet	1,387,786	yes	yes	99.90	81.22
		yes	no	99.82	79.66
		no	yes	100.0	77.36
		no	no	100.0	76.07
(fitting random labels)		no	no	99.82	9.86
MLP 3x512	1,735,178	no	yes	100.0	53.35
		no	no	100.0	52.39
(fitting random labels)		no	no	100.0	10.48
MLP 1x512	1,209,866	no	yes	99.80	50.39
		no	no	100.0	50.51
(fitting random labels)		no	no	99.34	10.61

[8]

Explicit **Regularisation** Effect

Max Performance Improvement ...
- By **Reg.: +3.56** (85.75 → 89.31)
- By **Arch.: +35.24** (50.51 → 85.75)

Regularisation helps …
incrementally **NOT** fundamentally

Architecture plays a critical role

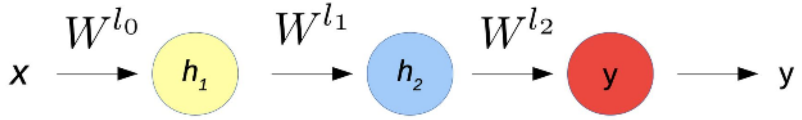
CIFAR-10 — **W/ Reg.** — **W/O Reg.**

model	# params	random crop	weight decay	train accuracy	test accuracy
Inception	1,649,402	yes	yes	100.0	89.05
		yes	no	100.0	89.31
		no	yes	100.0	86.03
		no	no	100.0	85.75
(fitting random labels)		no	no	100.0	9.78
Inception w/o BatchNorm	1,649,402	no	yes	100.0	83.00
		no	no	100.0	82.00
(fitting random labels)		no	no	100.0	10.12
Alexnet	1,387,786	yes	yes	99.90	81.22
		yes	no	99.82	79.66
		no	yes	100.0	77.36
		no	no	100.0	76.07
(fitting random labels)		no	no	99.82	9.86
MLP 3x512	1,735,178	no	yes	100.0	53.35
		no	no	100.0	52.39
(fitting random labels)		no	no	100.0	10.48
MLP 1x512	1,209,866	no	yes	99.80	50.39
		no	no	100.0	50.51
(fitting random labels)		no	no	99.34	10.61

[8]

Implicit Regularisation in SGD ...

Back Propagation



$$\color{green}{W_{jk}^{(i)}} = \color{red}{W_{jk}^{(i-1)}} - \eta\, o_j\, \delta_k$$

$$\delta_k = \begin{cases} (o_k - t_k)\, o_k\, (1 - o_k) & ,\ \text{if } k \in y \\ \left(\sum_{l \in L} \delta_l\, \color{green}{W_{kl}}\right) o_k\, (1 - o_k) & ,\ \text{if } k \in h_i \end{cases}$$

$$W^{l_2} = f(E)$$

$$W^{l_1} = f(E, W^{l_2})$$

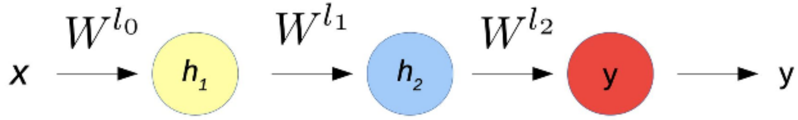
$$W^{l_0} = f(E, W^{l_2}, W^{l_1})$$

BP

Implicit Regularisation in SGD ...

Implicit regularisation ...
weights are **tied** together ...

Back Propagation



$$W_{jk}^{(i)} = W_{jk}^{(i-1)} - \eta\, o_j\, \delta_k$$

$$\delta_k = \begin{cases} (o_k - t_k)\, o_k\, (1 - o_k) & ,\ \text{if } k \in y \\ (\sum_{l \in L} \delta_l\, W_{kl})\, o_k\, (1 - o_k) & ,\ \text{if } k \in h_i \end{cases}$$

$$W^{l_2} = f(E)$$

$$W^{l_1} = f(E, W^{l_2})$$

$$W^{l_0} = f(E, W^{l_2}, W^{l_1})$$

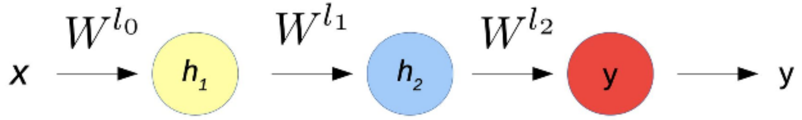
BP

Implicit Regularisation in SGD ...

Implicit regularisation ...
weights are **tied** together ...

Capacity ≡ #Params_effective
#Params_effective **<<** #Params

Back Propagation



$$W_{jk}^{(i)} = W_{jk}^{(i-1)} - \eta \, o_j \, \delta_k$$

$$\delta_k = \begin{cases} (o_k - t_k)\, o_k\, (1 - o_k) & , \text{ if } k \in y \\ (\sum_{l \in L} \delta_l \, W_{kl})\, o_k\, (1 - o_k) & , \text{ if } k \in h_i \end{cases}$$

$$W^{l_2} = f(E)$$

$$W^{l_1} = f(E, W^{l_2})$$

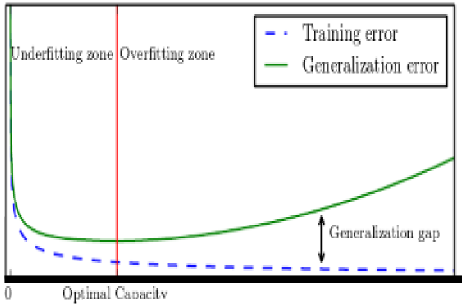
$$W^{l_0} = f(E, W^{l_2}, W^{l_1})$$

← BP

Implicit Regularisation in SGD ...

Implicit regularisation ...
weights are **tied** together ...

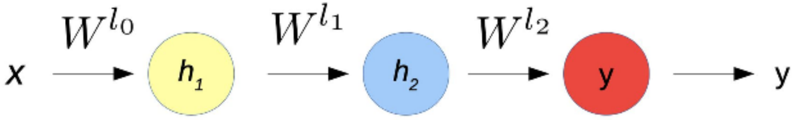
Error



Capacity
(model complexity)

DNNs

Back Propagation



$$W_{jk}^{(i)} = W_{jk}^{(i-1)} - \eta \, o_j \, \delta_k$$

$$\delta_k = \begin{cases} (o_k - t_k) \, o_k \, (1 - o_k) & , \text{ if } k \in y \\ (\sum_{l \in L} \delta_l \, W_{kl}) \, o_k \, (1 - o_k) & , \text{ if } k \in h_i \end{cases}$$

$$W^{l_2} = f(E)$$

$$W^{l_1} = f(E, W^{l_2})$$

$$W^{l_0} = f(E, W^{l_2}, W^{l_1})$$

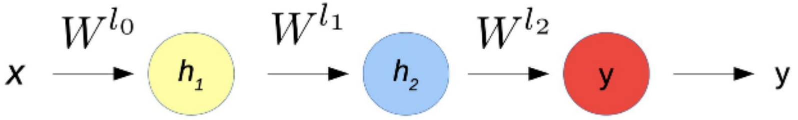
BP

Implicit Regularisation in SGD ...

Implicit regularisation …
weights are **tied** together ...

… is responsible for good *generalisation* of the DNNs.

Back Propagation



$$W_{jk}^{(i)} = W_{jk}^{(i-1)} - \eta \, o_j \, \delta_k$$

$$\delta_k = \begin{cases} (o_k - t_k)\, o_k\, (1 - o_k) & , \text{ if } k \in y \\ (\sum_{l \in L} \delta_l \, W_{kl})\, o_k\, (1 - o_k) & , \text{ if } k \in h_i \end{cases}$$

$$W^{l_2} = f(E)$$

$$W^{l_1} = f(E, W^{l_2})$$

$$W^{l_0} = f(E, W^{l_2}, W^{l_1})$$

BP

MVIP2022

Shahid Chamran University of Ahvaz

Conclusion (Part II)

- Classic wisdom about generalisation is insufficient
- #Parameters does NOT represent model complexity
- Optimisation remains easy, even when learning is hard
- Explicit regularisation helps, incrementally NOT fundamentally
- Why do DNNs generalise well?
  - Implicit regularisation in SGD and …

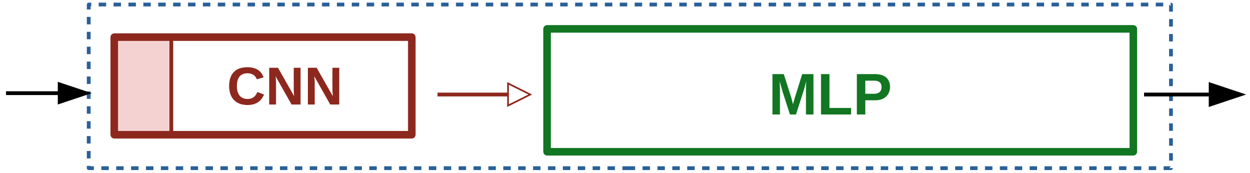
Outlines (Part III)

- Information Bottleneck
- Over-parameterisation and Generalisation
- Interpretation/Visualisation of Filters/Activations

MVIP2022
Shahid Chamran University of Ahvaz

We will investigate ...

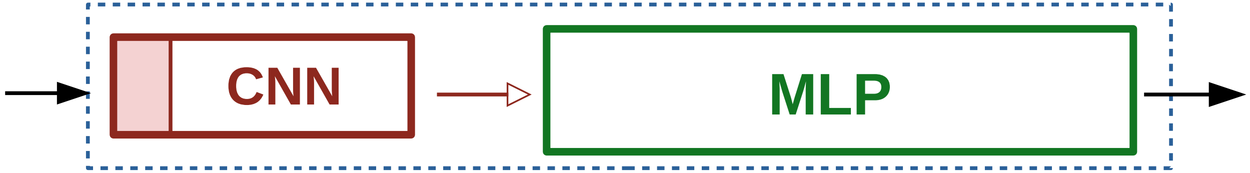
- Seriousness of gradient vanishing in low layers [16]
- Linear separability in high layers [17]



We will investigate ...

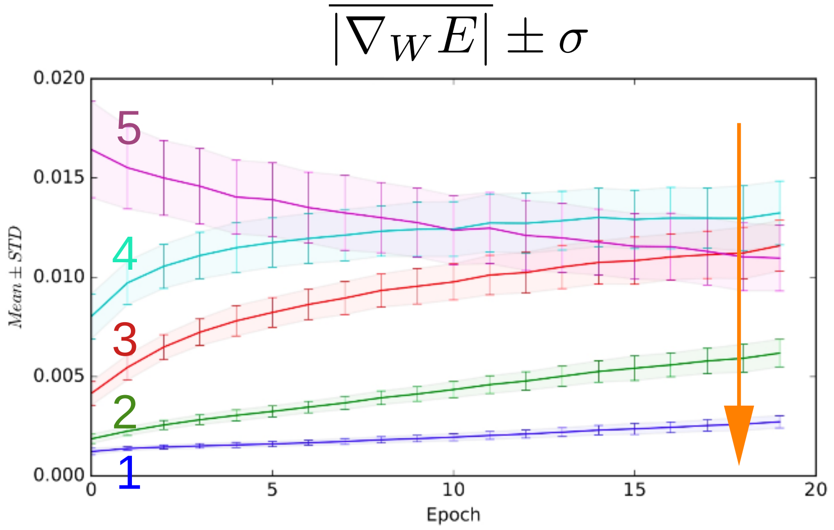
- Seriousness of gradient vanishing in low layers [16]

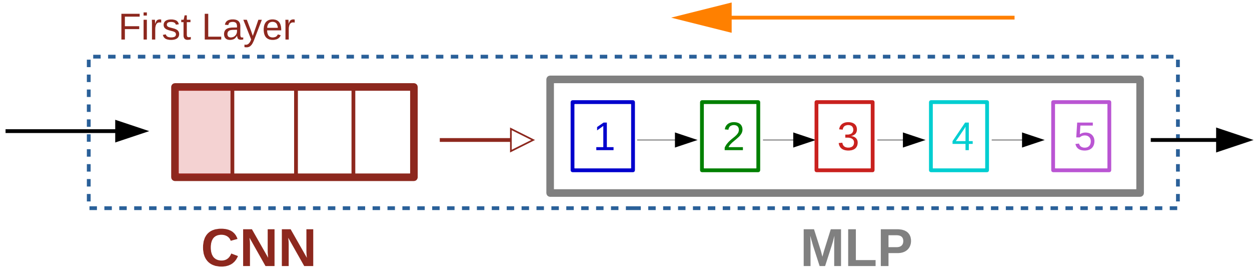
- Linear separability in high layers [17]



Seriousness of Gradient Vanishing

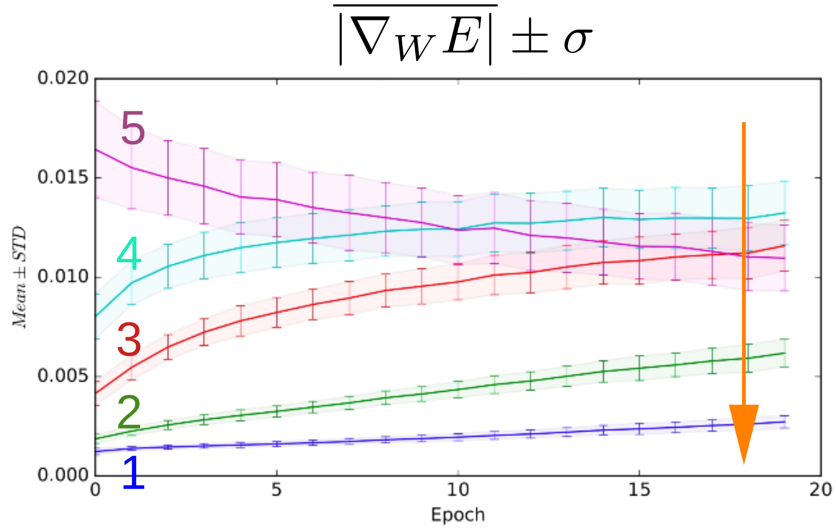
gradient vanishing …

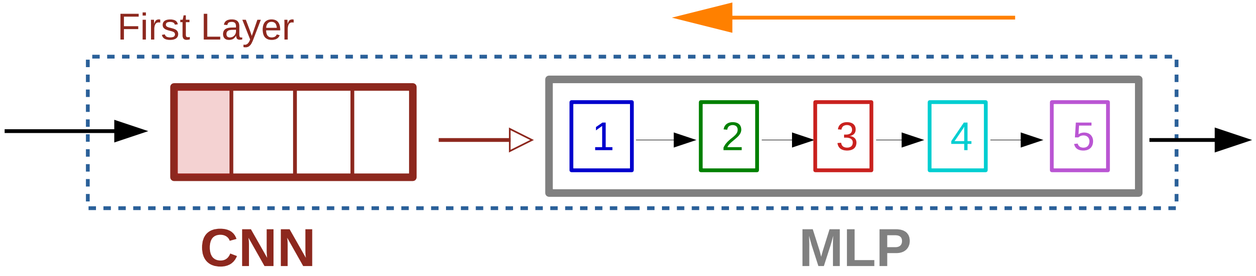




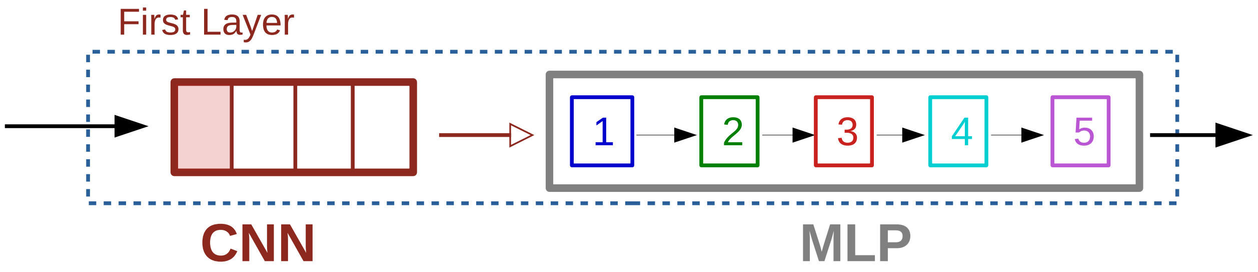
Seriousness of Gradient Vanishing

In light of gradient vanishing …
How optimal the first layer is?





How to investigate it?



* Error or accuracy reflect DNN’s collective behaviour

* *Layer-dependent* metric is needed ...

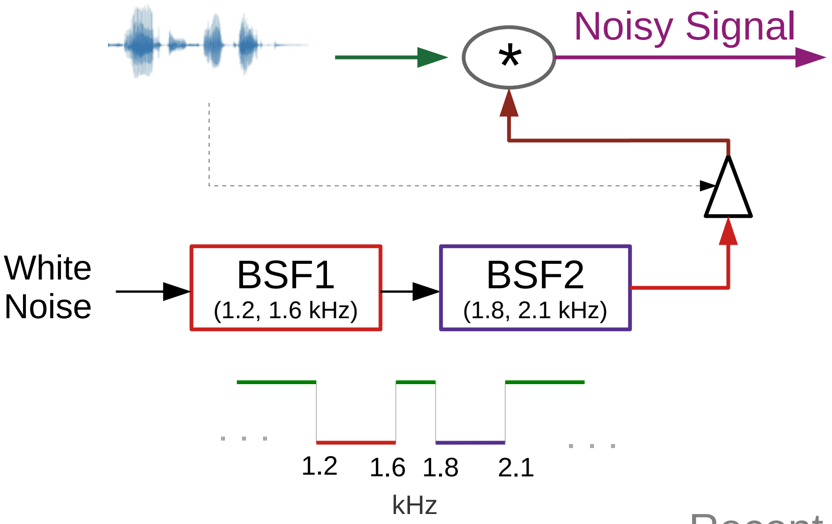
The proposed task ...

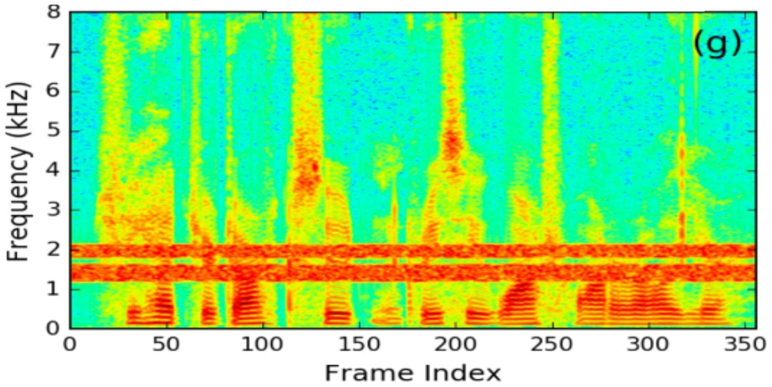
- Task: Phone recognition (TIMIT) using raw waveform

This is not strictly correct ...
Y is the state-clusterd triphones.

The proposed task ...

- Task: Phone recognition (TIMIT) using raw waveform
- How: add noise to training data ...



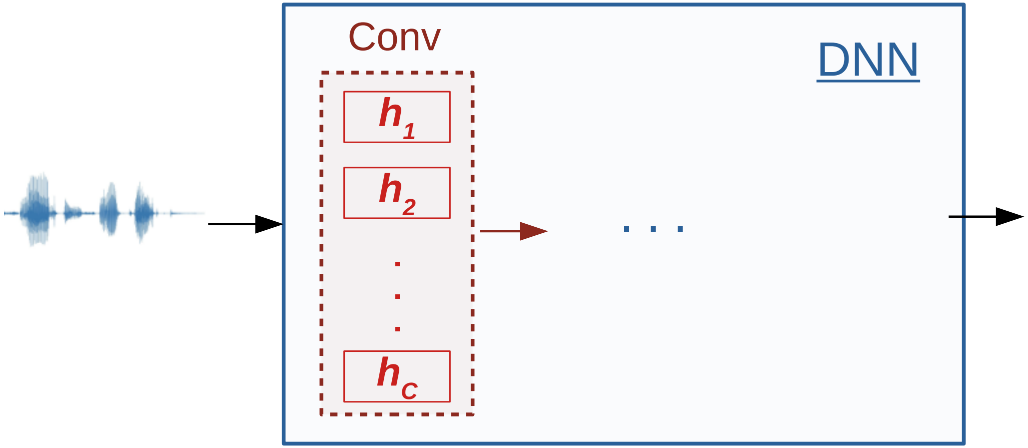


Gradient Vanishing Seriousness

- Task: Phone recognition (TIMIT) using raw waveform
- How: add noise to training data
- Metric: Average Frequency Response (AFR)

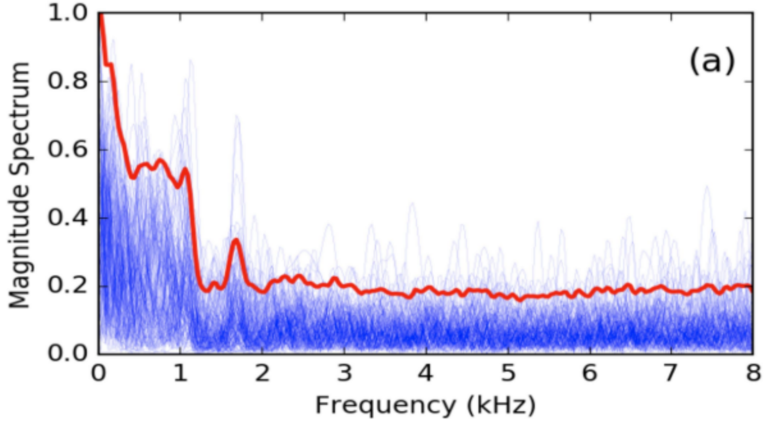
$$\text{AFR} = \frac{1}{C}\sum_{c=1}^{C} |H_c(\omega)|$$

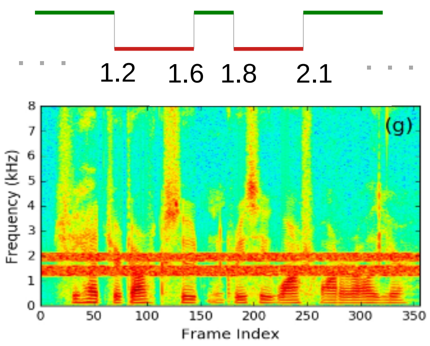
h: impulse response
H: frequency response
C: #channels



AFR Dynamics (1)

Epoch 1

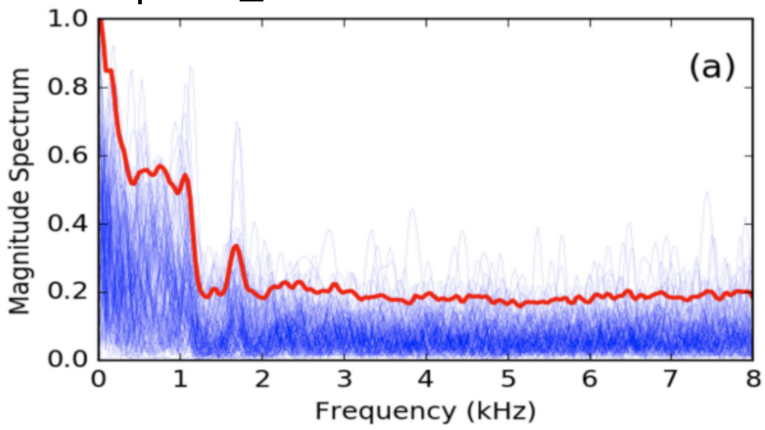




[16]

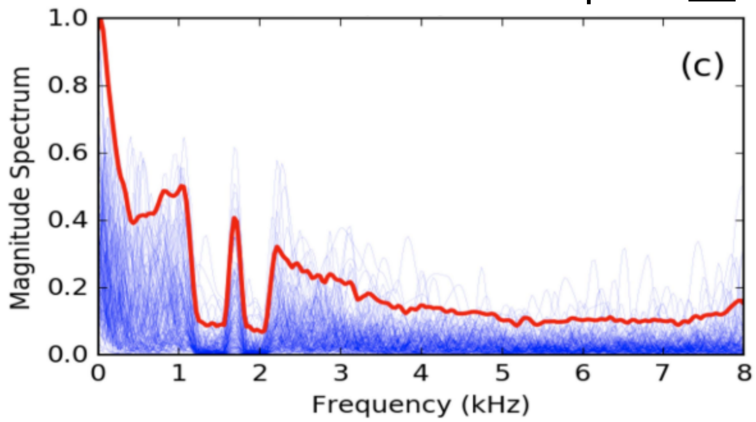
AFR Dynamics (1)

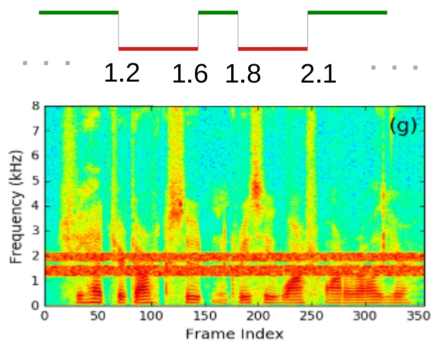
Epoch 1



. . .

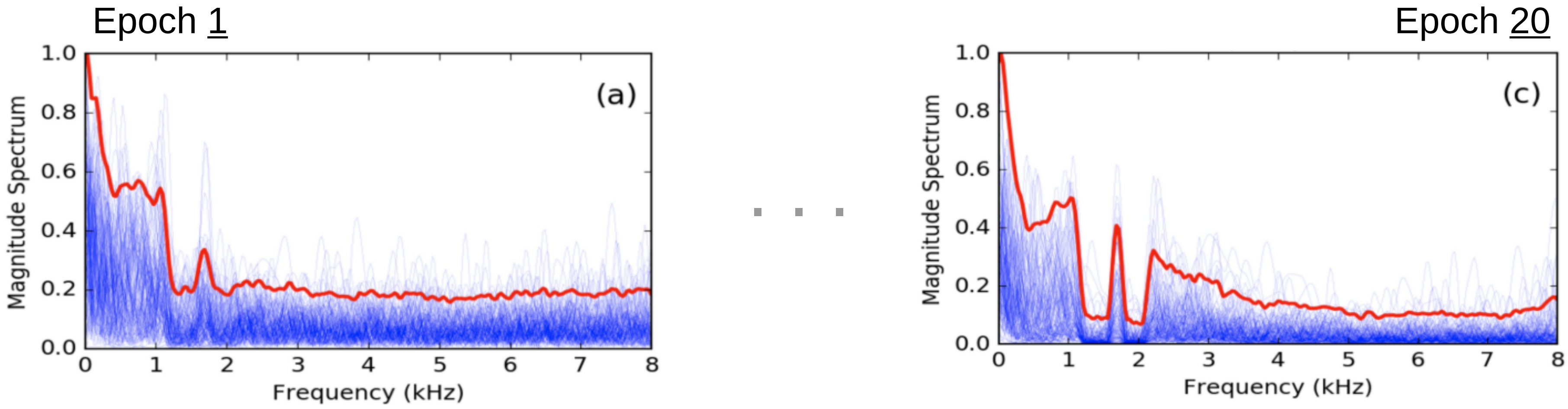
Epoch 20



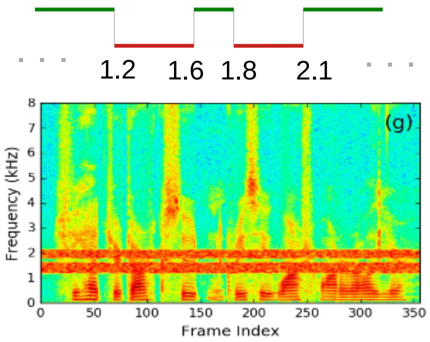


[16]

AFR Dynamics (2)



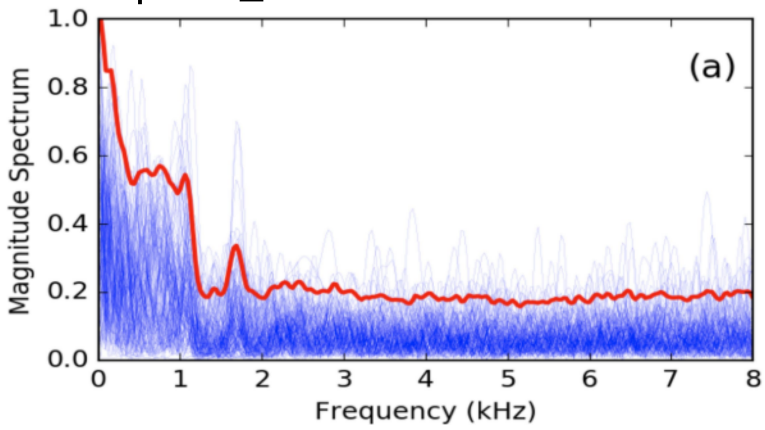
Using phone labels, the model finds the noisy sub-bands and filters them out.



[16]

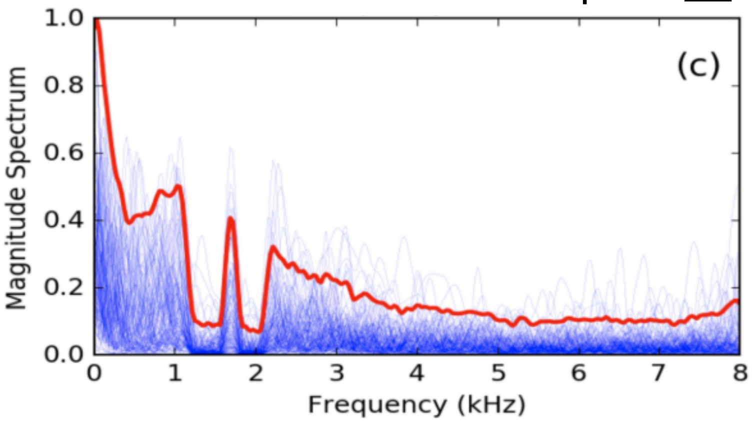
AFR Dynamics (2)

Epoch 1

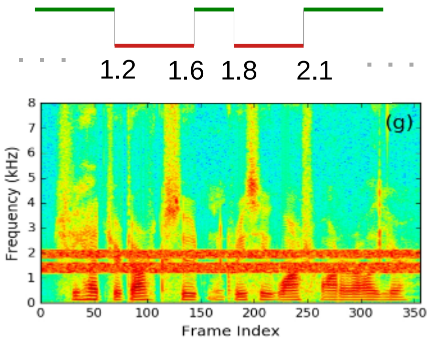


. . .

Epoch 20



Using phone labels, the model finds the noisy sub-bands and filters them out.

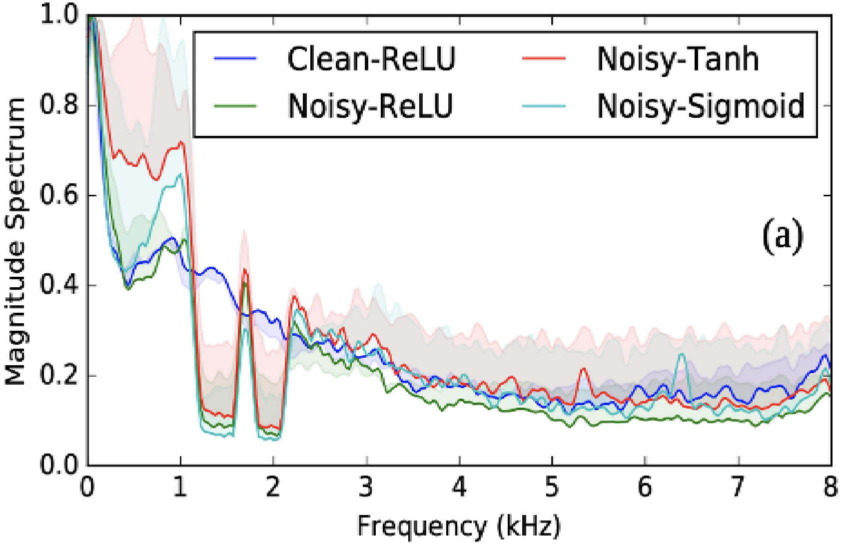


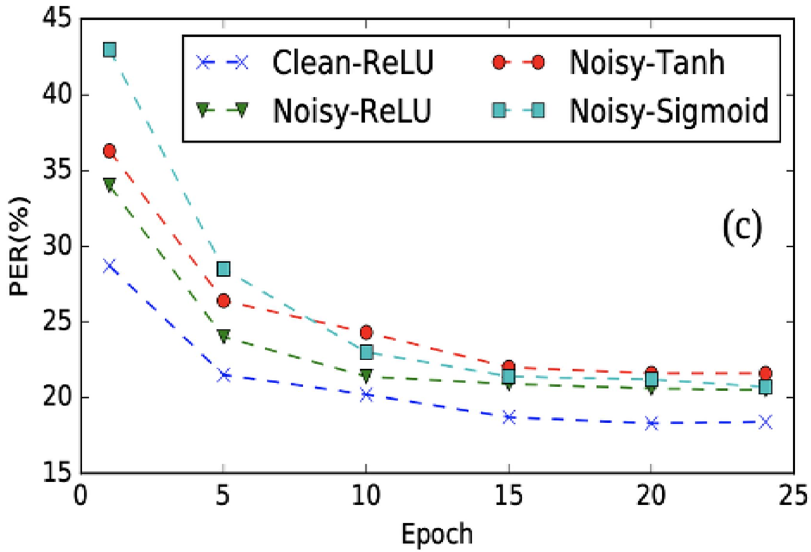
Gradient vanishing is NOT a serious problem ...

[16]

MVIP2022

Effect of Activation Function





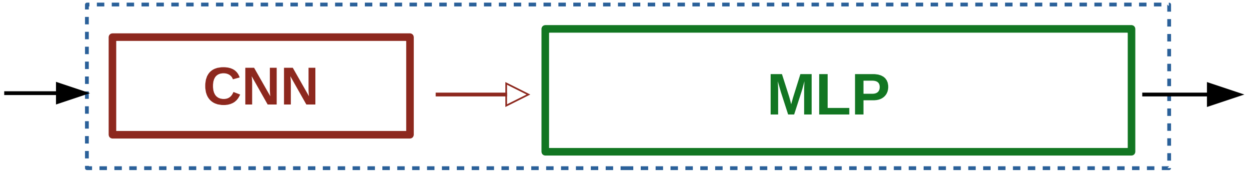
[16]

* … Sigmoid and Tanh … Noisy sub-bands successfully found ...

* Gradient vanishing is NOT a serious problem <u>in a reasonable setup</u>!

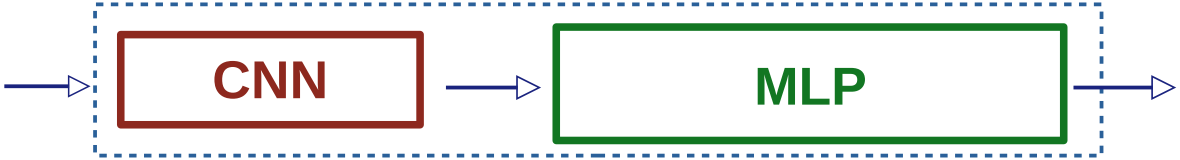
We will investigate ...

- Seriousness of gradient vanishing in low layers [16]
- Linear separability in high layers [17]



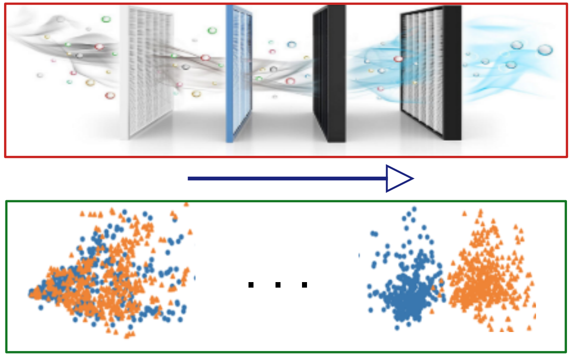
Towards output layer ...

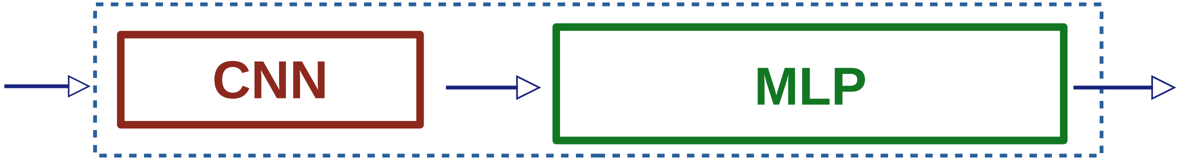
- DNN should ...
  - Filter out irrelevant information



Towards output layer ...

- DNN should ...
  - Filter out irrelevant information
  - Enhance linear separability
    - Softmax is a linear classifier

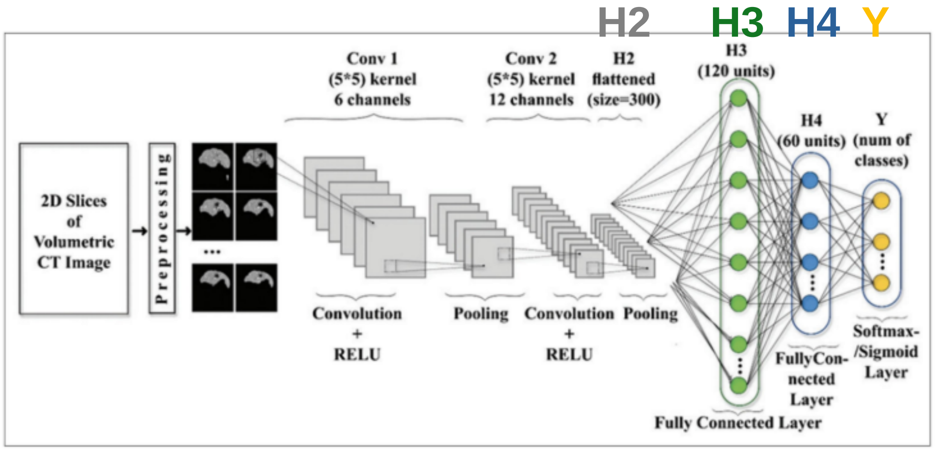




Investigating the Linear Separability ...

- **Task:** A binary classification (Question F, ImageCLEF2015)
- **How:** Dump activations → Dim. reduction to 2D (t-SNE, PCA, ...) →

  → Monitor linear separability across layers/epochs

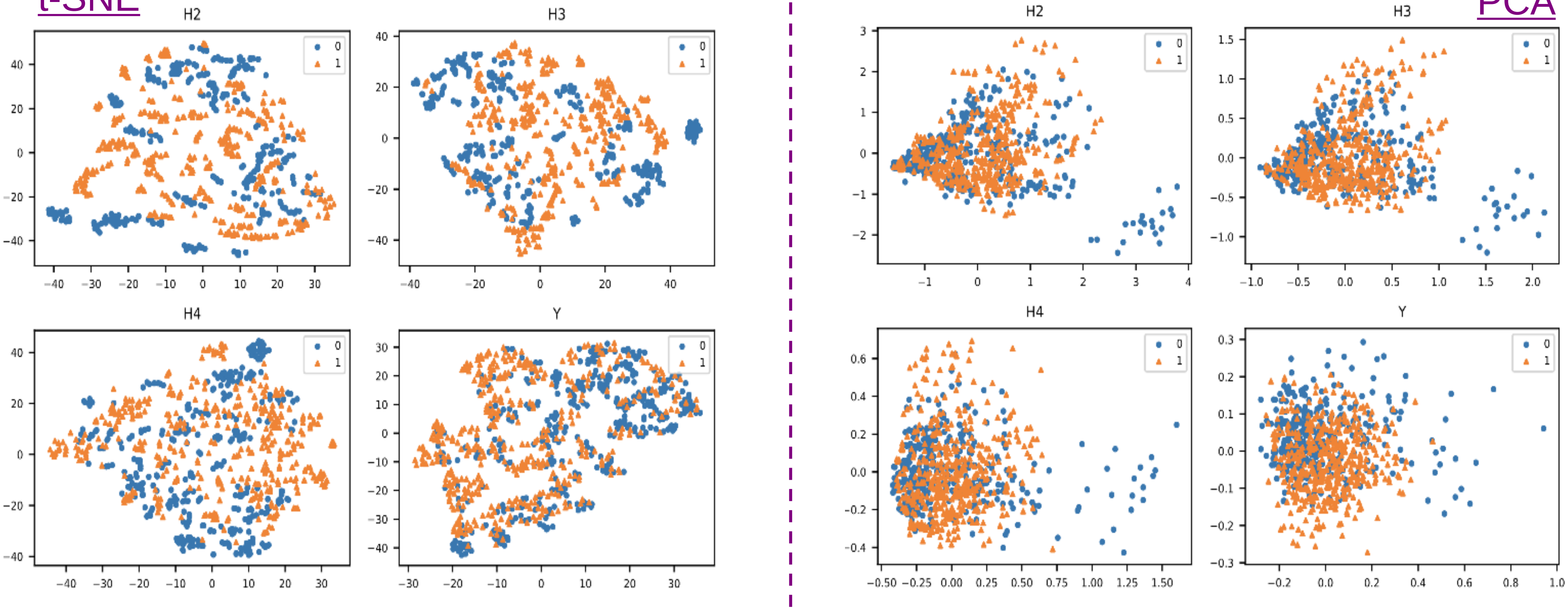


[17]

Epoch: 1

t-SNE

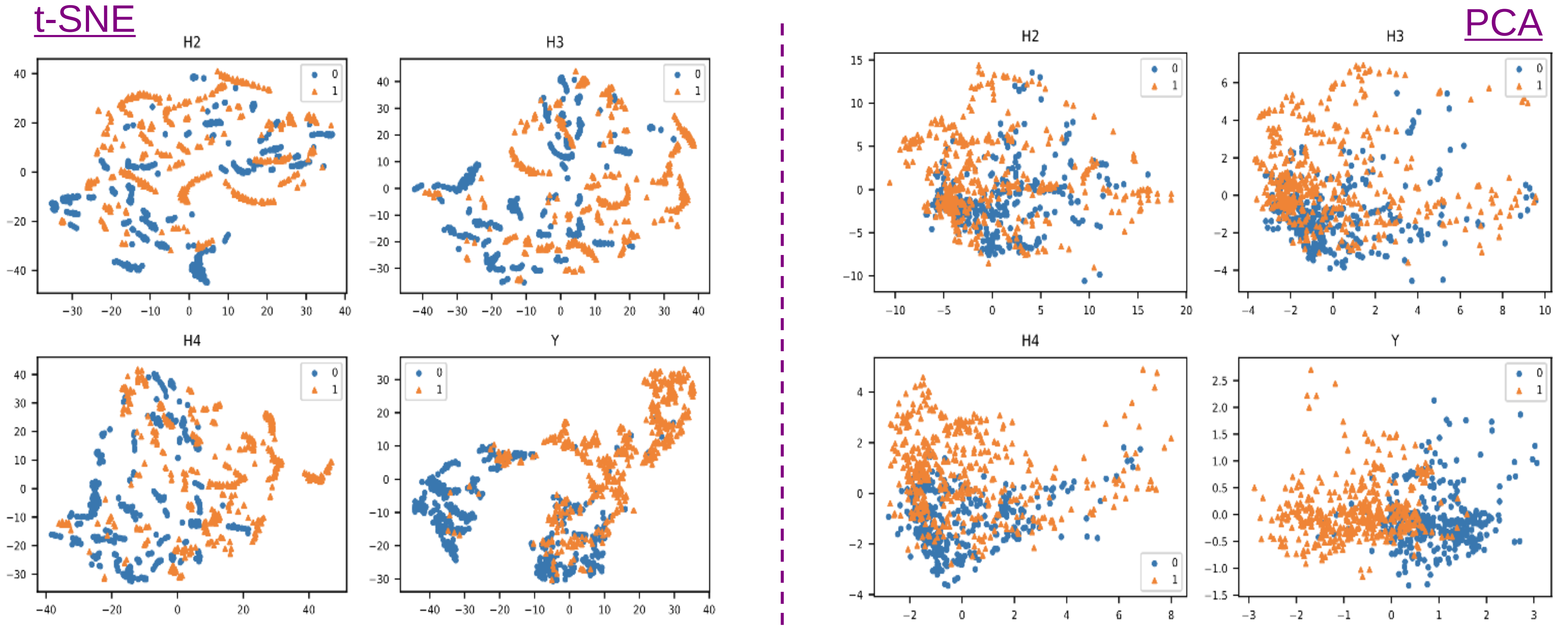
PCA



X → CNN … H2 → H3 → H4 → Y

[17]

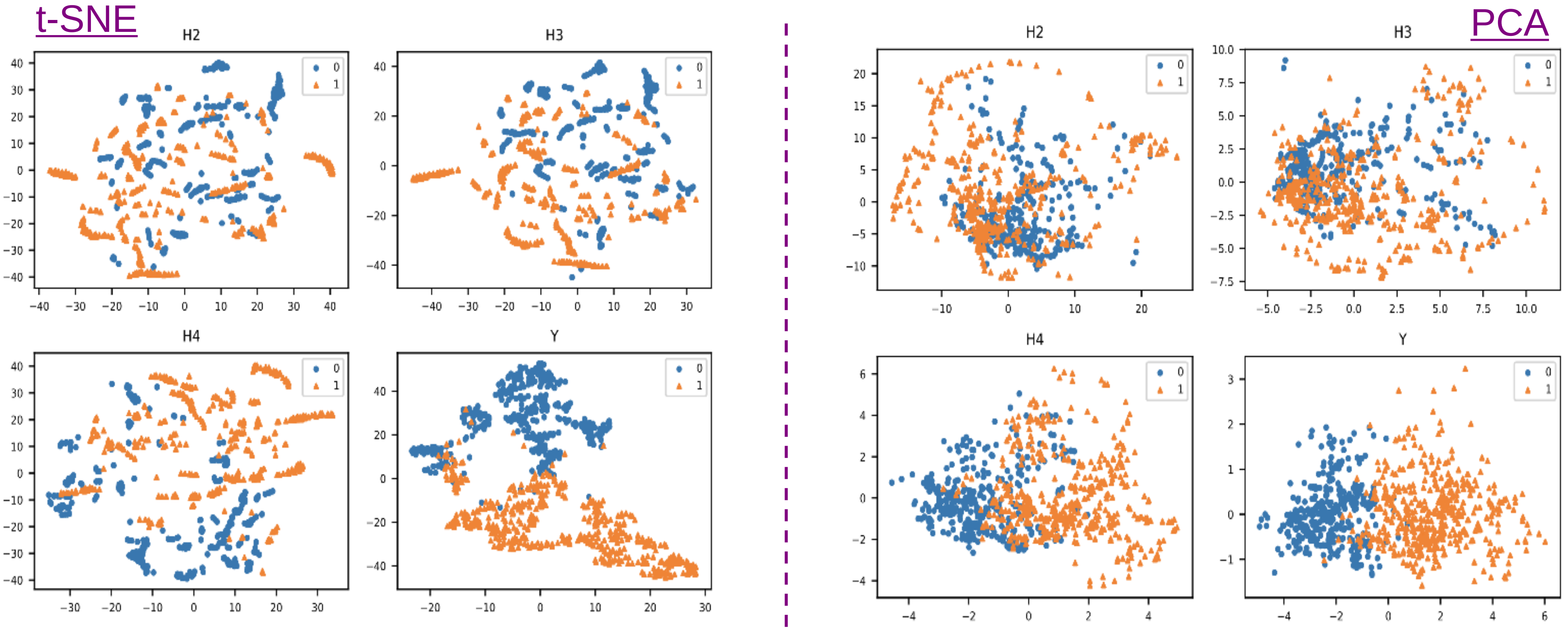
Epoch: 5



X → CNN … H2 → H3 → H4 → Y

[17]

Epoch: 10



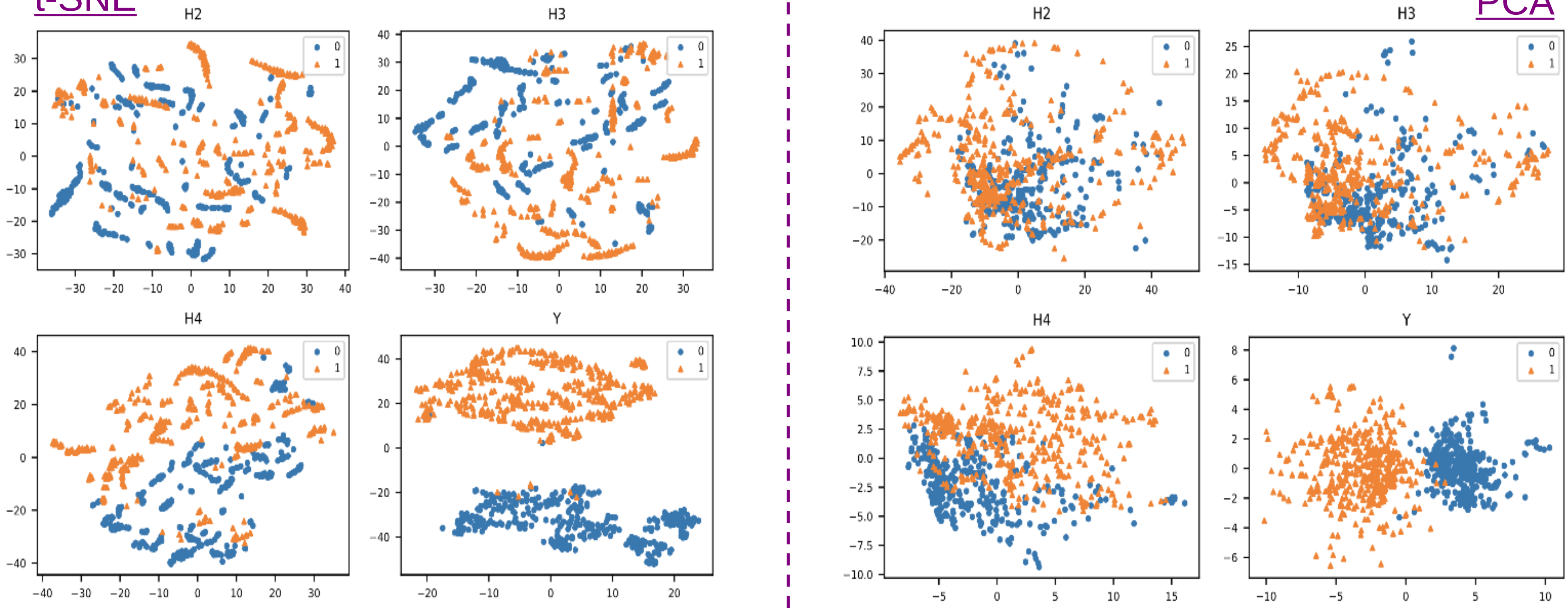
X → CNN … H2 → H3 → H4 → Y

[17]

Epoch: 15

t-SNE

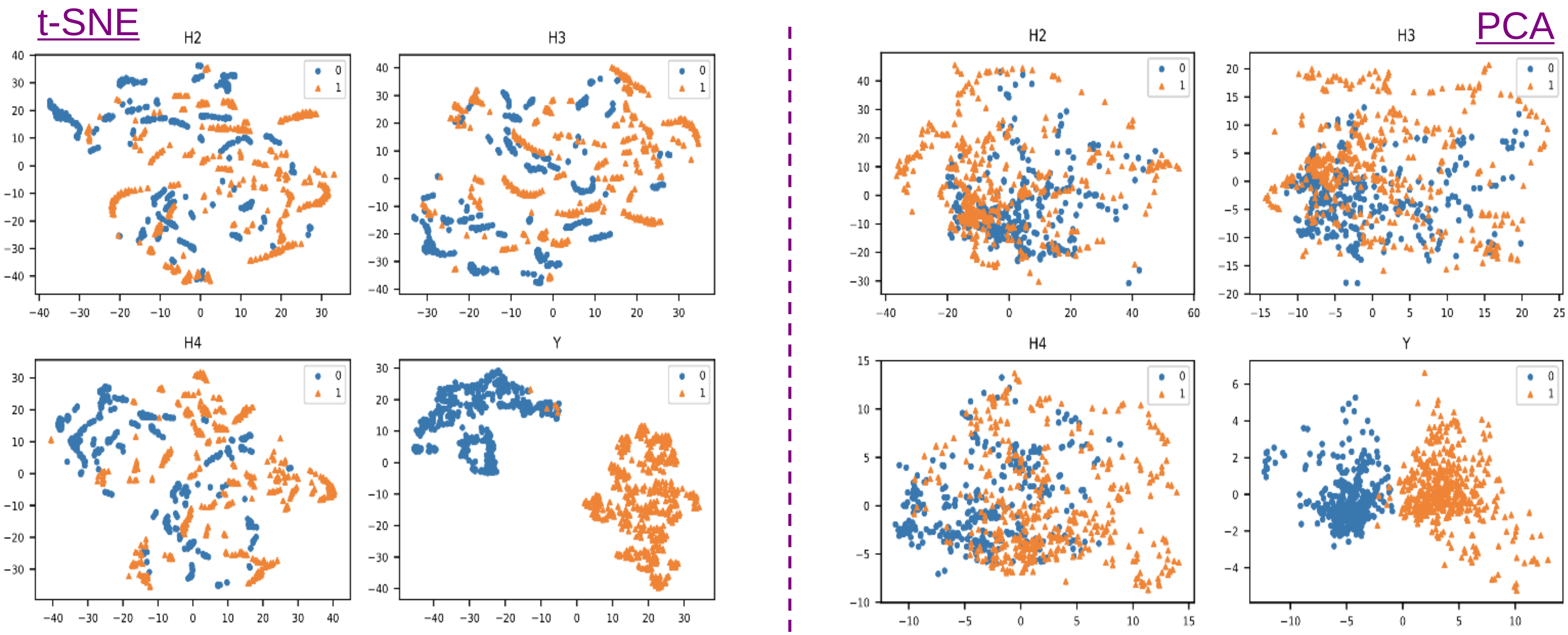
PCA



X → CNN … H2 → H3 → H4 → Y

[17]

Epoch: 20



X → CNN … H2 → H3 → H4 → Y

[17]

Conclusion (Part III)

- We studied/visualised the …
  - Gradient vanishing seriousness
  - Linear separability across layers/epochs

- Providing interpretation/visualisation make the reviewer/readers happy :-), embed them into your work!

That's It!

- Thank you for Your Attention!
- Q&A

- References ↓

References (Part I)

[1] R. Shwartz-Ziv and N. Tishby, “Opening the black box of deep neural networks via information,” *CoRR*, vol. abs/1703.00810, 2017.

[2] N. Tishby, F. C. Pereira, and W. Bialek, “The information bottleneck method,” in *Proc. of the 37th Annual Allerton Conference on Communication, Control and Computing*, 1999, pp. 368–377.

[3] A. M. Saxe, Y. Bansal, J. Dapello, M. Advani, A. Kolchinsky, B. D. Tracey, and D. D. Cox, “On the information bottleneck theory of deep learning.” in *ICLR*, 2018.

[4] I. Chelombiev, C. J. Houghton, and C. O’Donnell, “Adaptive estimators show information compression in deep neural networks,” in *ICLR*, 2019.

[5] J.-H. Jacobsen, A. W. M. Smeulders, and E. Oyallon, “i-RevNet: Deep invertible networks,” in *ICLR*, 2018.

[6] M. Noshad, Y. Zeng, and A. O. Hero, “Scalable mutual information estimation using dependencegraphs,” in *ICASSP*, 2019.

[7] T. M. Cover and J. A. Thomas, *Elements of information theory*, 2nd ed. Wiley-Interscience, 2006.

MVIP2022

Shahid Chamran University of Ahvaz

References (Part II)

[8] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, “Understanding deep learning requires rethinking generalization,” In *ICLR*, 2017.

[9] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, “Understanding deep learning (still) requires rethinking generalization,” *Commun. ACM*, vol. 64, no. 3, p. 107–115, 2021.

[10] C. Zhang, S. Bengio, M. Hardt, M. C. Mozer, and Y. Singer, “Identity crisis: Memorization and generalization under extreme overparameterization,” In *ICLR*, 2020.

[11] B. Neyshabur, S. Bhojanapalli, D. Mcallester, and N. Srebro, “Exploring generalization in deep learning,” In *NIPS*, 2017.

[12] Y. Jiang, B. Neyshabur, H. Mobahi, D. Krishnan, and S. Bengio, “Fantastic generalization measures and where to find them,” In *ICLR*, 2020.

[13] Y. N. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio, “Identifying andattacking the saddle point problem in high-dimensional non-convex optimization,” in *NIPS*, 2014.

[14] S. Bhojanapalli, B. Neyshabur, and N. Srebro, “Global optimality of local search for low rankmatrix recovery,” in *NIPS*, 2016.

[15] A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, “The Loss Surfaces of Multilayer Networks,” in *PMLR*, 2015.

References – Part III

[16] E. Loweimi, P. Bell, and S. Renals, “On the robustness and training dynamics of raw waveform models,” *in Proc. INTERSPEECH*, 2020.

[17] S. Loveymi, M. H. Dezfoulian, and M. Mansoorizadeh, “Automatic generation of structured radiology reports for volumetric computed tomography images using question-specific deep feature extraction and learning,” in *Journal of medical signals and sensors*, 2016.

MVIP2022

Shahid Chamran University of Ahvaz