Information Bottleneck Analysis of Deep Neural Networks via Lossy Compression

Ivan Butakov, Alexander Tolmachev, Sofia Malanchuk, Anna Neopryatnaya, Alexey Frolov, Kirill Andreev

`i.butakov@skoltech.ru`;
Paper; GitHub page

ICLR

Skoltech
Skolkovo Institute of Science and Technology

Outline

- Background
  - Information theory in deep learning
  - Manifold hypothesis
- Mutual estimation via compression
  - Lossless compression
  - Lossy compression
- Experiments
  - Synthetic data
  - Information Bottleneck analysis of CNN classifier

Background

Information theory in deep learning

Applications:

- Generalization bounds
- Model selection, explainable AI
- Unsupervised representation learning
- Training objectives, regularization terms

Central information-theoretic quantities:

- *Differential entropy*: $h(X) = -\mathbb{E} \log p(X) \quad (p \text{ is PDF of } X)$
- *Mutual information* (MI): $I(X; Y) = h(X) - h(X \mid Y)$

Main difficulty:

- **Hard to apply to real-world high-dimensional data** $(dim \gtrsim 10 - 100)$

Manifold hypothesis

Real-world data can be assumed to lie on or close to some low-dimensional manifold.

Manifold hypothesis

Real-world data can be assumed to lie on or close to some low-dimensional manifold.

NN-based MI estimators can grasp the latent structure of data, thus showing relative practical success in dealing with the curse of dimensionality.

Mutual estimation via compression

Mutual estimation via compression

We propose to utilize the manifold hypothesis explicitly by compressing data.

Mutual estimation via compression

We propose to utilize the manifold hypothesis explicitly by compressing data.

- We show that an **explicit data compression** allows for comparable or even better estimation quality.

Mutual estimation via compression

We propose to utilize the manifold hypothesis explicitly by compressing data.

- We show that an **explicit data compression** allows for comparable or even better estimation quality.
- We derive **error bounds** for the general case of MI estimation via lossy compression.

Result: Mutual information is not alternated via the lossless compression:

Theorem 1

Let $\xi\colon \Omega \to \mathbb{R}^{n'}$ *be an absolutely continuous random vector, let* $g\colon \mathbb{R}^{n'} \to \mathbb{R}^{n}$ *be an injective piecewise-smooth mapping with Jacobian* J*, satisfying* $n \geq n'$ *and* $\det\left(J^T J\right) \neq 0$ *almost everywhere. Let* $h(\xi)$ *and* $h(\xi \mid \eta)$ *be defined. Then*

$$I(\xi;\eta) = I\left((g^{-1} \circ g)(\xi);\eta\right) = I\left(g(\xi);\eta\right)$$

Here g^{-1} should be interpreted as a compression mapping (encoder), $g(\xi)$ – as a high-dimensional random variable.

Result: Mutual information alternation under lossy compression can be bounded.

Theorem 2

Let X, Y, and Z be random variables such that $I(X;Y)$ and $I\left((X,Z);Y\right)$ are defined. Let f be a function of two arguments such that $I(f(X,Z);Y)$ is defined. If there exists a function g such that $X = g(f(X,Z))$, then the following chain of inequalities holds:

$$I(X;Y) \leq I\left(f(X,Z);Y\right) \leq I((X,Z);Y) \leq I\left(f(X,Z);Y\right) + h(Z) - h(Z \mid X,Y)$$

$f(X,Z)$ can be interpreted as compressed noisy data, X as denoised data, and g as a perfect denoising decoder. Z regulates the deviation from the manifold.

Lossy compression

Result: Mutual information alternation under lossy compression can be bounded.

Theorem 2

Let X, Y, and Z be random variables such that $I(X;Y)$ and $I((X,Z);Y)$ are defined. Let f be a function of two arguments such that $I(f(X,Z);Y)$ is defined. If there exists a function g such that $X = g(f(X,Z))$, then the following chain of inequalities holds:

$$I(X;Y) \leq I(f(X,Z);Y) \leq I((X,Z);Y) \leq I(f(X,Z);Y) + h(Z) - h(Z \mid X,Y)$$

$f(X,Z)$ can be interpreted as compressed noisy data, X as denoised data, and g as a perfect denoising decoder. Z regulates the deviation from the manifold.

Lossy compression

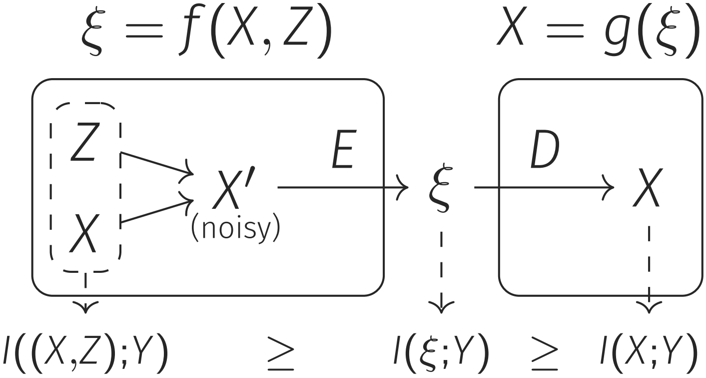


Figure 1: Conceptual scheme of Theorem 2 in application to the lossy compression via an autoencoder $A = D \circ E$.

Corollary 3

Let X, Y, Z, f, and g satisfy the conditions of the Theorem 2. Let random variables (X, Y) and Z be independent. Then $I(X;Y) = I((X,Z);Y) = I(f(X,Z);Y)$.

Corollary 4
Let X be a random vector of dimension n, let $Z \sim \mathcal{N}(0, \sigma^2 I_n)$, and X and Z be independent. Let E be a PCA-projector to a linear manifold of dimension n' with explained variances denoted by λ_i in the descending order. Then

$$0 \leq I(X + Z; Y) - I(E(X + Z); Y) \leq \frac{n - n'}{2} \log \left(1 + \frac{\lambda_{n'+1}}{\sigma^2} \right)$$

Experiments

- We leverage the **invariance of the mutual information under non-singular mappings** (see Theorem 1) to construct complex high-dimensional synthetic tests with known ground-truth mutual information.

- We leverage the **invariance of the mutual information under non-singular mappings** (see Theorem 1) to construct complex high-dimensional synthetic tests with known ground-truth mutual information.
- Multivariate Gaussian distribution with **known closed-form expression for mutual information** between subvectors is used as the base distribution.

Synthetic data

- We leverage the **invariance of the mutual information under non-singular mappings** (see Theorem 1) to construct complex high-dimensional synthetic tests with known ground-truth mutual information.
- Multivariate Gaussian distribution with **known closed-form expression for mutual information** between subvectors is used as the base distribution.
- Injective smooth mappings are used to construct **high-dimensional images** based on **low-dimensional representations**.

Synthetic data

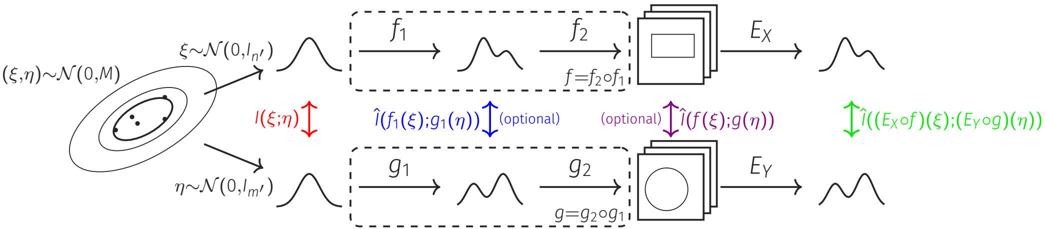


Figure 2: $f_1 : \mathbb{R}^{n'} \to \mathbb{R}^{n'}$ maps ξ to a structured latent representation of X (e.g., parameters of geometric shapes), and $f_2 : \mathbb{R}^{n'} \to \mathbb{R}^{n}$ maps latent representations to corresponding high-dimensional vectors (e.g., rasterized images of geometric shapes). The same for $g = g_2 \circ g_1$.

Synthetic data

Setup:

- Convolutional autoencoder is used to compress synthetic images.

Synthetic data

Setup:

- Convolutional autoencoder is used to compress synthetic images.
- Simplest non-parametric mutual information estimators (KDE and k-NN-based) are used to estimate MI between latent representations.

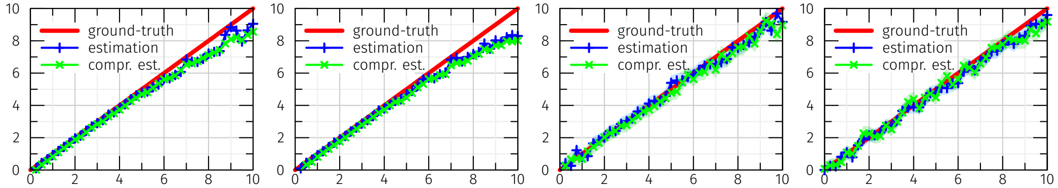
Synthetic data

Setup:

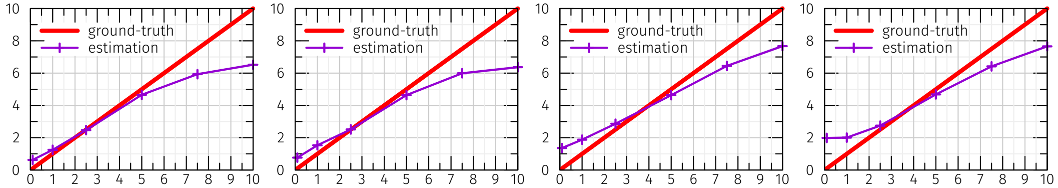
- Convolutional autoencoder is used to compress synthetic images.
- Simplest non-parametric mutual information estimators (KDE and k-NN-based) are used to estimate MI between latent representations.

Results:

- Autoencoder + k-NN-based Weighted Kozachenko-Leonenko (WKL) estimator shows decent quality, outperforming Mutual Information Neural Estimator (MINE).



(a) WKL, 16 × 16 px (b) WKL, 32 × 32 px (c) WKL, 16 × 16 px (d) WKL, 32 × 32 px



(e) MINE, 16 × 16 px (f) MINE, 32 × 32 px (g) MINE, 16 × 16 px (h) MINE, 32 × 32 px

Figure 3: Weighted Kozachenko-Leonenko and MINE, 16×16 and 32×32 images of Gaussians (columns 1-2, $n' = m' = 2$) and rectangles (columns 3-4, $n' = m' = 4$), $5 \cdot 10^3$ samples. Along x axes is $I(X;Y)$, along y axes is $\hat{I}(X;Y)$.

Information Bottleneck analysis of CNN classifier

Setup:

- A CNN classifier of the MNIST handwritten digits dataset is considered.

Setup:

- A CNN classifier of the MNIST handwritten digits dataset is considered.
- For every layer L_i, $I(X; L_i)$ and $I(L_i; Y)$ are tracked, where X denotes the input, Y denotes the target.

Information Bottleneck analysis of CNN classifier

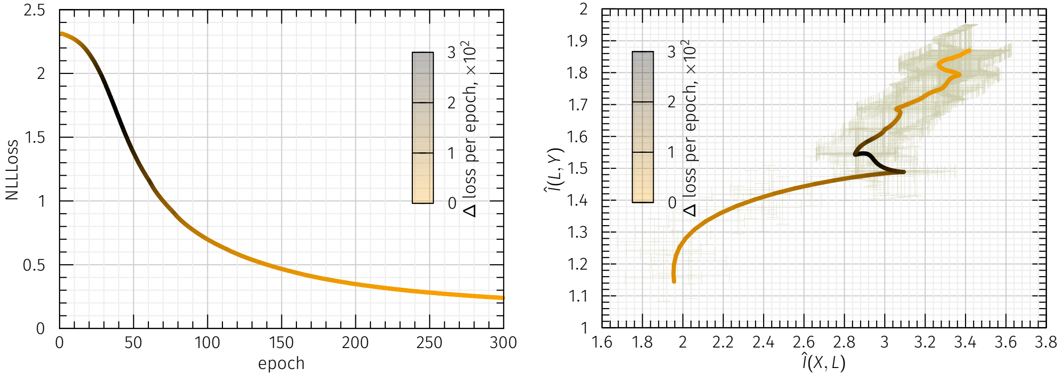
Setup:

- A CNN classifier of the MNIST handwritten digits dataset is considered.
- For every layer L_i, $I(X; L_i)$ and $I(L_i; Y)$ are tracked, where X denotes the input, Y denotes the target.
- After the training, for every L_i the value of $I(L_i; Y)$ is plotted against $I(X; L_i)$ (*information plane plot*).

Information Bottleneck analysis of CNN classifier

Setup:

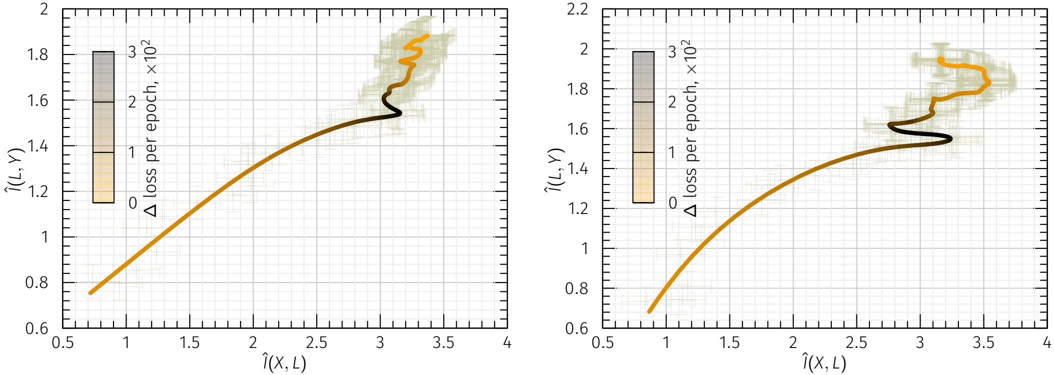
- A CNN classifier of the MNIST handwritten digits dataset is considered.
- For every layer L_i, $I(X; L_i)$ and $I(L_i; Y)$ are tracked, where X denotes the input, Y denotes the target.
- After the training, for every L_i the value of $I(L_i; Y)$ is plotted against $I(X; L_i)$ (*information plane plot*).
- Parts of the IP-plot with the increasing $I(L_i; Y)$ correspond to the so-called *fitting phase*, with the decreasing $I(L_i; Y)$ – to the *compression phase* (see the *fitting-compression hypothesis*).



(a) Negative log likelihood loss (train data)

(b) L_3 (convolutional, LeakyReLU)

Figure 4: Information plane plots for the MNIST classifier. The lower left parts of the plots (b)-(d) correspond to the first epochs. We use 95% asymptotic CIs for the MI estimates acquired from the compressed data. The colormap represents the difference of losses between two consecutive epochs.



(c) L_4 (fully-connected, LeakyReLU) **(d)** L_5 (fully-connected, LogSoftMax)

Figure 4: Information plane plots for the MNIST classifier. The lower left parts of the plots (b)-(d) correspond to the first epochs. We use 95% asymptotic CIs for the MI estimates acquired from the compressed data. The colormap represents the difference of losses between two consecutive epochs.

Results:

- IP-plots are complex, with **several fitting and compression phases**.

Information Bottleneck analysis of CNN classifier

Results:

- IP-plots are complex, with **several fitting and compression phases**.
- **The first switch** from fitting to compression corresponds to the **rapid decrease of the loss**.

Summary

Main contribution:

- New mutual information estimation method based on data compression
- Bounds for mutual information estimation via lossy compression
- Decent results during the high-dimensional synthetic tests
- Can complement any existing MI estimator.

Additional result:

- Our method in application to the CNN classifier reveals several compression and fitting phases in the IP plot. The first switch between the phases corresponds to the rapid decrease of the loss.

Thank you for your attention!