北京大學
PEKING UNIVERSITY

Evaluation of Generative Models:

Density Estimation & Latent Representation

Hao Dong

Peking University

Evaluation of Generative Models: Density Estimation & Latent Representation

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

北京大學
PEKING UNIVERSITY

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Kernel Density Estimation

- Given a sample dataset, how to get its probability density function?

- Parametric Estimation
  - Suppose it conforms to a certain probability distribution
  - Fit the parameters in the distribution according to it
  - Example: Likelihood Estimation, Mixed Gaussian

$$\hat{\theta}_{mle} = \arg\max_{\theta \in \Theta} \hat{\ell}(\theta \mid x_1, x_2, ..., x_n)$$

  - Limitations:
  - Subjective prior knowledge needs to be added
  - Hard to fit the real distribution model

Kernel Density Estimation

- Given a sample dataset, how to get its probability density function?

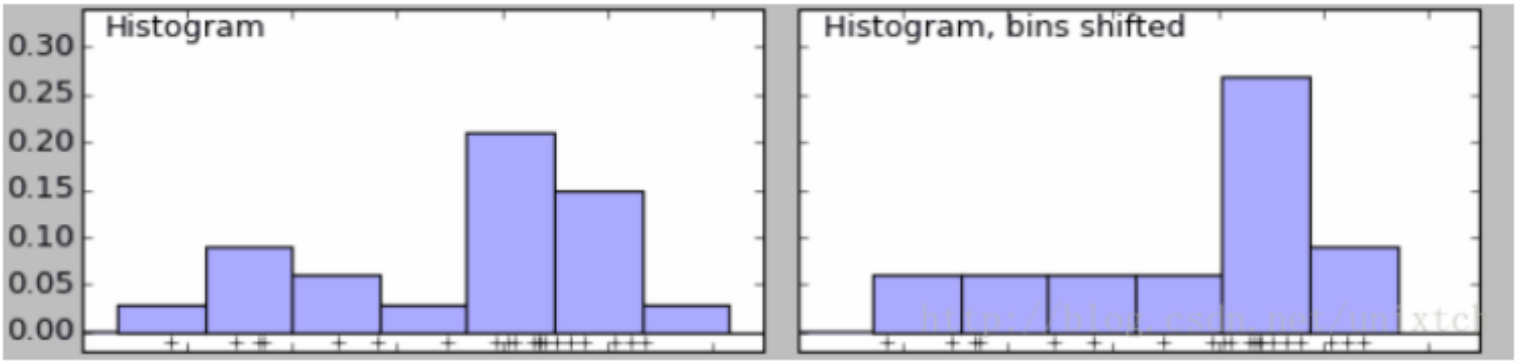
- Non-parametric Estimation
  - No subjective prior knowledge added
  - Only according to the data itself
  - Can get better model than Parametric Estimation

- Kernel Density Estimation is a kind of Non-parametric Estimation
  - Proposed by Rosenblatt (1955) and Emanuel Parzen (1962)
  - Also named Parzen window

Non-parametric Estimation: Histogram

- Given a sample dataset, how to observe its distribution?

- Histogram is an intuitive way to show the distribution
  - The height of each bar is proportional to the number of data point which fall into the interval
  - The width of the bar is an important parameter
  - Different bins partition results in different visual effect



Non-parametric Estimation: Histogram

- Limitations:
  - The distribution curve displayed by histogram is not smooth
  - Samples in a bin have equal probability density
  - For more accurate estimation, increase the number of bins
  - Then every point of the sample has its own probability

- However, it can also cause problems
  - The probability of values not appearing in the sample is 0
  - Discontinuity of probability density function

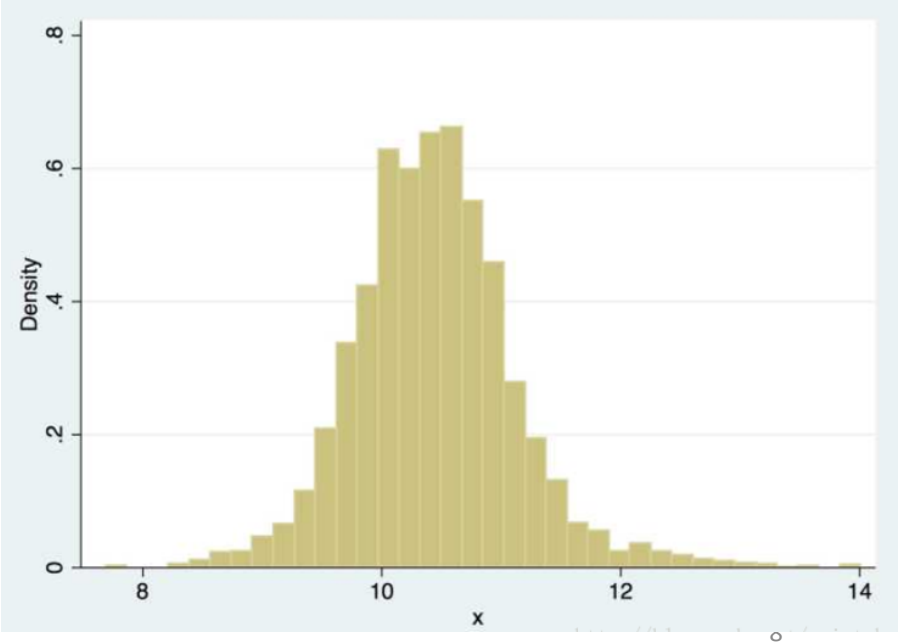
- Thus we should connect these discontinuous intervals

Kernel Density Estimation

- Idea
  - For the probability density of each sample point
  - Take advantage of its neighborhood information
  - Then the discontinuity problem of Histogram is solved
  - Then every point of the sample has its own probability

- For x's neighbourhood $[x-h, x+h]$
  - When $\mathrm{h} \to 0$, the probability density of x's neighborhood can be viewed as probability density of x

$$\hat{f}(x) = \frac{1}{2h} \lim_{h->0} \frac{N_{xi \in [x-h, x+h]}}{N_{total}}$$



Kernel Density Estimation

- Deduction $\hat{f}(x) = \frac{1}{2h} \lim_{h->0} \frac{N_{xi \in [x-h, x+h]}}{N_{total}}$

$$\hat{f}(x) = \frac{1}{2hN_{total}} \sum_{i=x-h}^{x+h} x_i = \frac{1}{hN_{total}} \sum_i \frac{|x - x_i|}{2h} < 1, h- > 0$$

- The choice of h can't be too big or too small
  - Bias-variance Tradeoff
  - Denote $K(x) = \frac{1}{2}\ 1_{\{x<1\}}$ $\hat{f}(x) = \frac{1}{hN_{total}} \sum_i K(\frac{|x - x_i|}{h})$

- The integral of probability density is 1

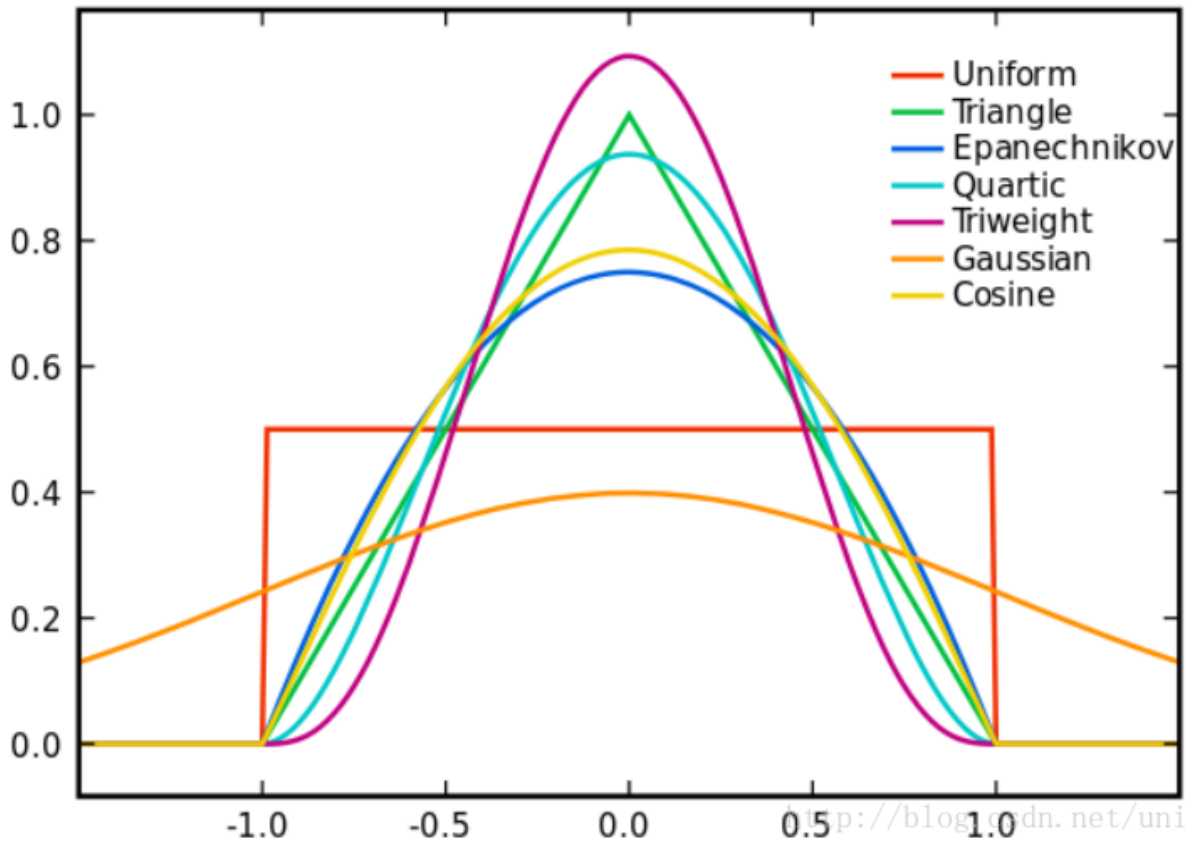
$$\int \hat{f}(x) = \frac{1}{hN_{total}} \sum_i \int K(\frac{|x - x_i|}{h}) dx = \frac{1}{N_{total}} \sum_i \int K(t) dt = \int K(t) dt$$

Kernel Functions

- The integral of $K(x)$ should be 1
  - There are many kinds of kernel functions
  - Uniform
  - Triangular
  - Biweight
  - Triweight
  - Epanechnikov
  - Normal

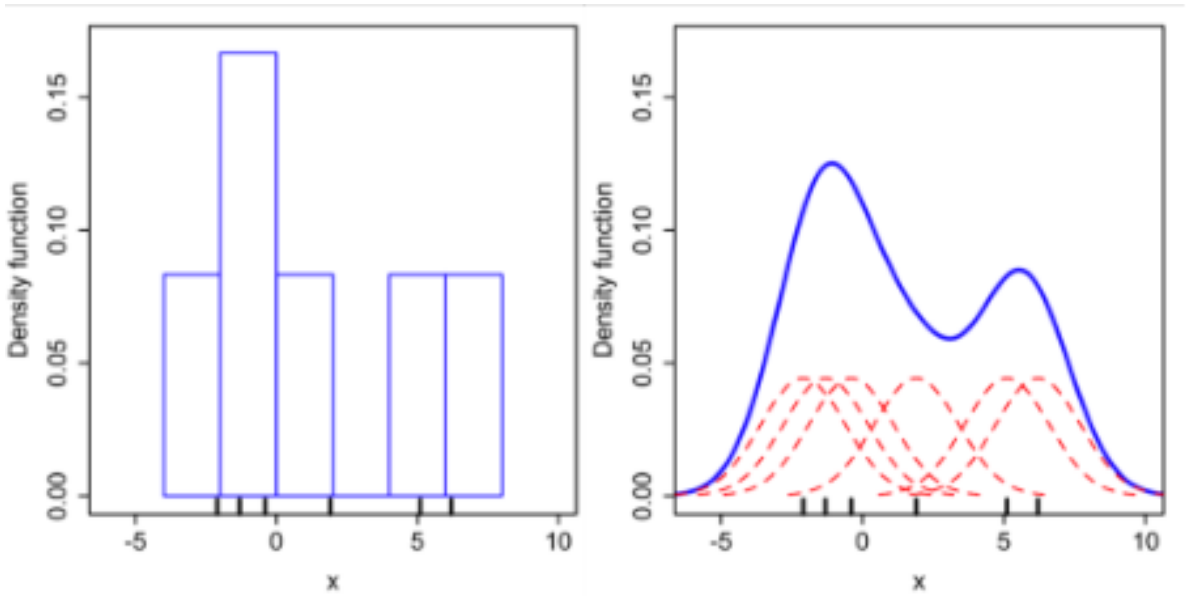
- Gaussian Kernel
  - Convenient to use

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right)$$



An Instance

- Histogram density estimate vs. KDE estimate with Gaussian kernel

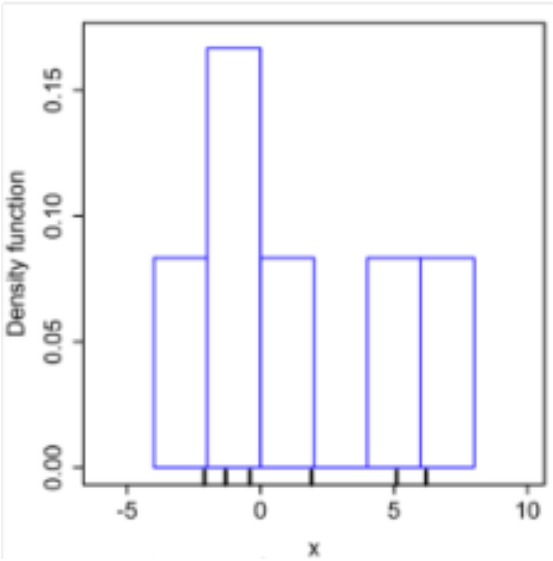


An Instance

- Given: A model $p_\theta(\boldsymbol{x})$ with an intractable/ill-defined density
- Let $S = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \boldsymbol{x}^{(3)}, \boldsymbol{x}^{(4)}, \boldsymbol{x}^{(5)}, \boldsymbol{x}^{(6)}\}$ be 6 data points drawn from p_θ
-

$\mathbf{x}^{(1)}$	$\mathbf{x}^{(2)}$	$\mathbf{x}^{(3)}$	$\mathbf{x}^{(4)}$	$\mathbf{x}^{(5)}$	$\mathbf{x}^{(6)}$
-2.1	-1.3	-0.4	1.9	5.1	6.2

- What is $p_\theta(-0.5)$?
- **Answer 1:** Since $-0.5 \notin S,\ p_\theta(-0.5) = 0$
- **Answer 2:** Compute a histogram by binning the samples
  - Bin width = 2
  - Min height = 1/12
  - Area under histogram should equal 1
  - $p_\theta(-0.5) = 1/6$
  - $p_\theta(-1.99) = 1/6$
  - $p_\theta(-2.01) = 1/12$



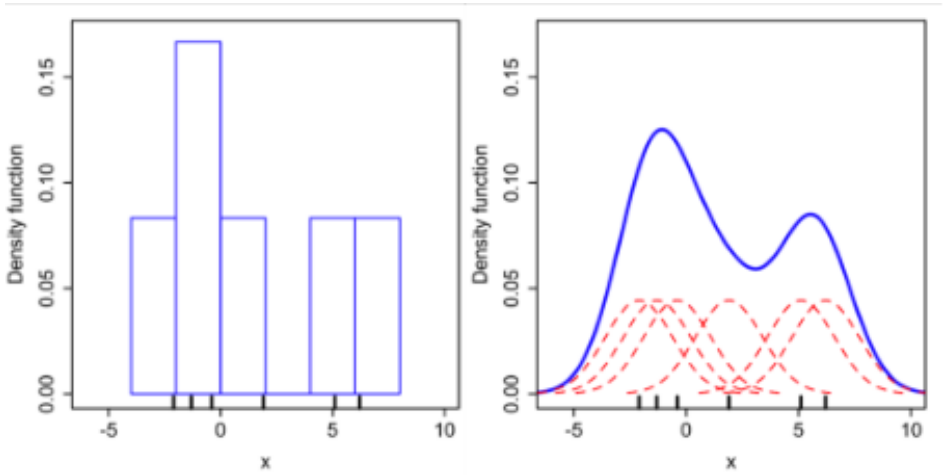
An Instance

- **Answer 3:** Compute kernel density estimate over S

$$\hat{p}(\mathbf{x}) = \frac{1}{n} \sum_{\mathbf{x}^{(i)} \in \mathcal{S}} K\left(\frac{\mathbf{x} - \mathbf{x}^{(i)}}{\sigma}\right)$$

where σ is called the bandwidth parameter and K is called the kernel function.

- Use Gaussian Kernel $K(u) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}u^2\right)$
- Histogram density estimate vs. KDE estimate with Gaussian kernel

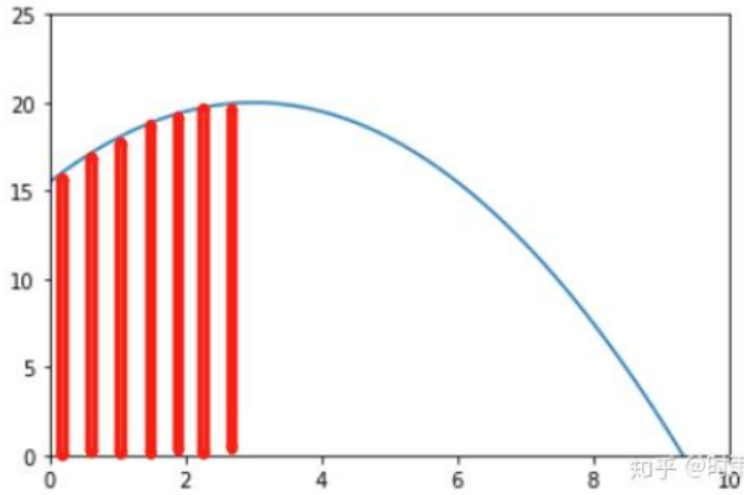


- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Importance Sampling

- Monte Carlo Integration
  - Importance Sampling is a sampling strategy of Monte Carlo Integration
  - The integral curve $f(x)$ is not analytical
  - We want to calculate $\int_a^b f(x)dx$
  - Sample n times in $[a, b]$: $\{x_1, x_2 \ldots, x_n\}$, the values are $\{f(x_1), f(x_2), \ldots, f(x_n)\}$
  - Then

$$\int_a^b f(x)dx = \frac{b-a}{N}\sum_{i=1}^{N} f(x_i)$$



Importance Sampling

- We want to calculate the expectation of $f(x)$
- $x \sim \pi(x)$, however $\pi(x)$ is infeasible to sample
  - Our target: $E[f] = \int_x \pi(x) f(x) dx$
  - Find a distribution $p(x)$ which is feasible to sample
  - Sample n times in $p(x)$: $\{x_1, x_2, \ldots, x_n\}$
  - Then

$$E[f] = \int_x p(x) \frac{\pi(x)}{p(x)} f(x) dx$$

$$E[f] = \frac{1}{N} \sum_{i=1}^{N} \frac{\pi(x_i)}{p(x_i)} f(x_i)$$

  - Where $\frac{\pi(x_i)}{p(x_i)}$ is the weight of importance

Importance Sampling

- **An example**
- We want to calculate the expectation of $\boldsymbol{f(x) = x}$
- $\boldsymbol{x}$ is under the normal distribution of mean 1 and standard deviation 1
- Infeasible to sample from the original distribution $\boldsymbol{N(1, 1)}$
- Can sample from a normal distribution $\boldsymbol{N(1, 0.25)}$
- Suppose that we sample for two times
- The first time: $\boldsymbol{x = 1.09}$
- The first time: $\boldsymbol{x = 2.36}$
- Each time, calculate $\frac{P(x)}{P(x)'} f(x_i)$

- The final result is $(\frac{0.3973}{0.7851} 1.09 + \frac{0.1582}{0.0197} 2.36)/2 = 0.7517$

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

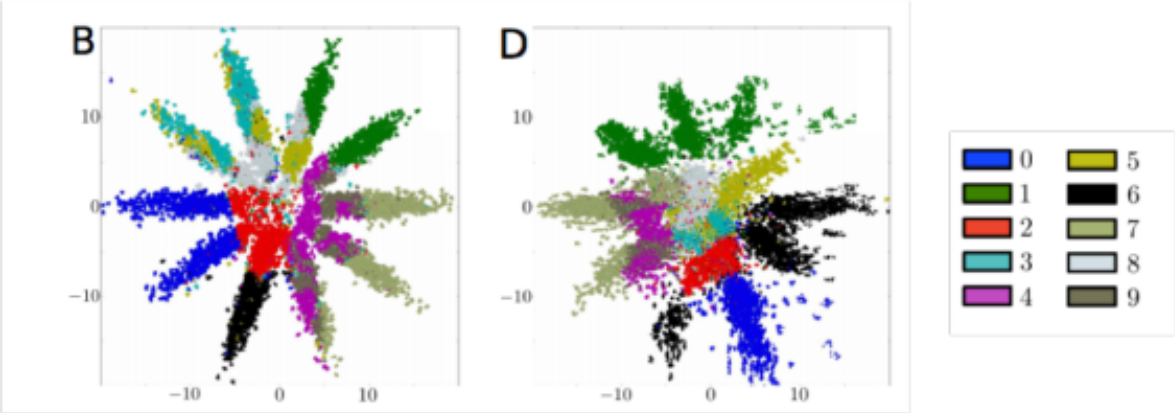
Evaluating Latent Representations

- What does it mean to learn “good” latent representations?
- For a downstream task, the representations can be evaluated based on the corresponding performance metrics e.g., accuracy for semi-supervised learning, reconstruction quality for denoising
- For unsupervised tasks, there is no one-size-fits-all
- Three commonly used notions for evaluating unsupervised latent representations
  - Clustering
  - Compression
  - Disentanglement

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Clustering

- Representations that can group together points based on some semantic attribute are potentially useful (e.g., semi-supervised classification)
- Clusters can be obtained by applying k-means or any other algorithm in the latent space of generative model



- 2D representations learned by two generative models for MNIST digits with colors denoting true labels. Which is better? B or D?

Clustering

- For labelled datasets, there exists many quantitative evaluation metrics
- We want data with the same label to be clustered in the same class
- Note labels are only used for evaluation, not obtaining clusters itself (i.e., clustering is unsupervised)
- Three commonly used metrics of evaluating clustering for labelled dataset
  - **Completeness score** (between [0, 1])
  - **Homogeneity score** (between [0, 1])
  - **V measure score** (also called normalized mutual information, between [0, 1])

Clustering

- $H(C)$ is the class entropy

$$H(C) = -\sum_{c=1}^{|C|} \frac{n_c}{n} \cdot \log\left(\frac{n_c}{n}\right)$$

- $H(C|K)$ is the conditional entropy of a given class

$$H(C|K) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|} \frac{n_{c,k}}{n} \cdot \log\left(\frac{n_{c,k}}{n_k}\right)$$

- The same with $H(K|C)$ and $H(K)$
  - n is the total number of samples
  - n_c and n_k belong are the number of samples of class C and class k respectively
  - $n_{c,k}$ are the number of samples divided from class C to class K.

Clustering

- **Completeness score(c):** maximized when all the data points that are members of a given class are elements of the same cluster

$$c = 1 - \frac{H(K|C)}{H(K)}$$

- **Homogeneity score(h):** maximized when all of its clusters contain only data points which are members of a single class

$$h = 1 - \frac{H(C|K)}{H(C)}$$

$$v = 2 \cdot \frac{h \cdot c}{h + c}$$

- **V measure score(v):** harmonic mean of completeness and homogeneity score

Clustering

- **Homogeneity, Completeness, and mean V measure score**

- **Advantages:**
  - **Clear score:** from 0 to 1, it reflects the worst to the best performance;
  - **The explanation is intuitive:** the harmonic mean of the difference can be qualitatively analyzed in terms of homogeneity and integrity;
  - **The cluster structure is not assumed:** the results of two clustering algorithms, such as k-means algorithm and spectral clustering algorithm, can be compared.

- **Limitations:**
  - Completely random labels do not always produce the same completeness and homogeneity values.
  - Thus the harmonic mean v-measure obtained is not the same.
  - In particular, random markers do not produce zero scores, especially when the number of clusters is large.

Clustering

- **Implementation**

```
>>> from sklearn import metrics
>>> labels_true = [0, 0, 0, 1, 1, 1]
>>> labels_pred = [0, 0, 1, 1, 2, 2]

>>> metrics.homogeneity_score(labels_true, labels_pred)
0.66...

>>> metrics.completeness_score(labels_true, labels_pred)
0.42...

>>> metrics.v_measure_score(labels_true, labels_pred)
0.51...

>>> metrics.homogeneity_completeness_v_measure(labels_true, labels_pred)
...
(0.66..., 0.42..., 0.51...)

>>> labels_true = [0, 0, 0, 1, 1, 1]
>>> labels_pred = [0, 0, 0, 1, 2, 2]
>>> metrics.homogeneity_completeness_v_measure(labels_true, labels_pred)
...
(1.0, 0.68..., 0.81...)
```

Why the V measure score is better in the second example?

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Compression

- Latent representations can be evaluated based on the **maximum compression** they can achieve without significant loss in reconstruction accuracy

Source: Santurkar et al., 2018

- Some standard metrics for reconstruction
  - Mean Squared Error (MSE)
  - Peak Signal to Noise Ratio (PSNR)
  - Structure Similarity Index (SSIM)

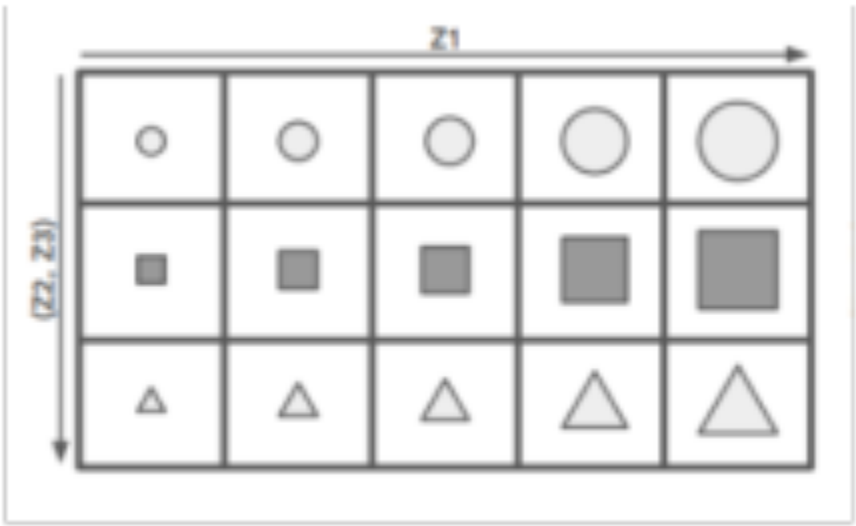
北京大學
PEKING UNIVERSITY

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Disentanglement

- Intuitively, we want representations that disentangle independent and interpretable attributes of the observed data

Source: Higgins et al., 2018

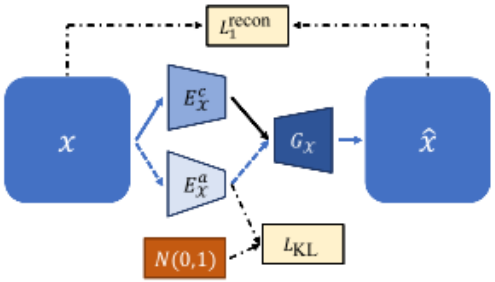


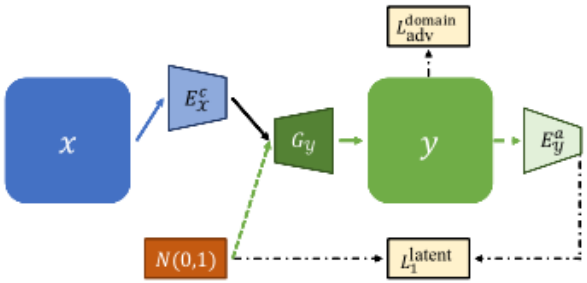
Source: Shu et al., 2019

- Provide user control over the attributes of the generated data
  - When Z_1 is fixed, size of the generated object never changes
  - When Z_1 is changed, the change is restricted to the size of the generated object

Disentanglement: DRIT

- Content representation
- Attribute representation
- Content encoder
- Attribute encoder





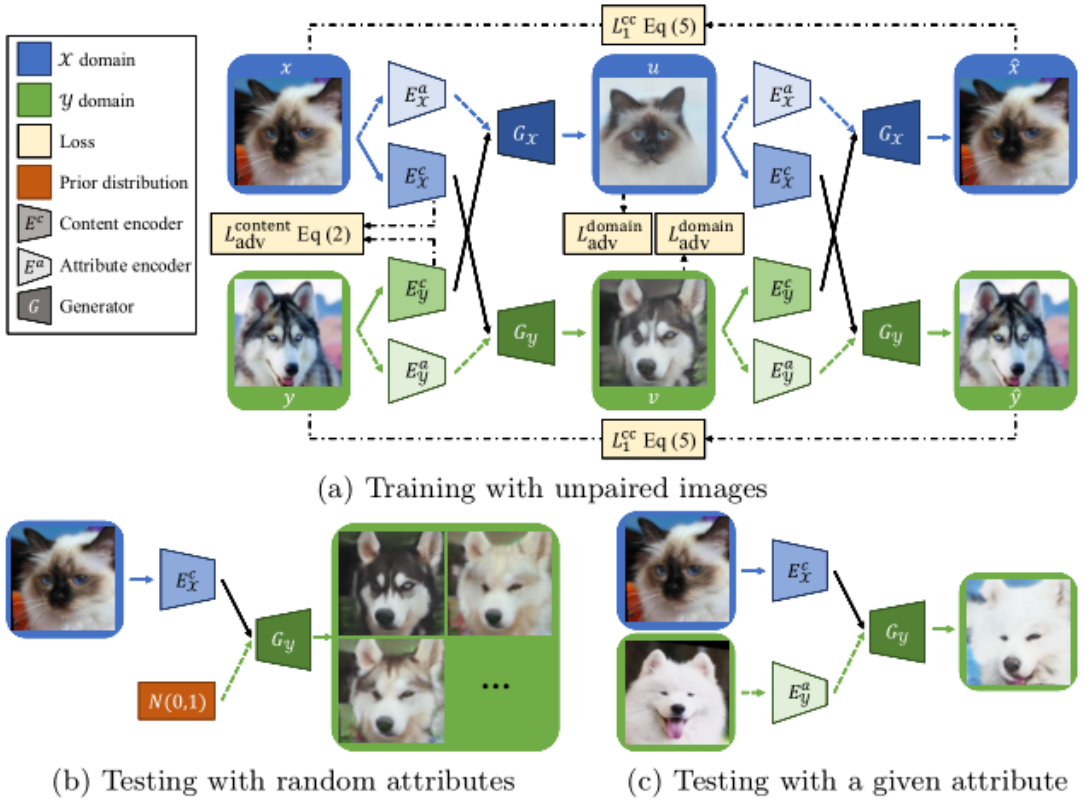


Fig. 3: **Method overview.** (a) With the proposed content adversarial loss $L_{\text{adv}}^{\text{content}}$ (Section 3.1) and the cross-cycle consistency loss L_1^{cc} (Section 3.2), we are able to learn the multimodal mapping between the domain $\mathcal{X}$ and $\mathcal{Y}$ with unpaired data. Thanks to the proposed disentangled representation, we can generate output images conditioned on either (b) random attributes or (c) a given attribute at test time.

Disentanglement

- Disentangling generative factors is theoretically impossible without additional assumptions

Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Rätsch, Sylvain Gelly, Bernhard Schölkopf, Olivier Bachem

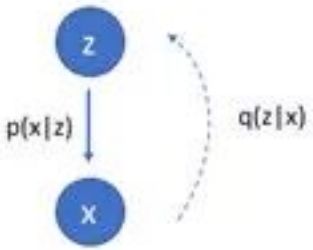
(Submitted on 29 Nov 2018 (v1), last revised 18 Jun 2019 (this version, v4))

The key idea behind the unsupervised learning of disentangled representations is that real-world data is generated by a few explanatory factors of variation which can be recovered by unsupervised learning algorithms. In this paper, we provide a sober look at recent progress in the field and challenge some common assumptions. We first theoretically show that the unsupervised learning of disentangled representations is fundamentally impossible without inductive biases on both the models and the data. Then, we train more than 12000 models covering most prominent methods and evaluation metrics in a reproducible large-scale experimental study on seven different data sets. We observe that while the different methods successfully enforce properties ``encouraged'' by the corresponding losses, well-disentangled models seemingly cannot be identified without supervision. Furthermore, increased disentanglement does not seem to lead to a decreased sample complexity of learning for downstream tasks. Our results suggest that future work on disentanglement learning should be explicit about the role of inductive biases and (implicit) supervision, investigate concrete benefits of enforcing disentanglement of the learned representations, and consider a reproducible experimental setup covering several data sets.

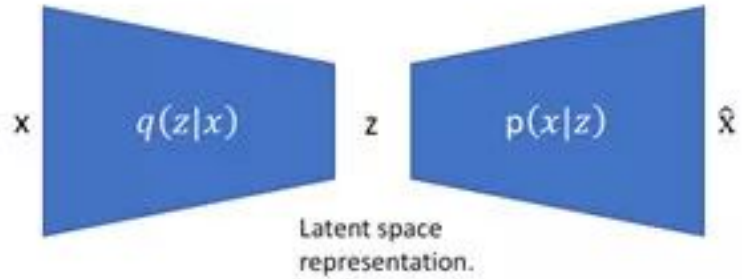
Subjects: **Machine Learning (cs.LG)**; Artificial Intelligence (cs.AI); Machine Learning (stat.ML)
Journal reference: Proceedings of the 36th International Conference on Machine Learning (ICML 2019)
Cite as: arXiv:1811.12359 [cs.LG]
(or arXiv:1811.12359v4 [cs.LG] for this version)

Recap: Vanilla VAE

- Variational Auto Encoder (VAE)



We'd like to use our observations to understand the hidden variable.



- Classical VAE Loss

$$
\begin{aligned}
L_{\mathrm{VAE}}(\theta, \phi) &= -\log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) \\
&= -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} p_\theta(\mathbf{x}|\mathbf{z}) + D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})\right) \\
\theta^*, \phi^* &= \arg\min_{\theta,\phi} L_{\mathrm{VAE}}
\end{aligned}
$$

Recap: Beta-VAE

- β-VAE is a variant of VAE
  - β-VAE enhances the ability of VAE in terms of disentanglement
  - Maximize the probability of generating real data
  - Minimize the KL divergence of real and estimated posterior distributions
- The corresponding Lagrange function is:

$$\begin{aligned}\mathcal{F}(\theta,\phi,\beta) &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) - \beta\big(D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right) - \delta\big)\\ &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right) + \beta\delta\\ &\geq \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\mathbf{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right)\end{aligned}$$

- To maximize the $F(\theta,\phi,\beta)$, the loss is:

$$L_{\mathrm{BETA}}(\phi,\beta) = -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}\log p_\theta(\mathbf{x}|\mathbf{z}) + \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right)$$

Recap: Beta-VAE

- β-VAE is a variant of VAE

$$\begin{aligned}\mathcal{F}(\theta, \phi, \beta) &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta\big(D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right) - \delta\big) \\ &= \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right) + \beta\delta \\ &\geq \mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\mathbf{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right)\end{aligned}$$

$$L_{\mathrm{BETA}}(\phi, \beta) = -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) + \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})\right)$$

- Lagrange multiplier β is a super parameter.
  - A higher beta value reduces the information richness of z representation in the latent variable space, but increases the ability of disentanglement
  - β can be used as a balance factor between representation ability and disentanglement ability.

Mutual Information Gap

- β-VAE provides an evaluation metric for disentanglement
  - Accuracy of a linear classifier that predicts a fixed factor of variation

- β-TCVAE provides another evaluation metric for disentanglement
- Mutual Information
  - The empirical mutual information between latent variable z_j and a real factor v_k can be represented by a joint distribution

$$q(z_j, v_k) = \sum_{n=1}^{N} p(v_k)p(n|v_k)q(z_j|n)$$

  - Assuming that latent variable factor $p(v_k)$ and generation process are known
  - Mutual information is the following formula.

$$I_n(z_j; v_k) = \mathbb{E}_{q(z_j, v_k)}\left[\log \sum_{n \in \mathcal{X}_{v_k}} q(z_j|n)p(n|v_k)\right] + H(z_j)$$

  - Where $H(z_j)$ is the Shannon entropy of latent variable z_j.

Mutual Information Gap

- Nevertheless…
  - If a real factor v_k has high mutual information with many latent variable z_js
  - In this case, we only want the maximum mutual information value
  - Thus, mutual information isn't a good metric
  - The gap between the largest and second largest mutual information works!

- Mutual Information Gap (MIG)
  - The largest mutual information value minus the second largest one:

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{H(v_k)}\left(I_n\left(z_{j^{(k)}};v_k\right)-\max_{j\neq j^{(k)}}I_n\left(z_j;v_k\right)\right)$$

- This formula is a widely used evaluation metric for disentanglement.

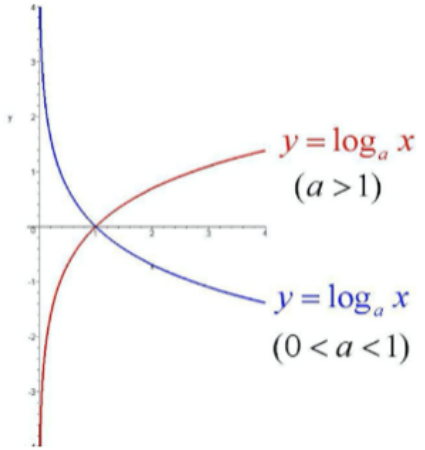
- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Negative Log Likelihood

- Negative Log Likelihood (NLL)

$$L(y) = -log(y)$$

- Applied in generative models
  - We want to find a set of parameters to minimize the loss
  - The function is the logarithm of probability distribution
  - The value of p is $0 \leq p \leq 1$
  - After logarithm, the curve is shown as the red curve in $[0,1]$
  - Minus the function, we can get the NLL curve in black



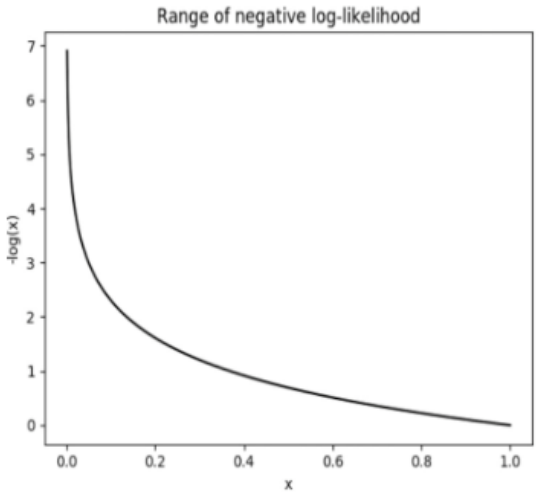


Figure: *The loss function reaches infinity when input is 0, and reaches 0 when input is 1.*

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

Evaluation of VAE

- Negative Log Likelihood (NLL)
- NLL is a term in the loss of classical VAE

$$
\begin{aligned}
L_{\mathrm{VAE}}(\theta, \phi) &= -\log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})\right) \\
&= -\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})} p_\theta(\mathbf{x}|\mathbf{z}) + D_{\mathrm{KL}}\left(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})\right) \\
\theta^*, \phi^* &= \arg\min_{\theta,\phi} L_{\mathrm{VAE}}
\end{aligned}
$$

- Also, it can be used as an evaluation metric to evaluate VAE
  - NLL represents the probability of generating real data
  - Less NLL indicated better generation of VAE

Summary

- Density Estimation
  - Kernel Density Estimation
  - Importance Sampling
- Latent Representation
  - Clustering
  - Compression
  - Disentanglement
- Others
  - NLL
  - VAE Evaluation

北京大學
PEKING UNIVERSITY

Thanks