A Gentle Introduction to Identifiable Generative Models

Erdun Gao

CLeaR, Unimelb

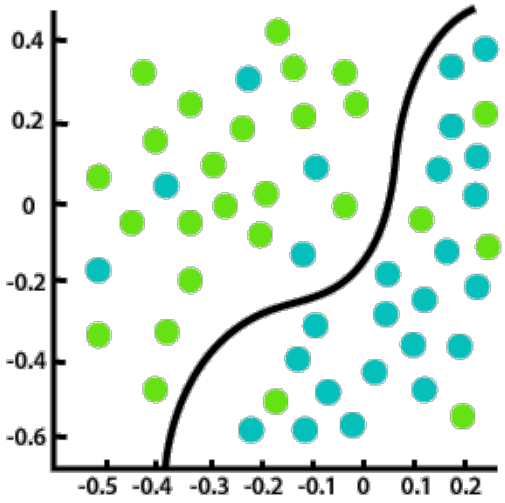
17 March 2022

Some of the contents/viewpoints are cited from *Nonlinear Independent Component Analysis*

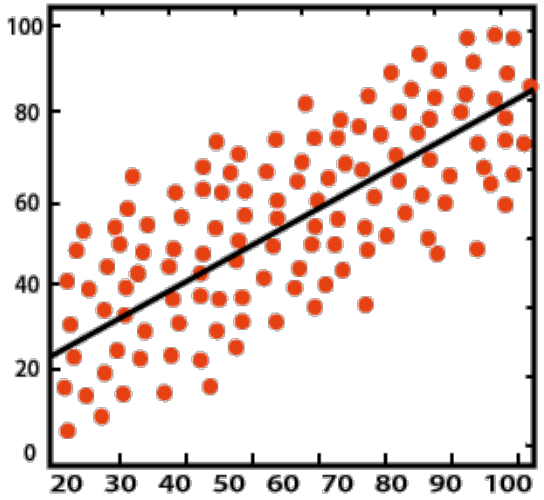
Table of Contents

Unsupervised Learning

- *Supervised Learning*



Classification



Regression

With N examples sampled from $P(X, Y)$, seek a function g and constraint the learning procedures by using a score function $f: X \times Y \rightarrow \mathbb{R}$ so that

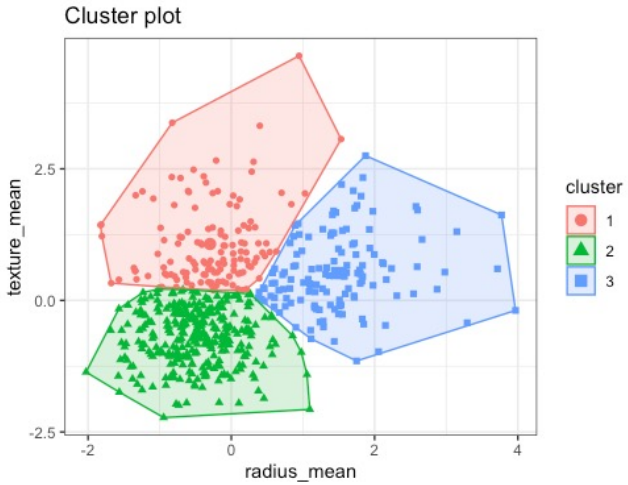
$$g(x) = \arg max\, f(x, y)$$

Unsupervised Learning

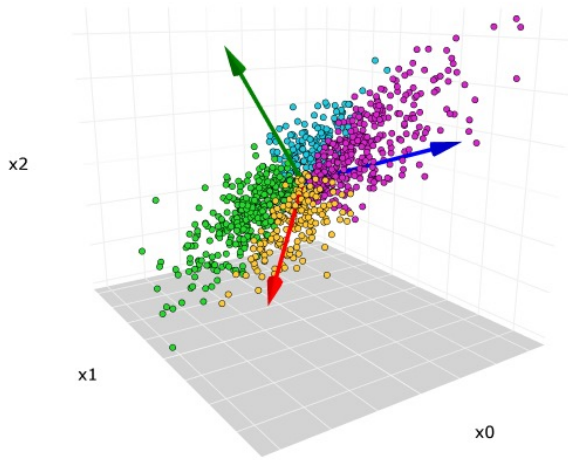
Problems:

😭 Labels may be difficult to obtain
😭 Human annotation may be required
😭 Labels may not be informative

$\Longrightarrow$ With only $P(X)$, what to learn?

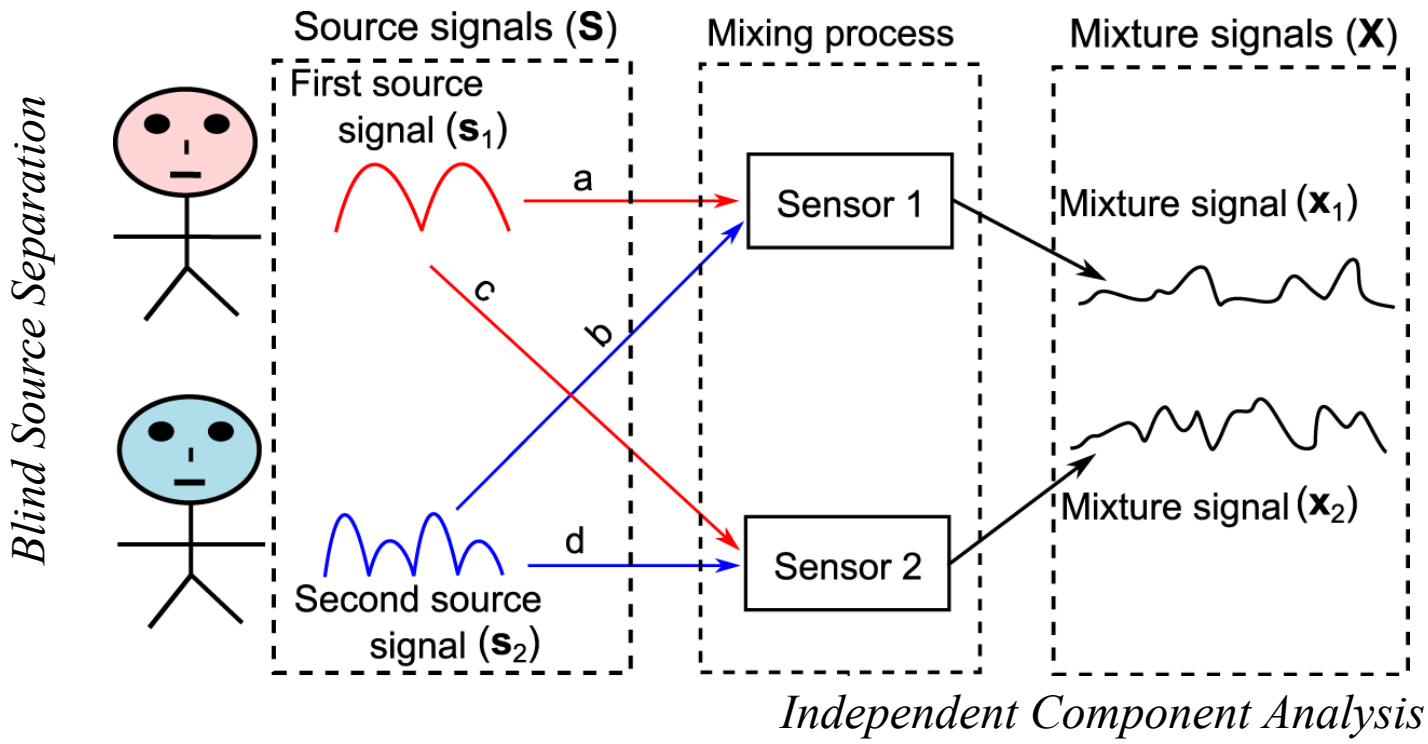


Clustering



Dimensions Reduction

Unsupervised Learning

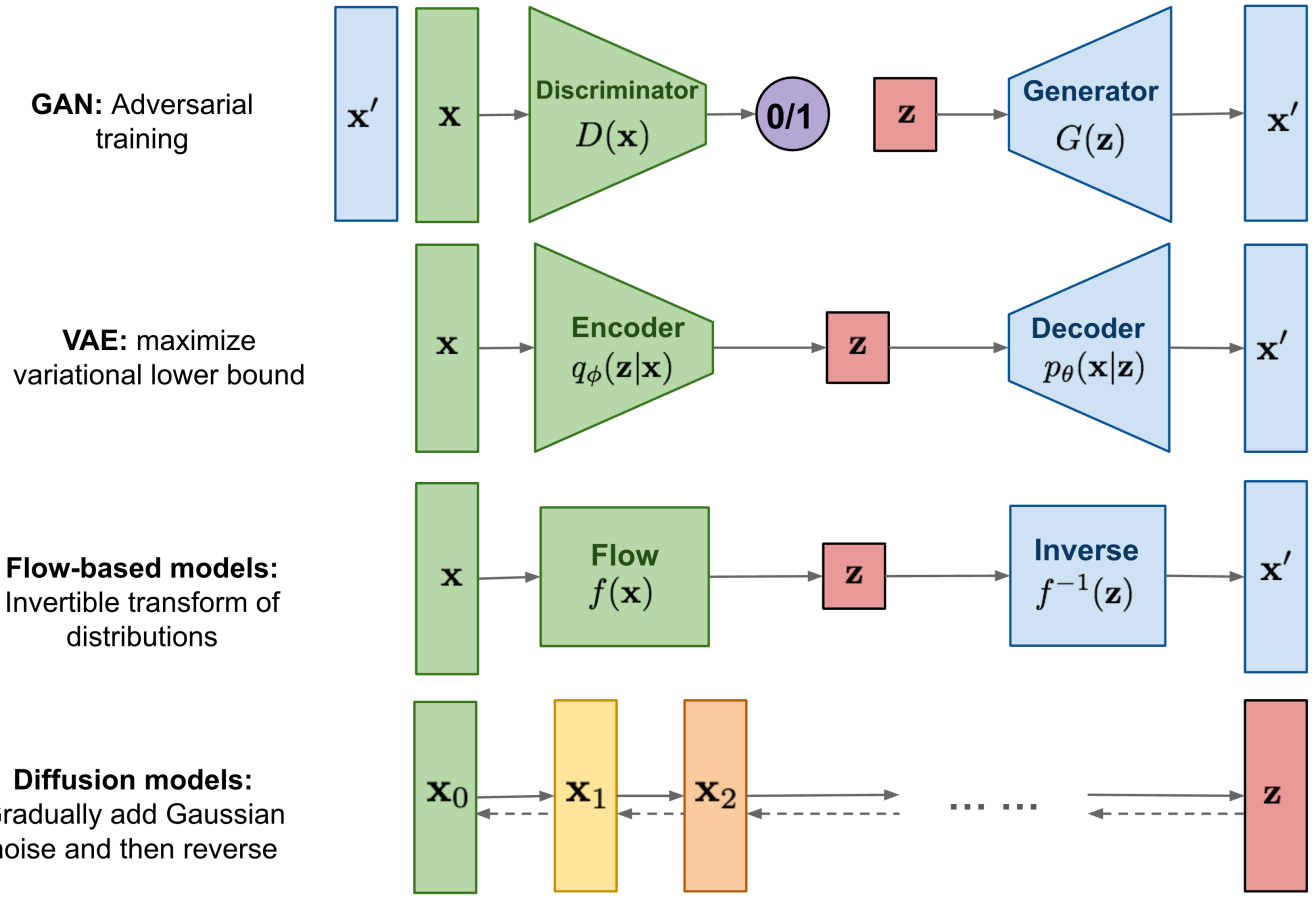


Assume observed signals $x = (x_1, x_2, \cdots, x_d)$ are generated as a transformation $f = (f_1, f_2, \cdots, f_d)$ of d independent source signals $s = (s_1, s_2, \cdots, s_d)$:

$$x_i = f_i(s)$$

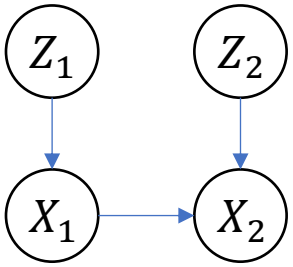
Unsupervised Learning

$$p(x) = \int p(x|z)p(z)dz$$



Identifiable Models

θ $p(X)$ $(x^1, x^2, \cdots, x^n)$



Assume the GT model is $X_1 = Z_1 \sim \mathcal{N}(0,1)$ and $X_2 = 2X_1 + Z_2 \sim \mathcal{N}(0,1)$

Then, the covariance matrix is

$$cov = \begin{bmatrix} 1 & 2 \\ 2 & 5 \end{bmatrix}$$

While the other model $X_2 = Z_2 \sim \mathcal{N}(0,5)$ and $X_2 = \frac{2}{5}Z_1 \sim \mathcal{N}(0,1)$ can also induce the totally same distribution.

Identifiable Models

A model is identifiable if it is theoretically possible to learn the ***true model's parameters*** after obtaining an infinite observations. Or, different parameters must induce different probability distributions.

Mathmatically, let $\mathcal{P} = \{P_\theta : \theta \in \Theta\}$ be a statistical model where the parameter space Θ is either finite or infinite dimensional. We say that $\mathcal{P}$ is identifiable if the mapping $\theta \mapsto P_\theta$ is one-to-one:

$$P_{\theta_1} = P_{\theta_2} \Longrightarrow \theta_1 = \theta_2 \text{ for all } \theta_1, \theta_2 \in \Theta$$

In the ICA problem, the identifiability is defined to recover the parameters of all functions in $\mathcal{F}$.

Theoretical results on ICA

- *Linear independent component analysis (ICA)*

$$x = As \; where \; x_i = \sum_{j=1}^{d} A_{ij} s_j \;\; \text{for all} \;\; i = 1,2,\cdots,d$$

A_{ij}constant parameters describing "mixing" and A is the mixing matrix.

Theorem:

1) All the independent components s_i must be non-Gaussian.
2) The number of observed linear mixtures must at least as large as the number of independent components.
3) A must be of full column rank (invertible).

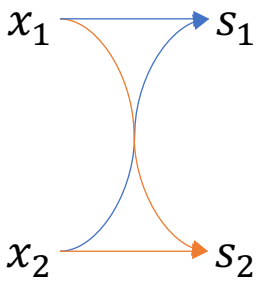
Then, observing only x we can recover both A and s.

Theoretical results on ICA

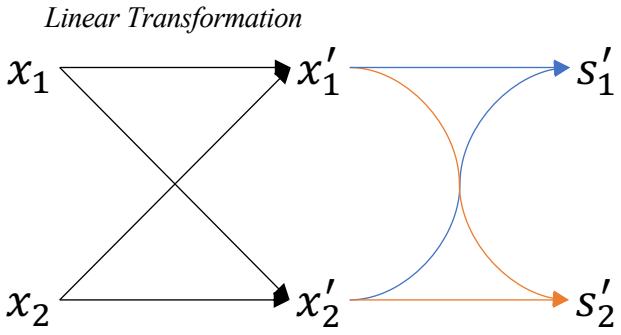
- *Non-Linear ICA*

  - 😀 Extending ICA to the non-linear case can get general disentanglement.

  - 😌 The general nonlinear ICA is not identifiable. That is to say, we cannot recover the origianl sources.



Ground-truth models



Hyvärinen A, Pajunen P. Nonlinear independent component analysis: Existence and uniqueness results[J]. Neural networks, 1999, 12(3): 429-439.

Nonlinear ICA

- *The final task of Nonlinear ICA*

$$x = f(s) \qquad \Rightarrow \qquad s = g(x)$$

Generally, we do not put many constraints on f but usually (1) smooth and (2) invertible. Then, we need to constraint the distribution of s.

With:

$$p(x)|\det(J_f)| = p(s) \quad \Leftrightarrow \quad p(x) = p(s)|\det(J_g)|$$

$$\log p(x) = \sum_{i=1}^{d} \log p(s_i) + \log|\det(J_g)|$$

$$\log p(s_i) = q_{i,0}(s_i) + \sum_{v=1}^{V} \lambda_{i,v}(\tau) q_{i,v}(s_i) - \log Z(\lambda_{i,1}(\tau), \cdots, \lambda_{i,V}(\tau))$$

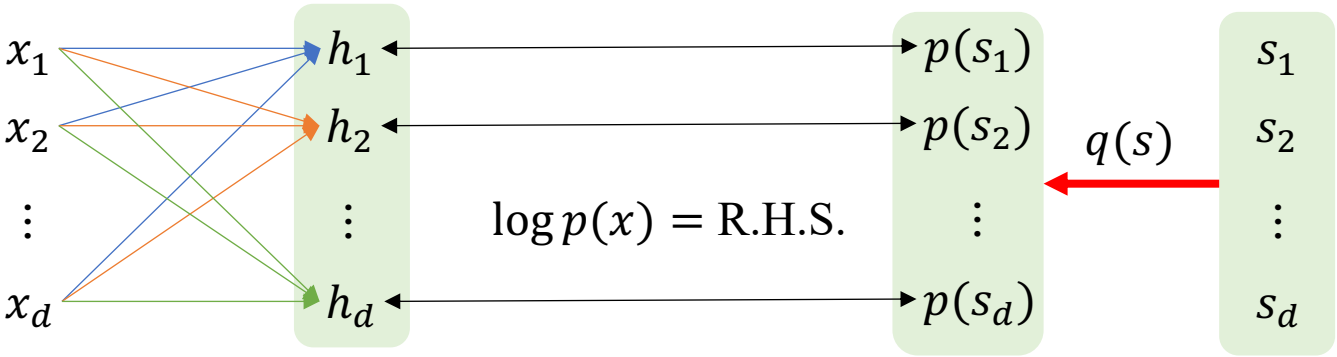
Nonlinear ICA

$$p(x)|\det(J_f)| = p(s) \quad \Leftrightarrow \quad p(x) = p(s)|\det(J_g)|$$

$$\log p(x) = \sum_{i=1}^{d} \log p(s_i) + \log|\det(J_g)|$$

One straightforward way:

😎 Notice that R.H.S. is the sum of d independent components. That is to say, it can expand a d dimensional vector space. More complex, easier to be identified (no free lunch).



Nonlinear ICA

$$p(x)|\det(J_f)| = p(s) \quad \Leftrightarrow \quad p(x) = p(s)|\det(J_g)|$$

$$\log p(x) = \sum_{i=1}^{d} \log p(s_i) + \log|\det(J_g)|$$

Two more easy problems:

😌 How to connect $\{h_1(x), h_2(x), \cdots, h_d(x)\}$ with $\log p(x)$?

😌 How to elimate the Jacobian term ? (Keep in mind that it will magically disappear.)

One terrible problem:

😵 How to identify $p(s_1), p(s_2), \cdots, p(s_d)$ from $\{h_1(x), h_2(x), \cdots, h_d(x)\}$?

Nonlinear ICA

Two more easy problems:

😌 How to connect $\{h_1(x), h_2(x), \cdots, h_d(x)\}$ with $\log p(x)$?

➢ *Logistic regression*

First, let us take the $t = 1$ (T classes totally) as pivot. Then, we have

$$p(t = 1|x; \theta, w) = \frac{1}{1 + \sum_{t=2}^{T} e^{w_t h(x;\theta)}} \quad p(t = \tau|x; \theta, w) = \frac{e^{w_\tau h(x;\theta)}}{1 + \sum_{t=1}^{T} e^{w_t h(x;\theta)}}$$

$$p(t = \tau|x) = \frac{p_\tau(x)p(t = \tau)}{\sum_{t=1}^{T} p_t(x)p(t = \tau)} \qquad \text{where } p_\tau(x) = p(x|t = \tau)$$

With infinite examples, the above things lead to the relationship

$$w_\tau h(x;\theta) = \log p_\tau(x) - \log p_1(x) + \log \frac{p(t = \tau)}{p(t = 1)}$$

😎

Nonlinear ICA

Two more easy problems:

😌 How to connect $\{h_1(x), h_2(x), \cdots, h_d(x)\}$ with $\log p(x)$?

The remaining question is:

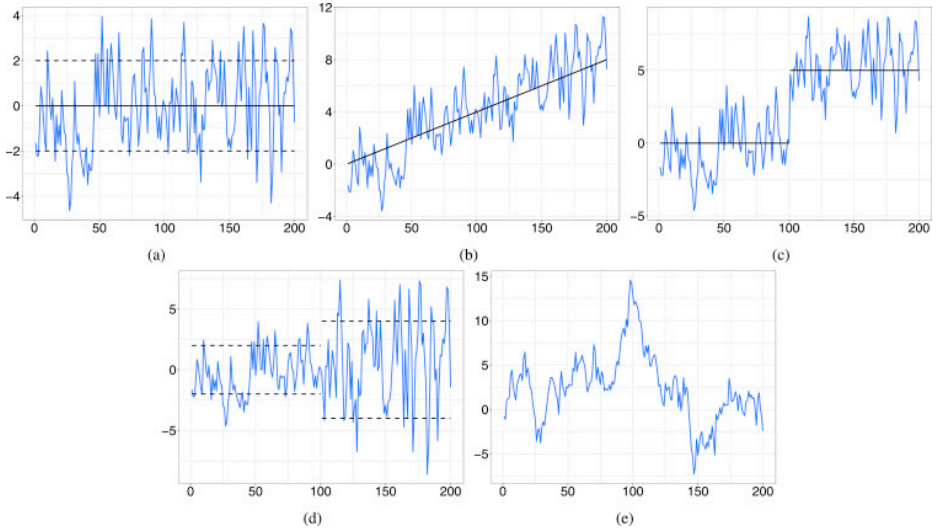
How can we constructure the (multinomial) logistic regression things?

Constrcure some classification tasks? 😄

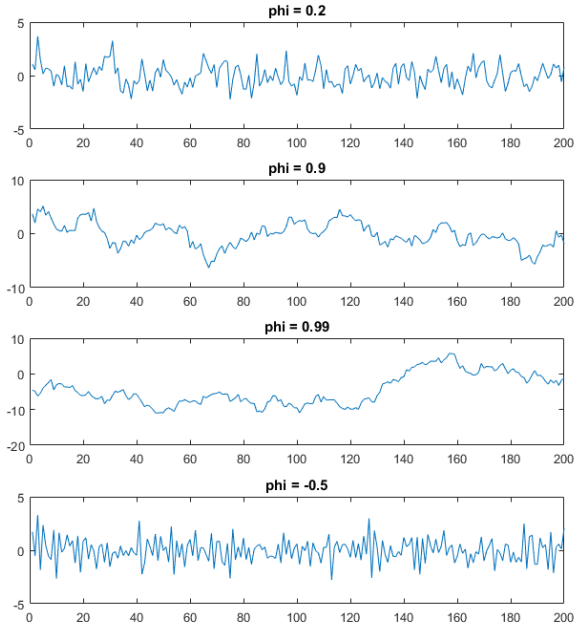
Unsupervised task 😔 Self-supervised? Yes! 😄

Nonlinear ICA

Temporal structural data:

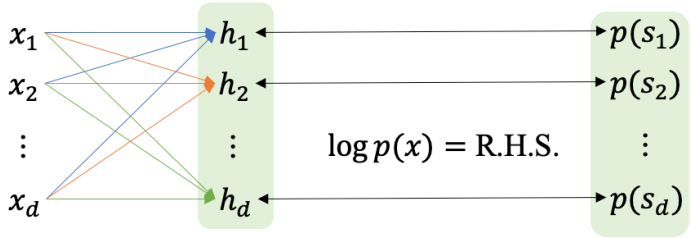


Nonstationary
(TCL, NIPS16)

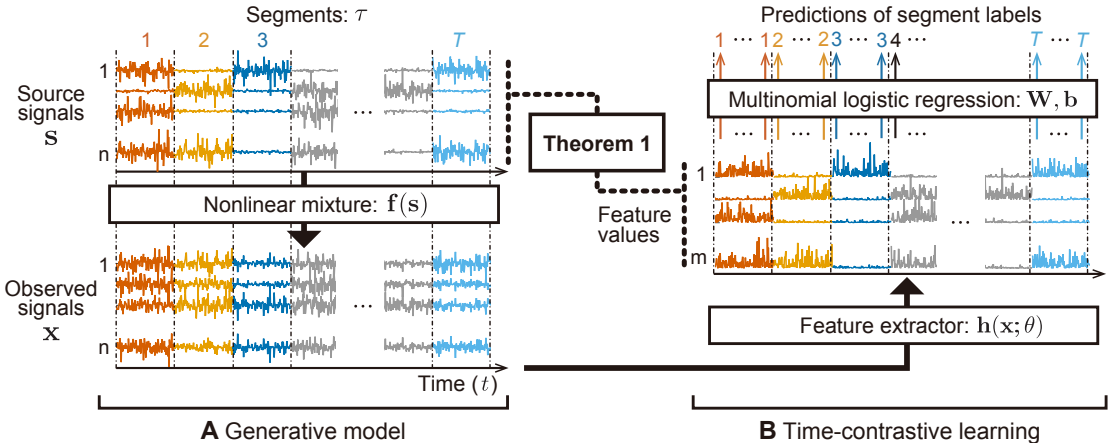


Autocorrelations
(PCL, AISTATS17)

Nonlinear ICA



- Temporal Contrastive Learning (TCL), NIPS16



The temporal data are assumed to be nonstationary and segmented to T parts, which leads to a T-classification task.

$$\log p(s_i) = q_{i,0}(s_i) + \sum_{v=1}^{V} \lambda_{i,v}(\tau) q_{i,v}(s_i) - \log Z(\lambda_{i,1}(\tau), \cdots, \lambda_{i,V}(\tau))$$

Theorem: The modulation parameter matrix $L_{\tau,i} = \lambda_{i,1}(\tau) - \lambda_{i,1}(1), \tau = 1, \cdots, T; i = 1, \cdots, N$ has full column rank N. Then, $q(s)$ can be identified up to an invertible linear transformation.

$$q(s) = Ah(x; \theta) + d$$

Nonlinear ICA

- Temporal Contrastive Learning (TCL), NIPS16

$$q(s) = Ah(x;\theta) + d$$

Proof Sketch:

GT: $$\log p_\tau(x) = \sum_{i=1}^{d} \lambda_{\tau,i} q(g_i(x)) + \log|\det(J_g)| - \log Z(\lambda_\tau)$$

Learning: $$\log p_\tau(x) = \sum_{i=1}^{d} w_{\tau,i} h_i(x) + \log p_1(x) - \log \frac{p(t=\tau)}{p(t=1)}$$

$$= \sum_{i=1}^{d} (w_{\tau,i} h_i(x) + \lambda_{1,i} q(g_i(x))) + \log|\det(J_g)| - \log Z(\lambda_1) - \log \frac{p(t=\tau)}{p(t=1)}$$

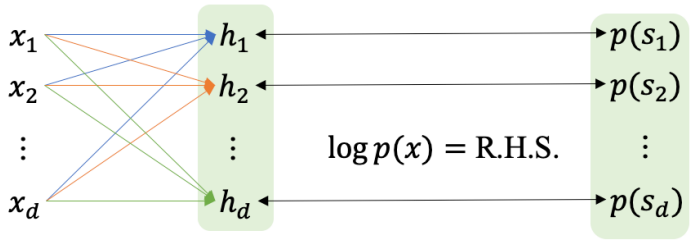
Then: $$\sum_{i=1}^{d} (\lambda_{\tau,i} - \lambda_{1,i}) q(g_i(x)) = \sum_{i=1}^{d} w_{\tau,i} h_i(x) + \log \frac{Z(\lambda_\tau)}{Z(\lambda_1)} - \log \frac{p(t=\tau)}{p(t=1)}$$

$$Lq(s) = Wh(x) + \beta$$

We have $L^+L = I$:

$$q(s) = L^+Wh(x) + L^+\beta$$

Nonlinear ICA



- Permutation Contrastive Learning (PCL), AISTATS17

Autocorrelation describes the correlations encoded in the temporal structure.

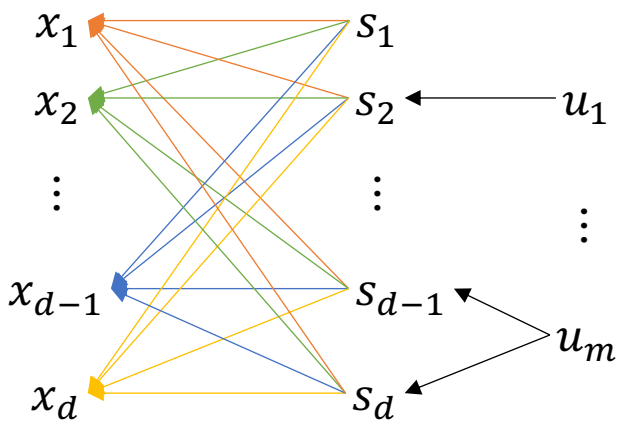
$$y(t) = \begin{pmatrix} x(t) \\ x(t-1) \end{pmatrix} \qquad y^*(t) = \begin{pmatrix} x(t) \\ x(t^*) \end{pmatrix}$$

$$r(y) = \sum_{i=1}^{d} B_i \left(h_i(y^1), h_i(y^2)\right)$$

In $y(t)$, $x(t)$ and $x(t-1)$ are correlated, so $\log p_1(y)$ models the joint distribution.
In $y^*(t)$, $x(t)$ and $x(t^*)$ are independent , so $\log p_2(y)$ can be decomposited.

Nonlinear ICA

- Generalized Contrastive Learning (GCL), AISTATS19



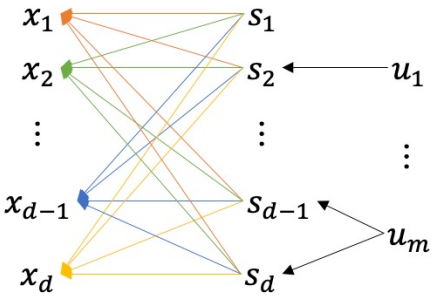
Assume that each s_i is statistically dependent on u, but conditionally independent of the other s_j:

$$\log p(s|u) = \sum_{i=1}^{d} q_i(s_i, u)$$

$$p(s_i|u) = \frac{Q_i(s_i)}{Z_i(u)} \exp\left[\sum_{i=1}^{d} \tilde{q}_{ij}\,(s_i)\lambda_{ij}(u)\right]$$

the sufficient statistics $\tilde{q}_{ij}$ are assumed linearly independent (over j for each i)

Nonlinear ICA

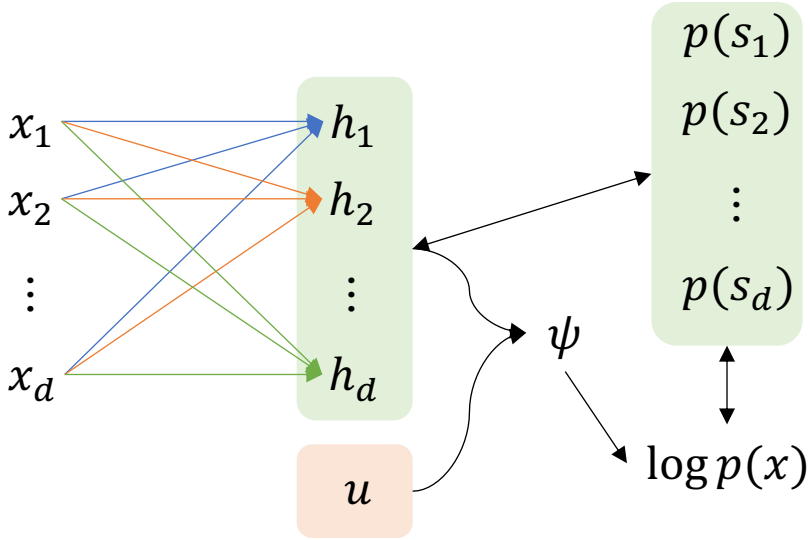


- Generalized Contrastive Learning (GCL), AISTATS19

Learning Algorithm:

$$\tilde{x} = (x, u) \qquad \tilde{x}^* = (x, u^*)$$

where u^* is a random value from the distribution of the u, but independent of x



$$r(x, u) = \sum_{i=1}^{d} \psi_i(h_i(x), u)$$

Nonlinear ICA

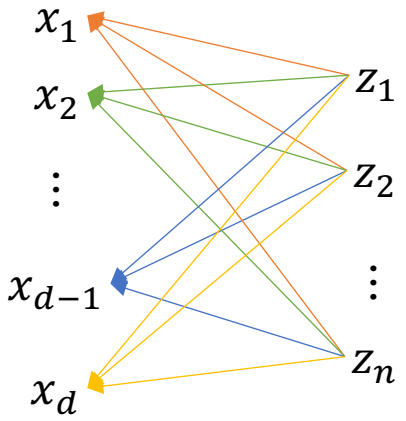
- Generalized Contrastive Learning (GCL), AISTATS19

Assumptions for identifiability:

1. The conditional log-pdf q_i is sufficiently smooth as a function of s_i for any fixed u.
2. [Assumption of Variability] For any $y \in \mathbb{R}^n$, there exists $2n+1$ values for u, denoted by $u_j, j = 0, \cdots, 2n$ such that the $2n$ vectors in $\mathbb{R}^{2n}$ given by
   $\big(\big(w(y,u_1) - w(y,u_0)\big), \big(w(y,u_2) - w(y,u_0)\big), \cdots, \big(w(y,u_{2n}) - w(y,u_0)\big)\big)$
   with
   $$w(y,u) = \left(\frac{\partial q_1(y_1,u)}{\partial y_1}, \cdots, \frac{\partial q_n(y_n,u)}{\partial y_n}, \frac{\partial^2 q_1(y_1,u)}{\partial y_1^2}, \cdots, \frac{\partial^2 q_n(y_n,u)}{\partial y_n^2}\right)$$
   are linearly independent.

The functions $h_i(x)$ give the independent components, up to scalar (component-wise) invertible transformations.

Identifiable DGM



- Variational AutoEncoder (VAE)

$$\log p_\theta(x) = \log \int p_\theta(x,z)dz$$

$$= \log \int q_\phi(z|x)\frac{p_\theta(x,z)}{q_\phi(z|x)}dz = \log E_{z\sim q_\phi(z|x)}\frac{p_\theta(x,z)}{q_\phi(z|x)}$$

$$\geq E_{z\sim q_\phi(z|x)}\log\frac{p_\theta(x,z)}{q_\phi(z|x)} = E_{z\sim q_\phi(z|x)}\log\frac{p_\theta(x|z)p_\theta(z)}{q_\phi(z|x)}$$

$$= E_{z\sim q_\phi(z|x)}\log p_\theta(x|z) - E_{z\sim q_\phi(z|x)}\log\frac{q_\phi(z|x)}{p_\theta(z)}$$

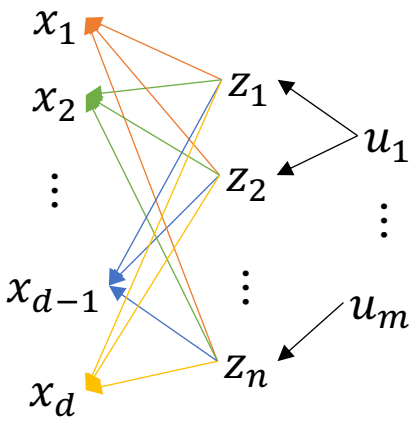
$$KL(q_\phi(z|x)||p_\theta(z))$$

Learning Objective:

$$\arg\max_{\phi,\theta}\mathbb{E}_{x\sim p(x)}[E_{z\sim q_\phi(z|x)}\log p_\theta(x|z) - KL(q_\phi(z|x)||p_\theta(z))]$$

Identifiable DGM

- Identifiable VAE(iVAE)



The conditional generative model is assume to be

$$p_\theta(x, z|u) = p_f(x|z)p_{T,\lambda}(z|u)$$

$$p_f(x|z) = p_\epsilon(x - f(z))$$

The conditional pdf is thus given by:

$$p_{T,\lambda}(z|u) = \prod_i \frac{Q_i(z_i)}{Z_i(u)} \exp\left[\sum_{j=1}^{k} T_{ij}(z_j)\lambda_{ij}(u)\right]$$

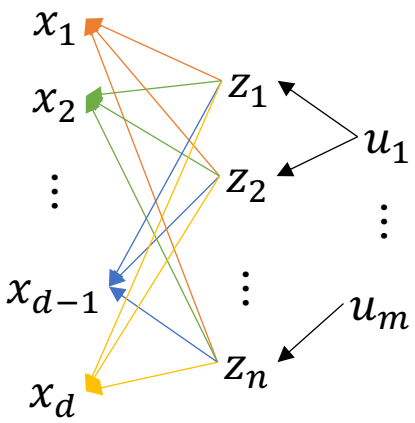
Then the ELBO for data log-likelihood is defined by:

$$\mathbb{E}_{\mathcal{D}}[\log p_\theta(x|u) \geq \mathcal{L}(\theta, \phi)] :=$$

$$\mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{q_\phi(z|x,u)}[\log p_\theta(x, z|u) - \log q_\phi(z|x, u)]\right]$$

Identifiable DGM

- Identifiable VAE(iVAE)



Definition:

$(f, T, \lambda) \sim (\tilde{f}, \tilde{T}, \tilde{\lambda}) \Leftrightarrow.$

$\exists\, A, c \mid T\left(f^{-1}(x)\right) = A\tilde{T}\left(\tilde{f}^{-1}(x)\right) + c, \forall x \in \mathcal{X}$

if A is invertible, we denote $\sim_A$-identifiable. If A is a block permutation matrix, we denote it by $\sim P$

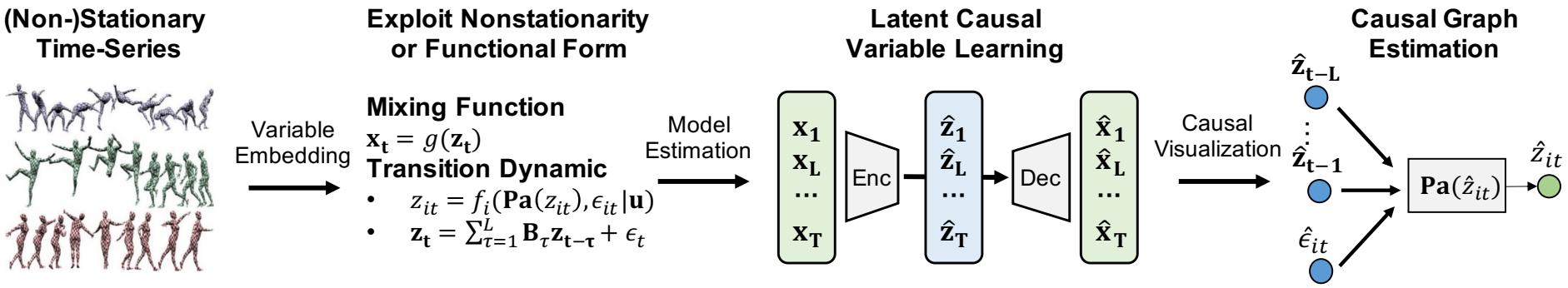
Assumptions for identifiability:

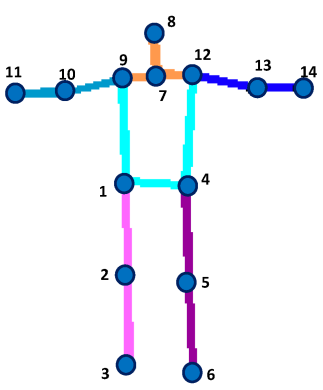
1. The sufficient statistics T_{ij} are differentiable almost everywhere and $\left(T_{ij}\right)_{1 \le j \le k}$ are linearly independent.
2. There exists $nk + 1$ distinct points $u^0, \cdots, u^{nk}$ such that the matrix
   $L = (\boldsymbol{\lambda}(u_1) - \boldsymbol{\lambda}(u_0), \cdots, \boldsymbol{\lambda}(u_{nk}) - \boldsymbol{\lambda}(u_0))$
   of size $nk \times nk$ is invertible.

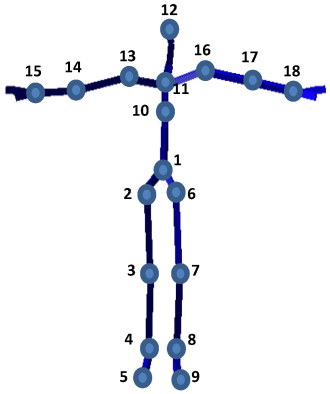
Then the parameters (f, T, λ) are $\sim_A$-identifiable.

Identifiable DGM

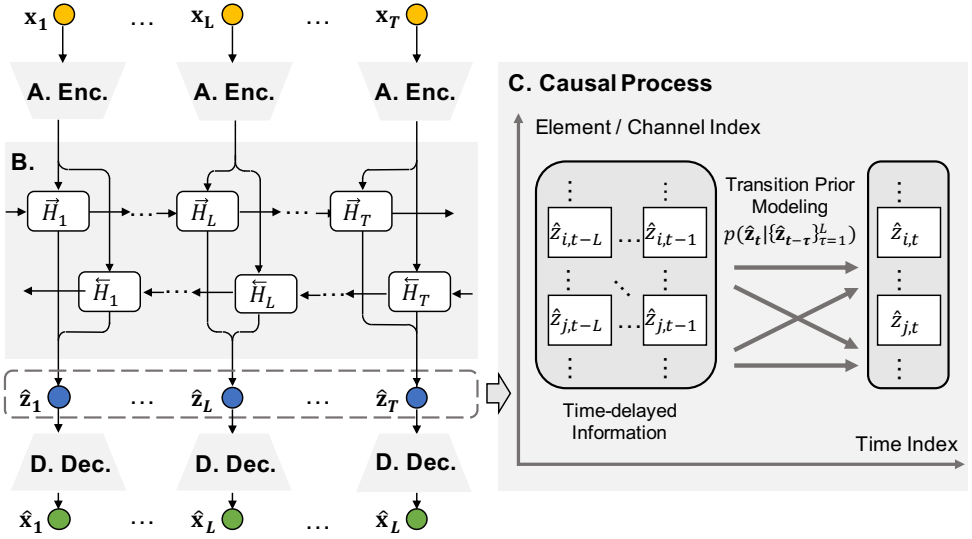
➢ Causality + Nonlinear ICA







(a) Human3.6M Skeleton (b) CMU/HDM05 Skeleton



More Works

- To unknown intrinsic dimension
  - Disentanglement by Nonlinear ICA with General Incompressible-flow Networks (GIN). [ICLR20]

- The sources are not conditional independent
  - ICE-BeeM: Identifiable Conditional Energy-Based Deep Models Based on Nonlinear ICA. [NeurIPS20]

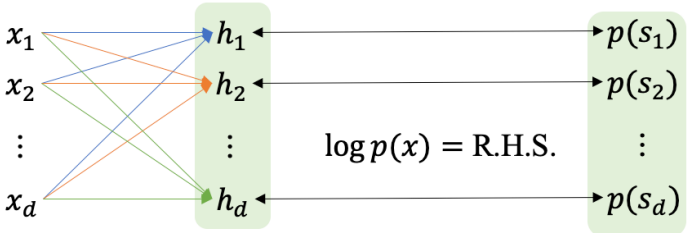
- $nk + 1$ distinct values of u are relaxed
  - Nonlinear ICA Using Volume-Preserving Transformations. [ICLR22]

- Don't need u
  - I Don't Need u: Identifiable Non-Linear ICA Without Side Information. [Paper]

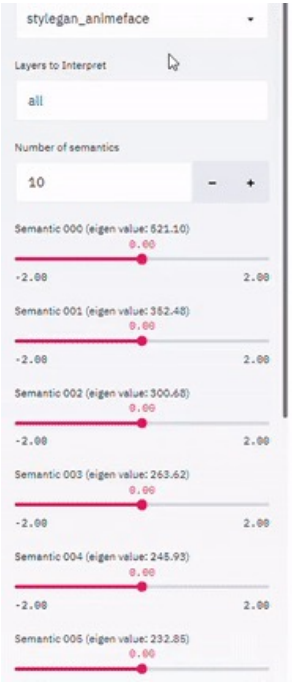
Invariant features separation

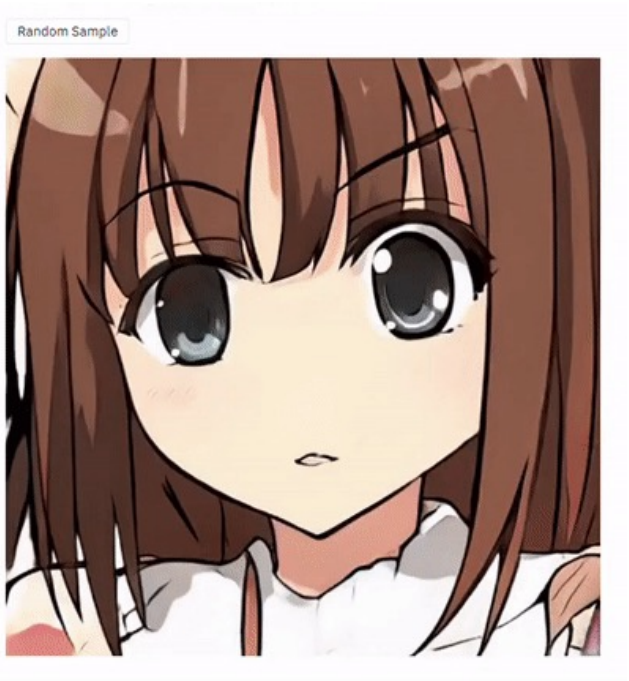
- ***Disentanglement***

$$x = f(s) \qquad \Rightarrow \qquad s = g(x)$$



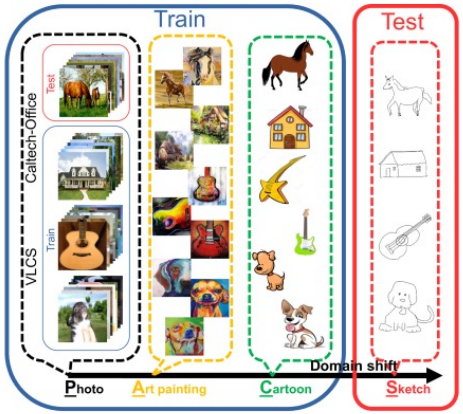
The aim is trying to recover the latent factors of the given observation.



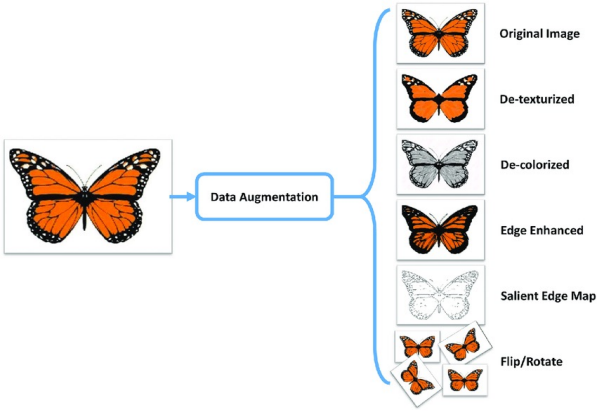


Closed-form factorization of latent semantics in GANs, CVPR21. [demo is cited from here]

Invariant features separation



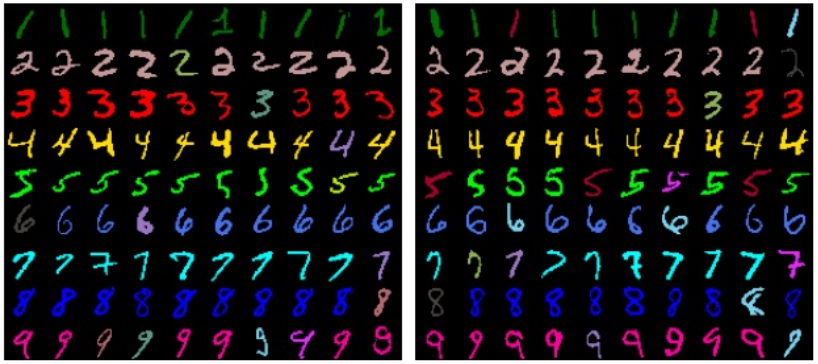
Domain shift



Data augmentation

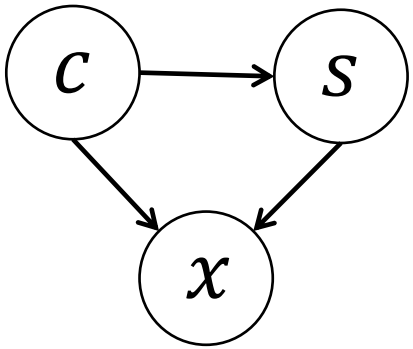
In many real senarios, one important aim is to learn a good (invariant sometimes) representation which can be well leveraged for the downstream tasks!

Notice that your task decides what are the good representation!



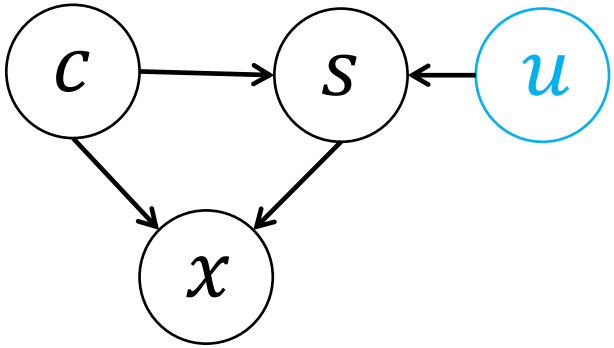
(a) T_r (b) T_g

Invariant features separation

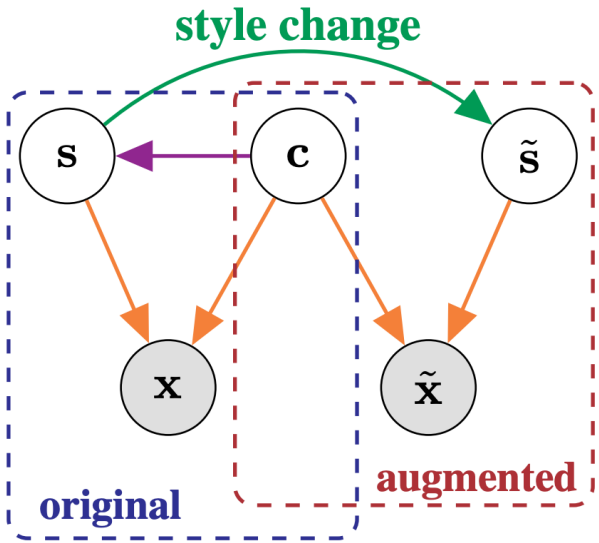


Data generation model

The the latent variables are divided into ***content*** c and ***style*** s, and allow for statistical and causal dependence of style on content. Assuming that only ***style changes*** among different domains.



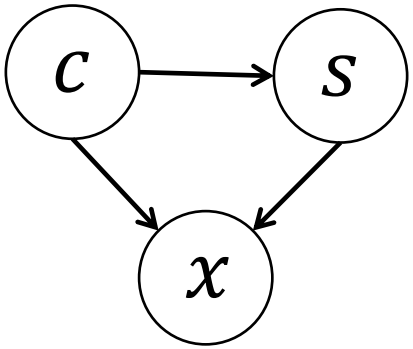
Domain adaptation/generalization

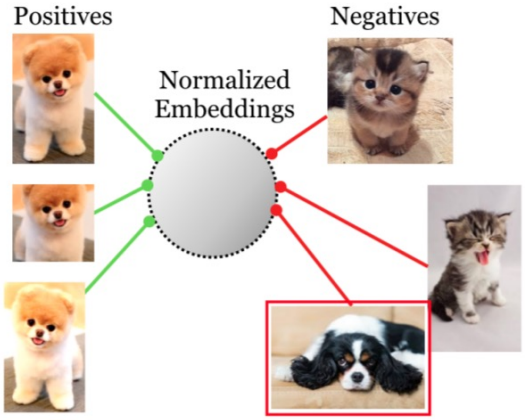


Domain augmentation

Invariant features separation

Task definition: Invariant features separation is defined to separate the content latent factors.





$$\mathcal{L}_{\text{InfoNCE}}(\mathbf{g};\tau,K)=\mathbb{E}_{\{\mathbf{x}_i\}_{i=1}^{K}\sim p_{\mathbf{x}}}\left[-\sum_{i=1}^{K}\log\frac{\exp\{\text{sim}(\tilde{\mathbf{z}}_i,\tilde{\mathbf{z}}_i')/\tau\}}{\sum_{j=1}^{K}\exp\{\text{sim}(\tilde{\mathbf{z}}_i,\tilde{\mathbf{z}}_j')/\tau\}}\right]$$

Motivation: Capturing the variations of s among different domains.
(If a factor s_i never changes among domains, then $s_i \in c$)

Assumptions: (1) The n_c content factors c stay invariant among different domains. (2) Some factors of n_s style factors s change in different domains.

Invariant features separation

$$\mathbf{z} \sim p_{\mathbf{z}}, \qquad \mathbf{x} = \mathbf{f}(\mathbf{z}). \tag{2}$$

$$A \sim p_A, \qquad \tilde{\mathbf{z}}|\mathbf{z}, A \sim p_{\tilde{\mathbf{z}}|\mathbf{z},A}, \qquad \tilde{\mathbf{x}} = \mathbf{f}(\tilde{\mathbf{z}}). \tag{3}$$

$A \subseteq \{1, ..., n_s\}$ is a subset which includes indexes of all changing latent factors in $\tilde{x}$

Theorem 4.2 (Identifying content with a generative model). *Consider the data generating process described in § 3, i.e., the pairs $(\mathbf{x}, \tilde{\mathbf{x}})$ of original and augmented views are generated according to (2) and (3) with $p_{\tilde{\mathbf{z}}|\mathbf{z}}$ as defined in Assumptions 3.1 and 3.2. Assume further that*

(i) $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ *is smooth and invertible with smooth inverse (i.e., a diffeomorphism);*

(ii) $p_{\mathbf{z}}$ *is a smooth, continuous density on* $\mathcal{Z}$ *with* $p_{\mathbf{z}}(\mathbf{z}) > 0$ *almost everywhere;*

(iii) for any $l \in \{1, ..., n_s\}$, $\exists A \subseteq \{1, ..., n_s\}$ *s.t.* $l \in A$; $p_A(A) > 0$; $p_{\tilde{\mathbf{s}}_A|\mathbf{s}_A}$ *is smooth w.r.t. both* $\mathbf{s}_A$ *and* $\tilde{\mathbf{s}}_A$; *and for any* $\mathbf{s}_A$, $p_{\tilde{\mathbf{s}}_A|\mathbf{s}_A}(\cdot|\mathbf{s}_A) > 0$ *in some open, non-empty subset containing* $\mathbf{s}_A$.

If, for a given n_s ($1 \leq n_s < n$), a generative model $(\hat{p}_{\mathbf{z}}, \hat{p}_A, \hat{p}_{\tilde{\mathbf{s}}|\mathbf{s},A}, \hat{\mathbf{f}})$ assumes the same generative process (§ 3), satisfies the above assumptions (i)-(iii), and matches the data likelihood,

$$p_{\mathbf{x},\tilde{\mathbf{x}}}(\mathbf{x}, \tilde{\mathbf{x}}) = \hat{p}_{\mathbf{x},\tilde{\mathbf{x}}}(\mathbf{x}, \tilde{\mathbf{x}}) \qquad \forall (\mathbf{x}, \tilde{\mathbf{x}}) \in \mathcal{X} \times \mathcal{X},$$

then it block-identifies the true content variables via $\mathbf{g} = \hat{\mathbf{f}}^{-1}$ in the sense of Defn. 4.1.

Theorem 4.3 (Identifying content with an invertible encoder). *Assume the same data generating process (§ 3) and conditions (i)-(iv) as in Thm. 4.2. Let* $\mathbf{g} : \mathcal{X} \to \mathcal{Z}$ *be any smooth and* invertible *function which minimises the following functional:*

$$\mathcal{L}_{\text{Align}}(\mathbf{g}) := \mathbb{E}_{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathbf{x},\tilde{\mathbf{x}}}} \left[\left|\left| \mathbf{g}(\mathbf{x})_{1:n_c} - \mathbf{g}(\tilde{\mathbf{x}})_{1:n_c} \right|\right|_2^2 \right] \tag{4}$$

Then $\mathbf{g}$ *block-identifies the true content variables in the sense of Definition 4.1.*

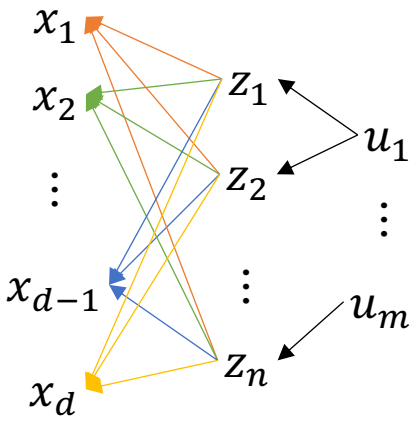
However, the applicability of FLOW is limited as it places strong constraints on the encoder architecture which makes it hard to scale these methods up to high-dimensional settings

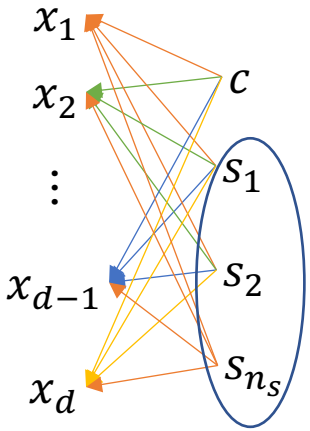
Theorem 4.4 (Identifying content with discriminative learning and a non-invertible encoder). *Assume the same data generating process (§ 3) and conditions (i)-(iv) as in Thm. 4.2. Let* $\mathbf{g} : \mathcal{X} \to (0,1)^{n_c}$ *be any smooth function which minimises the following functional:*

$$\mathcal{L}_{\text{AlignMaxEnt}}(\mathbf{g}) := \mathbb{E}_{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathbf{x},\tilde{\mathbf{x}}}} \left[\left|\left| \mathbf{g}(\mathbf{x}) - \mathbf{g}(\tilde{\mathbf{x}}) \right|\right|_2^2 \right] - H\left(\mathbf{g}(\mathbf{x})\right) \tag{5}$$

where $H(\cdot)$ *denotes the differential entropy of the random variable* $\mathbf{g}(\mathbf{x})$ *taking values in* $(0,1)^{n_c}$. *Then* $\mathbf{g}$ *block-identifies the true content variables in the sense of Defn. 4.1.*

More discussions





You need ***capture changes*** of latent factors.

How to decide the number of changes?? (1) for identifiability, it relates to the statistical sufficiencies of factors you care about. (2) for seperation, for each factor you want to exclude, you need have one independent change of it.

Thank you!