Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets 해석 및 리뷰 및 내용추가 정리

-by puostyoon(https://github.com/puostyoon)

이 논문에 대해 잘 설명해놓은 ppt:

https://www.slideshare.net/ssuser06e0c5/infogan-interpretable-representation-learning-by-information-maximizing-generative-adversarial-nets-72268213

잘 설명해놓은 블로그:

https://haawron.tistory.com/10

1. Abstract

GAN의 information-theoretic extension(GAN에 정보이론의 아이디어를 추가한 것)인 InfoGAN은 완전히 unsupervised 방식으로, 구분된(disentangled) representations을 학습할 수 있다. InfoGAN은 observation(gan에서 생성된 이미지)와 small subset of the latent variables사이의 mutual information을 최대화하는 방식의 GAN이다. 우리는 효율적으로 optimized 될 수 있는 mutual information objective의, lower bound를 derive한다. 구체적으로 InfoGAN은 MNIST dataset의 digit shapes로부터의 writing styles를 성공적으로 구분(disentangles)하고, 3D rendered images의 lighting(조명)으로부터 pose를 분리하고, SVHN dataset의 central digit으로부터 background digits를 분리한다. 또한 InfoGAN은 hair styles, presence/absence of eyeglasses, 그리고 CelebA face dataset으로부터의 emotions와 같은 visual concepts를 발견한다. Experiments는 InfoGAN이, 현존하는 supervised method로 학습한 representations에 맞먹는 interpretable representation을 학습한다는 것을 보여준다.

2. Introduction

Unsupervised learning은 대량으로 존재하는 unlabelled data로부터 value를 extracting하는 general problem으로 볼 수 있다. Unsupervised learning에 대한 유명한 framework는, representation learning의 framework이다 [1,2]. 이들의 목표는 unlabeled data를 사용해서, 중요한 의미적 특징(semantic features)들을 easily decodable factor로 exposes하는 representation을 학습하는 것이다. 그러한 representations를 학습할 수 있는 방법은 존재할 수 있다. 그리고 이 방법은 classification, regression, visualization, policy learning in reinforcement learning을 포함하는 다양한 downstream tasks에 유용할 수 있다.

Unsupervised learning은 좋지 않다(ill-posed). 왜냐하면 relevant downstream tasks들이 training time에서 알려지지 않은 상태이기 때문이다. 반면 disentangled representation은, data instance의 salient attributes(핵심 속성)을 명확히 represents하므로 'relevant but unknown tasks'에 효과적이다.

예를 들어, 얼굴 데이터셋에서, useful disentangled representation은 다음의 특징들: 'facial expression, eye color, hairstyle, presence or absence of eyeglasses, and the identity of the corresponding person' 각각에 대한 separate set of dimensions(별도의 차원 집합)를 allocate할 수 있다. Face recognition, object recognition처럼 data의 salient attributes에 대한 지식을 필요하는 natural tasks에 있어서, A disentangled representation이 유용할 수 있다. Image 속의 빨간 픽셀로 적혀있는 숫자가 홀수인지 짝수인지를 결정하는 것 같은 것들을 목표로 하는 unnatural supervised tasks에 대해서는 그렇지 않다. (disentangled representation이 유용하지 않을 것이다.) 그러므로 unsupervised learning algorithm이 유용하기 위해선, unsupervised learning algorithm은 downstream classification tasks의 likely set에 대한 직접적인 노출 없이, downstream classification tasks의 likely set을 반드시 효과적으로 추측해야 한다.

Generative modelling이 unsupervised learning 연구의 상당한 부분을 이끌어가고있다. Observed data를 "create"하거나 synthesize할 수 있는 능력이 어떠한 형태의 이해를 수반한다는 믿음이 generative modelling에 대한 연구를 motivate했다. 그리고 임의의 나쁜 representations을 가지는 prefect generative model을 만드는 것이 쉬움에도 불구하고, generative model은 스스로 disentangled representation을 학습할 것이라는 희망이 있었다. (원문: It is motivated by the belief that the ability to synthesize, or "create" the observed data entails some form of understanding, and it is hoped that a good generative model will automatically learn a disentangled representation, even though it is easy to construct perfect generative models with arbitrarily bad representations.) 가장 저명한 generative models에는 variational autoencoder (VAE) [3]와 generative adversarial network (GAN) [4] 가 있다.

이 논문에서 우리는 GAN objective에 대해 간단한 조정을 하여 interpretable and meaningful representations를 학습하도록 하는 것을 보인다. GAN의 noise variables의 fixed small subset과 relatively straightforward한 GAN의 목적함수 사이의 mutual information을 최대화함으로써 GAN objective를 조정한다. (원문: We do so by maximizing the mutual information between a fixed small subset of the GAN's noise variables and the observations, which turns out to be relatively straightforward.) 우리의 방법은 단순하지만, 우리의 방법이 놀랍게 효과적이라는 것을 발견했다: 우리의 방법은 다양한 이미지 데이터셋들 : digits(MNIST), faces(CelebA), house numbers(SVHN)에서의 highly semantic and meaningful hidden representations를 발견할 수 있었다. 우리의 unsupervised disentangled representation의 quality는 supervised label information을 사용했던 이전의 작업들[5-9]에 맞먹는다. 이 결과들은 mutual information cost와 합쳐진 generative modelling이 disentangled representations를 학습하는 알찬 접근방식이 될 수 있다는 것을 보여준다.

이 논문의 나머지 부분에서 우리는 related work를 review하고, 이전에 disentangled representation을 학습하는 방법들이 필요로 했던 supervision에 주목한다. 우리는 mutual information을 최대화하는 것이 어떻게 interpretable representation을 만들어내는지 describe하고, mutual information을 최대화하는 효과적인 알고리즘을 이끌어낼 것이다. 마지막으로 experiments section에서, 먼저 InfoGAN을, 비교적 clean datasets을 사용했던 이전의 접근방식들과 비교한다. 그리고 InfoGAN이 이전의 unsupervised 접근방식들이 학습한 representations와 비교할 수 없을

정도로 높은 quality로 interpretable representation을 학습할 수 있다는 것을 보여준다.

3. Related Work

Unsupervised representation learning에 대해 수많은 작업들이 존재한다. 이전의 방법들은 stacked (often denoising) autoencoders 혹은 restricted Boltzmann machines [10-13]들을 기반으로 했다. 많은 유망한 최근의 작업들은 skip-thought vectors를 inspire했던 Skip-gram model [14]과 unsupervised learning of images에 대한 몇몇 기술들[16]로부터 originate했다.

또다른 흥미로운 직종(line of work)에는, MNIST dataset을 이용한 semi-supervised variant를 가지고 놀라운 결과를 냈던 ladder network [17]가 있다. 더 최근엔, VAE에 기반한 model이 MNIST에서 훨씬 더 나은 semi-supervised results를 성취했다 [18]. GANs [4]는 Radford 외 다른 사람들에 의해 이용되어서 [19] code space에 대한 기본적인 linear algebra를 supports하는 image representation을 학습했다. Lake 외 다른 사람들은 [20] Bayesian programs에 대한 probabilistic inference를 이용하여 representations을 학습했고, 이는 OMNI dataset에서 설득력있는(convincing) one-shot results를 성취했다.

게다가, 이전의 연구들은 supervised data를 이용하여 disentangled representations을 학습하려 했었다. 그러한 방법 중에 하나는 supervised learning을 이용하여 representation의 subset이 supplied label과 일치하도록 학습하는 것이다: bilinear models [21]는 style과 content를 분리한다; multi-view perceptron [22]는 face identity와 view point를 분리한다; 그리고 Yang et al. [23]은 a sequence of latent factor transformations를 만들어내는 recurrent variant를 개발하였다. 이와 유사하게, VAEs [5]와 Adversarial Autoencoders [9] 는 class label이 다른 variations로부터 분리된 representations(representations in which class label is separated from other variations)를 학습하는 것을 보여주었다. (이하 related work의 내용들은, 저의 사전 지식이 부족하고 모르는 내용들과 모르는 내용이 너무 많아 일단 번역하지 않았습니다.)

Recently several weakly supervised methods were developed to remove the need of explicitly labeling variations. disBM [24] is a higher-order Boltzmann machine which learns a disentangled representation by "clamping" a part of the hidden units for a pair of data points that are known to match in all but one factors of variation. DC-IGN [7] extends this "clamping" idea to VAE and successfully learns graphics codes that can represent pose and light in 3D rendered images. This line of work yields impressive results, but they rely on a supervised grouping of the data that is generally not available. Whitney et al. [8] proposed to alleviate the grouping requirement by learning from consecutive frames of images and use temporal continuity as supervisory signal.

Unlike the cited prior works that strive to recover disentangled representations, InfoGAN requires no supervision of any kind. To the best of our knowledge, the only other unsupervised method that learns disentangled representations is hossRBM [13], a higher-order extension of the spike-and-slab restricted Boltzmann machine that can disentangle emotion from identity on the Toronto Face Dataset [25]. However, hossRBM can only disentangle discrete latent factors, and its computation cost grows exponentially in the number of factors. InfoGAN can disentangle both discrete and

continuous latent factors, scale to complicated datasets, and typically requires no more training time than regular GAN.

4. Background: Generative Adversarial Networks

Goodfellow et al. [4] 는 Generative adversarial networks (GAN)을 도입했다. GAN은 minmax game을 이용하여 deep generative models를 훈련하는 framework이다. 목표는 generator distribution $P_G(x)$ 이 실제 data distribution $P_{data}(x)$ 와 일치하도록 generator를 학습시키는 것이다. Data distribution의 모든 x에 확률을 명시적으로 assign하려고 하기보다, GAN은 noise variable z ~ $P_{noise}(z)$를 sample G(z)로 transforming 시킴으로써 generator distribution P_G로부터 samples를 만들어내는 generator network G를 학습한다. 이 Generator는 실제 data distribution인 P_{data}로부터 얻은 sample과 generator의 distribution인 P_G로부터 얻은 sample을 구분하는 것을 목표로 하는 adversarial discriminator network D에 대항함으로써 훈련된다. 더 형식적으로, 이 minmax game은 다음의 표현식으로 나타내진다:

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim P_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim \text{noise}}[\log(1 - D(G(z)))] \quad (1)$$

5. Mutual Information for Inducing Latent Codes

GAN formulation은 simple factored continuous input noise vector z를 사용한다. 이 때 generator가 이 noise를 사용하는 방식에 어떠한 제한도 두지 않는다. 결과적으로 the noise가 generator에 의해 highly entangled way(꼬여있는 방식, 분리되지 않은 방식)으로 사용되어 z의 각각의 dimensions이 data의 semantic feature에 대응하지 않게 된다.

그러나, 많은 domain들은 자연스럽게(naturally) variation의 의미있는 영역들로 분해된다.(gan은 z의 각 dimension이 semantic feature에 대응하지 않는데, 원래 자연스러운 방식은 domain들이 variation의 의미있는 영역들로 분해되는 것이다.) 예를 들어, MNIST dataset으로부터 images를 생성할 때, model이 자동적으로 discrete random variable이 숫자(0-9)의 numerical identity를 나타내도록 discrete random variable들을 할당하는 방식을 선택하게 된다면 이상적일 것이다. 그리고 model이 두 개의 additional(추가적인) continuous random variable이 숫자 획(stroke)의 angle과 thickness를 나타내도록 선택한다면 이상적이다. 이 attribute들은 independent이고, 눈에 띈다. 이 상황은, 우리가 어떠한 supervision없이 단순히 어떤 하나의 MNIST digit이 독립된 하나의 1-of-10 variable(10개중 한 개, 숫자가 10개이므로 어떤 MNIST digit 한 개는 숫자를 나타내는 variable 10개중 한 개에 의해 만들어짐)과 두 개의 독립된 continuous variables(두깨와 각)에 의해 만들어졌다는 것을 구체화함으로써 이러한 concepts(숫자가 1~10 중 어떠한 숫자인지, 숫자 획의 각, 두께는 어떤지)를 recover할 수 있는 상황인 것이다.

이 논문에서, 우리는 a single unstructured noise vector를 사용하는 대신 input noise vector를 두 개의 부분으로 decompose할 것을 제안한다: (i) incompressible noise의 source로 간주되는 z ; (ii) c, which we will call the latent code and will target the salient structured semantic features of the data distribution.

수학적으로, set of structured latent variables를 c1,c2,….,c_L 로 표기한다. 간단히 나타내자면, The set of structured latent variables는 $\mathrm{P}(\mathrm{c1}, \mathrm{c2}, \ldots, c_L) = \prod_{i=1}^{L} P(c_i)$ 로 표현되는 factored distribution으로 간주할 수 있다. 쉽게 표기하기위해서, 우리는 모든 latent variable c_i의 concatenation을 latent codes c로 표기할 것이다.

이제 이 latent factors를 unsupervised way로 발견하는 방법을 제시한다: generator network에 incompressible noise z와 latent code c 둘 다를 제공하여, generator가 G(z,c) 의 형태가 되도록 한다. 한편, standard GAN에선, generator가 $P_G(x|c) = P_G(x)$ 를 만족하는 solution을 찾아서 additional latent code를 얼마든지 무시할 수 있다. 이러한 trivial codes의 문제(additional latent code를 무시할 수 있게 되는 문제)에 대응하기 위해, information-theoretic regularization을 제공한다: latent codes c 그리고 generator distribution G(z,c) 사이에 high mutual information이 있어야 함. 그래서 $\mathrm{I}(\mathrm{c}; \mathrm{G}(\mathrm{z}, \mathrm{c}))$[1]가 높아질 것이다.

정보이론에서, X 와 Y 사이의 mutual information인 $\mathrm{I}(\mathrm{X}; \mathrm{Y})$는 random variable X에 대한 random variable Y의 지식으로부터 학습된 "정보의 양"을 측정한다. (원문: I(X; Y), measures the "amount of information" learned from knowledge of random variable Y about the other random variable X.) mutual information은 두 개의 entropy loss terms의 차로 표현될 수 있다:

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \qquad (2)$$

이 정의에 대한 직관적 해석은 다음과 같다: $\mathrm{I}(\mathrm{X}; \mathrm{Y})$는 Y가 관찰되었을 때 X에서의 불확실성의 감소를 뜻한다. (Y를 관찰하기 이전의 X의 불확실성과, Y가 관찰된 이후 X의 불확실성의 차이. 즉 Y를 관찰한 이후 X의 불확실성이 감소한 정도) X와 Y가 independent이면 $\mathrm{I}(\mathrm{X}; \mathrm{Y})=0$이다. 왜냐하면 one variable(X 혹은 Y 둘 중 하나)을 아는 것이 다른 하나에 대해 아무것도 알려주지 않기 때문이다; 반면, X와 Y가 deterministic, invertible function에 의해 연관되어 있을 때 mutual information이 최대가 된다. (deterministic function이란 같은 input에 대해 항상 같은 output을 제공하는 함수, invertible function은 일대일 함수를 뜻함 당연히 X와 Y 사이에 deterministic, one-to-one 관계가 성립한다면, X값만 가지고도 Y를 알 수 있으니 X와 Y 사이의 mutual information이 최대가 된다.) 이 해석은 cost를 formulate하기 쉽게 해준다: $x \sim P_G(x)$ 인 어떠한 x가 주어졌을 때, 우리는 $P_G(c|x)$가 작은 entropy를 가지길 원한다. (우리가 원하는 것은 $\mathrm{I}(\mathrm{c}; \mathrm{G}(\mathrm{z}, \mathrm{c}))$ 값이 커지는 것이다. 그러므로 $x \sim P_G(x)$ 인 x가 주어졌을 때, 즉 $P_G(x)$ 값이 알려졌을때 c값이 정해지길 원한다. 즉 $x \sim P_G(x)$인 x가 주어졌을 때 $P_G(c|x)$의 불확실성(엔트로피)가 줄어들길 원한다. 저자가 $P_G(x)$와 x 를 헷갈리게 사용한 것 같다.) 다시 말해서, latent code c에 들어있는 정보는 generation process에서 lost 되어서는 안된다는 뜻이다. 유사한 mutual information에 영감을 받은 목적함수가 이전에 clustering의 맥락에서(in the context of clustering) 고려된 적 있다[26-28].

---

[1] Imformation theory에 나오는 기호들이다. 다음의 링크들을 참조하자: (https://ratsgo.github.io/statistics/2017/09/22/information/), (https://bskyvision.com/774), (https://datascienceschool.net/02%20mathematics/10.02%20%EC%A1%B0%EA%B1%B4%EB%B6%80%EC%97%94%ED%8A%B8%EB%A1%9C%ED%94%BC.html)

(Similar mutual information inspired objectives have been considered before in the context of clustering.) 그러므로 다음의 information-regularized minmax game을 해결할 것을 제안한다:

$$\min_G \max_D V_I(D,G) = V(D,G) - \lambda I(c; G(z,c)) \tag{3}$$

(V는 보통 목적함수를 나타내는 기호이다. 엔트로피나 정보량 같은 특별한 정보이론에서 사용되는 기호는 아니다. 람다는 learning rate 같은 계수 상수 같은 것일 것.)

6. Variational Mutual Information Maximization

In practice, mutual information term $I(c; G(z,c))$ 를 직접적으로 maximize하는 것은 힘들다. 왜냐하면 $I(c; G(z,c))$ 를 directly maximize하려면 posterior $P(c|x)$ 에 access해야 하기 때문이다. 다행히 auxiliary distribution $Q(c|x)$ 를 정의하여 $P(c|x)$ 를 근사하고, $I(c; G(z,c))$ 의 lower bound를 얻을 수 있다:

$$\begin{aligned} I(c; G(z,c)) &= H(c) - H(c|G(z,c)) \\ &= \mathbb{E}_{x\sim G(z,c)}[\mathbb{E}_{c'\sim P(c|x)}[\log P(c'|x)]] + H(c) \\ &= \mathbb{E}_{x\sim G(z,c)}[\underbrace{D_{\mathrm{KL}}(P(\cdot|x) \parallel Q(\cdot|x))}_{\geq 0} + \mathbb{E}_{c'\sim P(c|x)}[\log Q(c'|x)]] + H(c) \\ &\geq \mathbb{E}_{x\sim G(z,c)}[\mathbb{E}_{c'\sim P(c|x)}[\log Q(c'|x)]] + H(c) \end{aligned} \tag{4}$$

위 식의 세번째 줄에서 대괄호의 위치를 주의하자. 그리고 위 식에서, 아마도 · 에 들어갈 기호는 c′ 인 것 같다. 그리고 위 식에서 KL-divergence는 무조건 0이상이라는 KL divergence의 특징을 이용하였다(https://hyunw.kim/blog/2017/10/27/KL_divergence.html https://hyunw.kim/blog/2017/10/26/Cross_Entropy.html 참고). 그리고 위 식의 Expectation기호 E의 아랫첨자 x~G(z,c)가 뜻하는 바는 x가 G(z,c)의 분포를 따르고 있다는 것이다. 그래서 Expectation은 $\sum xP(x)$ 형태로 구하는데, 이때 x가 G(z,c)를 따르므로, P(x)의 x부분에 G(z,c)를 사용하겠다는 뜻이다. 그런데 c′~P(c|x) 같은 경우에선, c′가 P(c|x)의 분포를 따른다는 이야기가 아니라, expectation을 구할 때, 곱해지는 확률값으로 P(c|x)를 사용하겠다는 뜻이다. 그리고 사실 Expectation 기호는 적분이나 합(sigma 기호)를 간단히 나타내기 위해 남용되는 경우가 많으니 entropy의 정의를 사용해서 수식을 이해하면 되고, expectation기호는 적당히 받아들이면 된다. mutual information를 lower bounding하는 이 테크닉은 Variational Information Maximization [29] 라고 알려져있다. 추가로 우리는 latent codes의 entropy $H(c)$가 common distributions에 대해서 간단한 analytical form을 가지고 있기 때문에, $H(c)$ 또한 optimize될 수 있음에 주목했다. 하지만 이 논문에서 최적화할 때는, 단순함을 위해 latent code distribution을 고정시키고, $H(c)$를 상수로 취급할 것이다. 지금까지 posterior $P(c|x)$ 를 명시적으로 계산해야 한다는 문제를 이 lower bound를 이용하여 우회했다. 그러나 우리는 여전히 inner expectation의 posterior((4)번 식의 마지막 줄의 두 겹의 expectation기호 중 안쪽의 expectation을 말하는 듯)에서부터 sample할 줄 알아야만 한다. 다음으로, posterior로부터 sample할 필요를 없애주는 간단한 lemma를 서술하겠다. 이 lemma의 증명은 Appendix에 defer to되어있다(증명을 Appendix로 미뤘다는 뜻).

Lemma 5.1 *For random variables X, Y and function $f(x, y)$ under suitable regularity conditions:* $\mathbb{E}_{x\sim X, y\sim Y|x}[f(x, y)] = \mathbb{E}_{x\sim X, y\sim Y|x, x'\sim X|y}[f(x', y)]$.

Appendix에 나와있는 증명:

Proof

$$\begin{aligned}
\mathbb{E}_{x\sim X, y\sim Y|x}[f(x, y)] &= \int_x P(x) \int_y P(y|x) f(x, y) dy dx \\
&= \int_x \int_y P(x, y) f(x, y) dy dx \\
&= \int_x \int_y P(x, y) f(x, y) \int_{x'} P(x'|y) dx' dy dx \\
&= \int_x P(x) \int_y P(y|x) \int_{x'} P(x'|y) f(x', y) dx' dy dx \\
&= \mathbb{E}_{x\sim X, y\sim Y|x, x'\sim X|y}[f(x', y)]
\end{aligned} \tag{7}$$

Lemma A.1 (Lemma 5.1)을 사용해서, mutual information $\mathrm{I}(\mathrm{c}; \mathrm{G}(\mathrm{z}, \mathrm{c}))$의 variational lower bound $L_I(G, Q)$를 정의할 수 있다. 사실 Lemma 5.1을 어떻게 사용한 건지는 잘 모르겠고, law of total expectation(https://www.youtube.com/watch?v=GnEyIawrWBg)을 사용했다고 생각하면 이해하기 쉽다. 또한 이해할 땐 https://github.com/sungreong/Infogan 이 링크의 수식설명부분을 참고했다. 링크에선 거창하게 설명했지만, 결국 $E_{x\sim G(z,c)}\left[E_{c'\sim P(c|x)}[logQ(c'|x)]\right] = \sum_x P(x) \sum_c P(c|x)\, logQ(c|x) = \sum_c \sum_x P(c)P(c|x) logQ(c|x) = \sum_c \sum_x P(c, x) logQ(c|x) = \sum_c \sum_x P(c)P(x|c)\, logQ(c|x) = \sum_c P(c) \sum_x P(x|c)\, logQ(c|x) = E_{c\sim P(c), x\sim G(z,c)}[logQ(c|x)]$ 이런 형태이다. 여기서, 왜 $\sum_c P(c) \sum_x P(x|c)\, logQ(c|x) = E_{c\sim P(c), x\sim G(z,c)}[logQ(c|x)]$ 냐고 따질 수 있는데, 일단 저 expectation기호가 잘못 표기된 것이든 뭐든 P(c|x)를 구할 필요가 없어졌다는 사실이 중요하다. 둘째로, x~G(z,c)이므로 $E_{c\sim P(c), x\sim G(z,c)}[logQ(c|x)]$에서 x에 대한 expectation을 구할 때 logQ(c|x)에 곱해지는 확률은 P(x|c)라고 생각할 수 있기 때문에 저 expectation은 제대로 쓰여진 기호라고 볼 수 있다.

$$\begin{aligned}
L_I(G, Q) &= E_{c\sim P(c), x\sim G(z,c)}[\log Q(c|x)] + H(c) \\
&= E_{x\sim G(z,c)}[\mathbb{E}_{c'\sim P(c|x)}[\log Q(c'|x)]] + H(c) \\
&\leq I(c; G(z, c))
\end{aligned} \tag{5}$$

우리는 Monte Carlo simulation을 이용하면 $L_I(G, Q)$을 쉽게 근사할 수 있다는 사실에 주목했다. 특히, L_I 은 Q에 대해선 직접적으로, 그리고 G에 대해선 reparametrization trick을 이용해서 최대화될 수 있다. (In particular, L_I can be maximized w.r.t. Q directly and w.r.t G via reparametrization trick.) 그러므로 GAN의 training procedure에 대한 변화 없이 GAN의 목적함수에 $L_I(G, Q)$ 이 더해질 수 있다. 그리고 우리는 최종 알고리즘(resulting algorithm)을 Information Maximizing Generative Adversarial Networks (InfoGAN)이라 부른다.

Eq (4) 는 auxiliary distribution Q가 true posterior distribution에 가까워짐에 따라 lower bound가 tight해진다는 것을 보여준다 : $E_x[D_{KL}(P(\cdot|x)||Q(\cdot|x))]$ ->0. ($P(\cdot|x)$와 $Q(\cdot|x)$의 분포가 같으면,

그렇게되면 $D_{KL}(P(\cdot|x)||Q(\cdot|x))$ 의 값이 0이 된다. 그렇게 되면 Eq (4) 의 lower bound가 tight해진다.) 게다가, discrete latent codes에 대해 variational lower bound가 최대가 될 때, 즉 $L_I(G,Q) = H(c)$ 일 때, the bound는 tight해지고 the maximal mutual information이 얻어진다. (왜냐하면 $Q(c'|x)$가 확률값이므로 0~1의 값을 가진다. $\log Q(c'|x)$는 음수가 되므로 lower bound L_I의 최댓값은 $H(c)$이다.) Appendix에서, 어떻게 InfoGAN이 Wake-Sleep algorithm [30]과 연관하여 다른 방식으로 해석될 수 있는지에 주목했다.

그러므로 InfoGAN은 following minmax game with a variational regularization of mutual information and a hyperparameter λ 로 정의된다:

$$\min_{G,Q} \max_{D} V_{\text{InfoGAN}}(D,G,Q) = V(D,G) - \lambda L_I(G,Q) \qquad (6)$$

(위 식(6) 에서의 V(D,G)는 식(1) 에 나와있는 원본 GAN의 목적함수라고 생각해도 될 것이다. 식(6)에서 G,Q는 $V_{InfoGAN}$를 최소화시키려 한다. 그러려면 mutual information의 lower bound L_I을 크게 만들어야 한다. 즉 위의 minmax game에서 G는 mutual information을 크게 하는 방식으로 학습한다는 뜻이다.)

7. Implementation

In practice, 우리는 auxiliary distribution Q를 neural network로써 parametrize한다. 대부분의 실험(experiments)에서, Q와 D는 모든 convolutional layers를 공유하고, conditional distribution $Q(c|x)$ 에 대해선 output parameter를 만들기 위한 한 개의 최종 fully connected layer(affine layer를 뜻하는 듯)만 있다. (In most experiments, Q and D share all convolutional layers and there is one final fully connected layer to output parameters for the conditional distribution Q(c|x).) 그리고 이는 즉 InfoGAN이 GAN에 대해 무시할만한 수준의 계산만을 더 한다는 것이다. 또한 $L_I(G,Q)$가 항상 normal GAN obejctives보다 빨리 converge한다는 것을 발견했다. 그래서 InfoGAN은 essentially comes for free with GAN(GAN과 공짜로 딸려나온다.).

Categorical latent code c_i에 대해, 우리는 $Q(c_i|x)$를 represent하기 위해 natural choice of softmax nonlinearity를 이용한다. Continuous latent code c_j에 대해선, true posterior $P(c_j|x)$가 뭔지에 따라서 더 많은 옵션이 있다. (there are more options depending on what is true posterior $P(c_j|x)$.) 우리의 experiments에서, 단순히 $Q(c_j|x)$를 factored Gaussian으로 간주하는 것으로 충분하다는 것을 알아냈다

비록 InfoGAN이 extra hyperparameter λ를 도입하지만, 이 하이퍼파라미터는 tune하기 쉽고, discrete latent codes에 대해선 1로 설정해주는것으로 충분하다. Latent code가 continuous variables를 포함할 땐 더 작은 λ가 일반적으로 사용된다. 왜냐하면 이때는 $\lambda L_I(G,Q)$가 differential entropy를 포함하기 때문에, $\lambda L_I(G,Q)$가 GAN objectives와 같은 scale에 있도록 하기 위함이다.

GAN이 train하기 어렵다고 알려졌기 때문에, DC-GAN[19]에서 소개된 기술을 바탕으로 우리의 experiments를 설계한다. 이는 InfoGAN training을 stablize하기 충분해서 우리는 새 trick을 도입할 필요가 없었다. 자세한 experimental setup은 Appendix에 나와있다.

8. Experiments

우리 experiments의 첫 번째 목표는 mutual information이 효과적으로 maximize될 수 있는지를 조사해보는 것이다. 두 번째 목표는 생성기에서 한번에 하나의 latent factor만을 vary했을 때 그 factor를 varying하는 것이 generated images에서 어떤 하나의 semantic variation만을 초래하는지를 평가하기 위해서 generator의 한 개의 latent factor를 varying함으로써, InfoGAN이 disentangled and interpretable representations을 학습할 수 있는지를 평가하는 것이다. (The second goal is to evaluate if InfoGAN can learn disentangled and interpretable representations by making use of the generator to vary only one latent factor at a time in order to assess if varying such factor results in only one type of semantic variation in generated images.) DC-IGN [7] 또한 이 방법을 이용하여 3D image datasets에 대한 그들의 learned representations를 평가하였다.

8.1 Mutual Information Maximization

제안된 방법(proposed method)을 이용해서 Latent code c와 generated images G(z,c) 사이의 mutual information이 효과적으로 최대화될 수 있는지를 평가하기 위해, 우리는 latent codes c~Cat(k=10, p=0.1)[2]에 대한 uniform categorical distribution을 가지는 MNIST dataset에서 InfoGAN을 훈련시킨다. (To evaluate whether the mutual information between latent codes c and generated images G(z,c) can be maximized efficiently with proposed method, we train InfoGAN on MNIST dataset with a uniform categorical distribution on latent codes c~Cat(K=10, p=0.1).) Fig 1에서, the lower bound $L_I(G,Q)$이 빠르게 H(c) ≈ 2.30으로 최대화되었다. 이는 the bound (4)가 tight하며, maximal mutual information이 이루어졌음을 뜻한다. ($L_I(G,Q)$의 최댓값은 H(c) 이고 이 값은 2.30인데, figure1을 보면 $L_I(G,Q)$이 빠른 속도로 2.30으로 다가가고있다. 즉 식 (4) 에서 구한 lower bound가 최댓값과 상당히 가까운 lower bound이고 그러므로 tight하다는 뜻이다.)

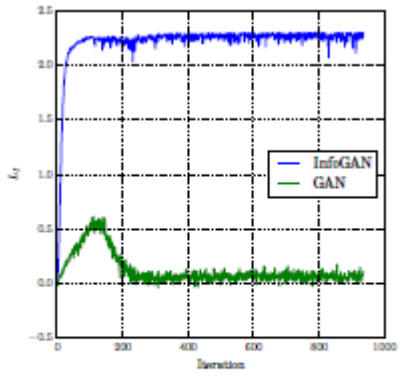


Figure 1: Lower bound L_I over training iterations

As a baseline(기준선으로, 즉 비교대상으로써), 우리는 또한 auxiliary distribution Q를 가진 regular

[2] Cat은 categorical distribution의 표기법이다. 다음 링크 참조: https://en.wikipedia.org/wiki/Categorical_distribution

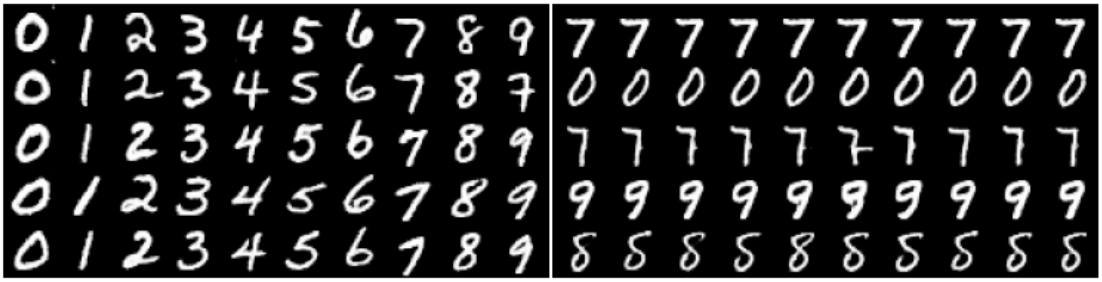
GAN을 학습시킨다. 근데 이 regular GAN은 latent codes와의 mutual information을 maximize하도록 encourage되지는 않는다. Q를 parametrize하기 위해 expressive neural network를 사용했기 때문에, Q가 reasonably approximates the true posterior P(c|x) 라고 가정할 수 있다. (식 (4)에서, P(c|x)를 approximate하기 위해 Q distribution을 도입했다는 점을 생각해보자.) 그리고 Q를 parametrize하기 위해 expressive neural network를 사용했기 때문에, latent codes와 regular GAN에서 generate된 image사이에 약간의 mutual information이 있다고 가정할 수 있다. (Since we use expressive neural network to parametrize Q, we can assume that Q reasonably approximates the true posterior P(c|x) and hence there is little mutual information between latent codes and generated images in regular GAN.) 다양한 neural network architecture로 실험해봤을 때 latent codes와 generated images사이에 더 높은 mutual information이 있을 수 있다는 점에 주목했다. 하지만 우리의 실험에선 그런 경우를 찾아보진 못했다. 이 비교는 regular GAN에서, generator가 latent codes를 사용하리라는 보장이 없다는 것을 보여주기 위함이다. (fig 1을 보면, regular gan에선 mutual information I의 lower bound인 L이 0이 되어버린다. 즉 mutual information이 있음-즉 latent codes를 사용함-에 대한 보장이 없음을 보여주고 있다.)

8.2 disentangled Representation

Mnist에서 digit shape styles을 분리(disentangle)하기 위해 data의 discontinuous variation을 model할 수 있는 한 개의 categorical code, $c_1 \sim Cat(K = 10, p = 0.1)$ 과 본질적으로 continuous한 variations(variations that are continuous in nature)을 capture하기 위한 두 개의 continuous codes: $c_2, c_3 \sim Unif(-1,1)$ [3]을 가지고 latent codes를 model하기로 선택했다.

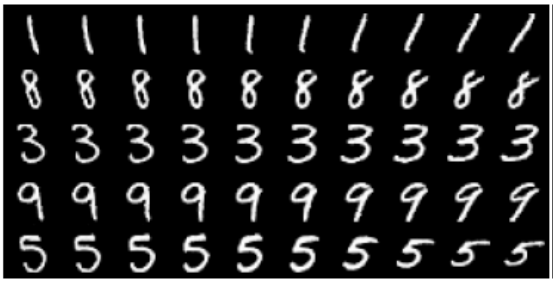
Figure 2에서, discrete code c1이 shape에서의 drastic change를 captures함을 보인다. Categorical code c1을 변경하는 것은 대부분의 경우에서(most of the time) 숫자를 변경하였다.(Changing categorical code c1 switches between digits most of the time.) In fact, label 없이(without any label) InfoGAN을 훈련해도, c1의 각각의 category를 하나의 digit type에 matching함으로써 c1은 MNIST digits을 5%의 에러율을 가지고 classifying하는 classifier로써 사용될 수 있다. Figure 2a의 두 번째 행에서, digit 7이 9로 classify 된 것을 관찰할 수 있다.

[3] Unif기호는 uniform distribution의 표기법인 것 같다. -1부터 1까지 0.5의 확률값을 가지고(범위 길이가 2이므로)나머지 구간에선 0의 확률값을 가지는 uniform distribution을 표기한 것 같다.

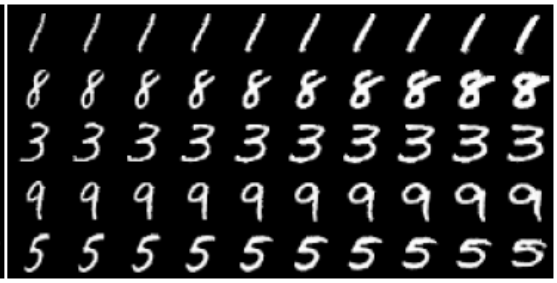


(a) Varying c_1 on InfoGAN (Digit type)

(b) Varying c_1 on regular GAN (No clear meaning)



(c) Varying c_2 from -2 to 2 on InfoGAN (Rotation)



(d) Varying c_3 from -2 to 2 on InfoGAN (Width)

Figure 2: **Manipulating latent codes on MNIST:** *In all figures of latent code manipulation, we will use the convention that in each one latent code varies from left to right while the other latent codes and noise are fixed. The different rows correspond to different random samples of fixed latent codes and noise. For instance, in (a), one column contains five samples from the same category in c_1, and a row shows the generated images for 10 possible categories in c_1 with other noise fixed.* In (a), each category in c_1 largely corresponds to one digit type; in (b), varying c_1 on a GAN trained without information regularization results in non-interpretable variations; in (c), a small value of c_2 denotes left leaning digit whereas a high value corresponds to right leaning digit; in (d), c_3 smoothly controls the width. We reorder (a) for visualization purpose, as the categorical code is inherently unordered.

Continuous code c2,c3은 style에서의 continuous variations을 capture한다: c2는 숫자의 rotations을 models하고 c3는 width를 control한다. 놀라운점은, 두 경우들에서 generator는 단순히 digits을 stretch하거나 rotate하지 않고, 결과 이미지들이 자연스럽게 보이게끔 thickness나 stroke style과 같은 다른 details를 adjust한다는 것이다. InfoGAN에 의해 학습된 latent representation이 generalizable가능한지를 확인하기 위해서, latent codes를 exaggerated 방식으로 maipulate하였다: latent codes를 -1 에서 1까지 plotting하는 대신, network가 trained on 해본 적 없는 더 넓은 영역을 covering하는 -2부터 2까지에서 latent codes를 plotting했는데 여전히 의미있는 generalization을 얻을 수 있었다.

다음으로 우리는 두 개의 3D images datasets에서 InfoGAN을 evaluate하였다: faces[31] and chairs[32], 그리고 DC-IGN은 이 두 개의 datasets에서 highly interpretable graphics codes를 보였었다.

Faces Datasets에서, DC-IGN은 supervision을 사용해서 azimuth (pose), elevation, and lighting과 같은 latent factors들을 continuous latent variables을 이용해서 represent하는 방법을 학습했다. 같은 Dataset을 사용해서, InfoGAN이 same dataset에서의 azimuth (pose), elevation, and lighting을 recover하는 disentangled representation을 학습한다는 것을 보인다. 이번 experiments에서, five continuous codes $c_i \sim Unif(-1,1), (1 \leq i \leq 5)$를 가지고 latent codes를 model하기로 선택했다.

DC-IGN이 supervision을 필요로 하기 때문에 예전에는, unlabeled였던 variation에 대한 latent code를 학습할 수 없어서 data로부터 자동적으로 salient latent factors of variation을 발견할 수 없었다. 반면 InfoGAN은 스스로 그러한 variation을 발견할 수 있다: 예를 들어, Figure 3d에서, wide에서 narrow로 face를 smoothly change하는 latent codes가 학습되었는데, 심지어 이 variation은 prior work(DC-IGN에서의 work)에서 explicitly generated되지도 않았고, labeled되지도 않았었다. (즉 DC-IGN에서 data를 labelling하여 학습시킬 때 pose, elevation, lighting같은 것들을 label에서 학습시켰고, 이 땐 pose, elevation, lighting을 관장하는 latent factor를 학습할 수 있었다. 근데 이때도 wide, narrow에 관한 latent factor를 label해서 explicitly 생성한 적은 없었는데, 이번에 labelling 하지 않고 data로부터 신경망이 스스로 latent factor들을 알아내게 하는 방식의 학습에서 wide, narrow라는 새로운 variation을 신경망이 스스로 발견한 것이다.)

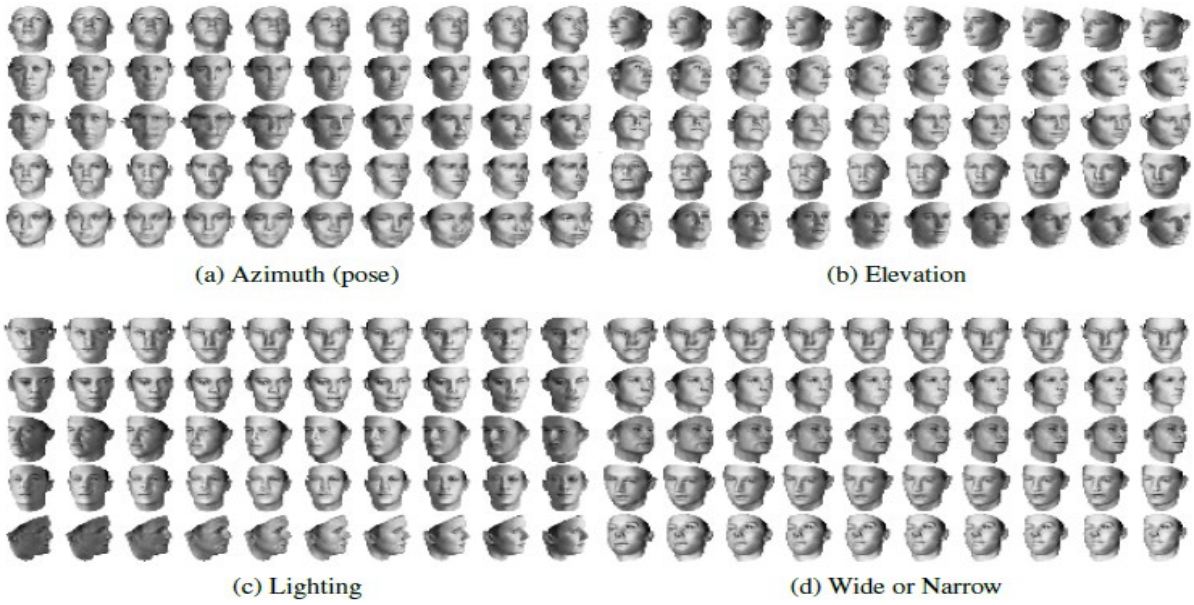


Figure 3: **Manipulating latent codes on 3D Faces:** We show the effect of the learned continuous latent factors on the outputs as their values vary from -1 to 1. In (a), we show that one of the continuous latent codes consistently captures the azimuth of the face across different shapes; in (b), the continuous code captures elevation; in (c), the continuous code captures the orientation of lighting; and finally in (d), the continuous code learns to interpolate between wide and narrow faces while preserving other visual features. For each factor, we present the representation that most resembles prior supervised results [7] out of 5 random runs to provide direct comparison.

Chairs dataset에서, DC-IGN은 rotation을 represents 하는 continuous code를 학습할 수 있다. InfoGAN은 또한 the same concept(rotation)을 continuous code로써 learn할 수 있다 (Figure 4a). 그리고 우리는 InfoGAN이 하나의 continuous code를 이용하여 다양한 widths를 지닌 비슷한 chair types 사이를 continuously interpolate할 수 있음을 보인다 (Figure 4b). 이 실험에서 우리는 4개의 categorical code c_1, c_2, c_3, c_4 ~Cat(K=20, p=0.05), 그리고 한 개의 continuous code c_5 ~ Unif(-1, 1)를 가지고 latent factors를 model하기로 선택하였다.

(a) Rotation (b) Width

Figure 4: **Manipulating latent codes on 3D Chairs:** In (a), we show that the continuous code captures the pose of the chair while preserving its shape, although the learned pose mapping varies across different types; in (b), we show that the continuous code can alternatively learn to capture the widths of different chair types, and smoothly interpolate between them. For each factor, we present the representation that most resembles prior supervised results [7] out of 5 random runs to provide direct comparison.

다음으로 우리는 Street View House Number (SVHN) dataset에서 InfoGAN을 테스트한다. SVHN dataset은 noisy하고, variable-resolution and distracting digits를 가지며, 같은 object에 대한 multiple variations을 가지지 않기 때문에 SVHN dataset에서 interpretable representation을 학습하는 것은 더 힘들다. 이 실험에서, 우리는 네 개의 10-dimensional categorical variables와 두 개의 uniform continuous variables을 latent codes로 사용하였다. Figure 5에서 학습된 latent factors 중 두 개를 보여주었다.

마지막으로 Figure 6에서, InfoGAN이 또다른 challenging(학습하기 힘든) dataset인 CelebA[33]에서 많은 visual concepts를 학습할 수 있음을 보였다. CelebA datasets은 다양한 pose variations와 background clutter가 있는 20만장의 유명인 사진 dataset이다. 이 dataset에서, latent variation을 각각이 10차원인 10개의 uniform categorical variables을 이용하여 model한다. (여기서 10차원인 uniform categorical variables라는 것은 한 variable이 각 category에 속할 확률이 0.1로 uniform한, 즉 동일한 categorical variable을 말하는 것 같다. 이러한 variable이 10개가 있다는 뜻이다.) 놀랍게도, 이 복잡한 dataset에서조차, InfoGAN은 3D images에서처럼(여기서 말하는 3D image는 아까의 얼굴, 의자 3d image 데이터셋을 말하는 것 같다.) azimuth를 recover할 수 있었다. 심지어 이 데이터셋에서 어떤 하나의 얼굴도 다양한 pose position으로 나타난 적도 없는데 말이다. (Surprisingly, even in this complicated dataset, InfoGan can recover azimuth as in 3D images even though in this dataset no single face appears in multiple post positions.) 게다가 InfoGAN은 안경의 presence or absence, hairstyles and emotion과 같은 다른 highly semantic variations을 disentangle할 수 있었다. 이는 어떠한 supervision없이도 상당한 수준(a level of)의 visual understanding이 얻어졌다는 것을 보여준다.

(a) Continuous variation: Lighting (b) Discrete variation: Plate Context

Figure 5: **Manipulating latent codes on SVHN**: In (a), we show that one of the continuous codes captures variation in lighting even though in the dataset each digit is only present with one lighting condition; In (b), one of the categorical codes is shown to control the context of central digit: for example in the 2nd column, a digit 9 is (partially) present on the right whereas in 3rd column, a digit 0 is present, which indicates that InfoGAN has learned to separate central digit from its context.

(a) Azimuth (pose) (b) Presence or absence of glasses

(c) Hair style (d) Emotion

Figure 6: **Manipulating latent codes on CelebA:** (a) shows that a categorical code can capture the azimuth of face by discretizing this variation of continuous nature; in (b) a subset of the categorical code is devoted to signal the presence of glasses; (c) shows variation in hair style, roughly ordered from less hair to more hair; (d) shows change in emotion, roughly ordered from stern to happy.

9. Conclusion

이 논문은 Information Maximizing Generative Adversarial Network(InfoGAN)이라 불리는 representation learning algorithm을 소개한다. 이전의 supervision을 필요로 했던 approaches와 대비해서, InfoGAN은 완전히 unsupervised이고, challenging datasets에 대해서 interpretable and disentangled representations를 학습한다. 게다가, InfoGAN은 top of GAN에서 negligible computation만을 추가로 요구하고 훈련시키기 쉽다. Representation을 induce하기 위해 mutual information을 사용한다는 핵심 아이디어는 유망한 future work 분야인 VAE [3]과 같은 다른

방법들에도 적용될 수 있다. 이 work의 또다른 가능성있는 extensions에는 다음과 같은 것들이 있다: learning hierarchical latent representations, improving semi-supervised learning with better codes [34], and using InfoGAN as a high-dimensional data discovery tool.

References

[1] Y. Bengio, "Learning deep architectures for ai," *Foundations and trends in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[2] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 8, pp. 1798–1828, 2013.

[3] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *ArXiv preprint arXiv:1312.6114*, 2013.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS*, 2014, pp. 2672–2680.

[5] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *NIPS*, 2014, pp. 3581–3589.

[6] B. Cheung, J. A. Livezey, A. K. Bansal, and B. A. Olshausen, "Discovering hidden factors of variation in deep networks," *ArXiv preprint arXiv:1412.6583*, 2014.

[7] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum, "Deep convolutional inverse graphics network," in *NIPS*, 2015, pp. 2530–2538.

[8] W. F. Whitney, M. Chang, T. Kulkarni, and J. B. Tenenbaum, "Understanding visual concepts with continuation learning," *ArXiv preprint arXiv:1602.06822*, 2016.

[9] A. Makhzani, J. Shlens, N. Jaitly, and I. Goodfellow, "Adversarial autoencoders," *ArXiv preprint arXiv:1511.05644*, 2015.

[10] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[11] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[12] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *ICLR*, 2008, pp. 1096–1103.

[13] G. Desjardins, A. Courville, and Y. Bengio, "Disentangling factors of variation via generative entangling," *ArXiv preprint arXiv:1210.5474*, 2012.

[14] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *ArXiv preprint arXiv:1301.3781*, 2013.

[15] R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *NIPS*, 2015, pp. 3276–3284.

[16] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *ICCV*, 2015, pp. 1422–1430.

[17] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," in *NIPS*, 2015, pp. 3532–3540.

[18] L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther, "Improving semi-supervised learning with auxiliary deep generative models," in *NIPS Workshop on Advances in Approximate Bayesian Inference*, 2015.

[19] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *ArXiv preprint arXiv:1511.06434*, 2015.

[20] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[21] J. B. Tenenbaum and W. T. Freeman, "Separating style and content with bilinear models," *Neural computation*, vol. 12, no. 6, pp. 1247–1283, 2000.

[22] Z. Zhu, P. Luo, X. Wang, and X. Tang, "Multi-view perceptron: A deep model for learning face identity and view representations," in *NIPS*, 2014, pp. 217–225.

[23] J. Yang, S. E. Reed, M.-H. Yang, and H. Lee, "Weakly-supervised disentangling with recurrent transformations for 3d view synthesis," in *NIPS*, 2015, pp. 1099–1107.

[24] S. Reed, K. Sohn, Y. Zhang, and H. Lee, "Learning to disentangle factors of variation with manifold interaction," in *ICML*, 2014, pp. 1431–1439.

[25] J. Susskind, A. Anderson, and G. E. Hinton, "The Toronto face dataset," Tech. Rep., 2010.

[26] J. S. Bridle, A. J. Heading, and D. J. MacKay, "Unsupervised classifiers, mutual information and 'phantom targets'," in *NIPS*, 1992.

[27] D. Barber and F. V. Agakov, "Kernelized infomax clustering," in *NIPS*, 2005, pp. 17–24.

[28] A. Krause, P. Perona, and R. G. Gomes, "Discriminative clustering by regularized information maximization," in *NIPS*, 2010, pp. 775–783.

[29] D. Barber and F. V. Agakov, "The IM algorithm: A variational approach to information maximization," in *NIPS*, 2003.

[30] G. E. Hinton, P. Dayan, B. J. Frey, and R. M. Neal, "The" wake-sleep" algorithm for unsupervised neural networks," *Science*, vol. 268, no. 5214, pp. 1158–1161, 1995.

[31] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *AVSS*, 2009, pp. 296–301.

[32] M. Aubry, D. Maturana, A. Efros, B. Russell, and J. Sivic, "Seeing 3D chairs: Exemplar part-based 2D-3D alignment using a large dataset of CAD models," in *CVPR*, 2014, pp. 3762–3769.

[33] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *ICCV*, 2015.

[34] J. T. Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," *ArXiv preprint arXiv:1511.06390*, 2015.