

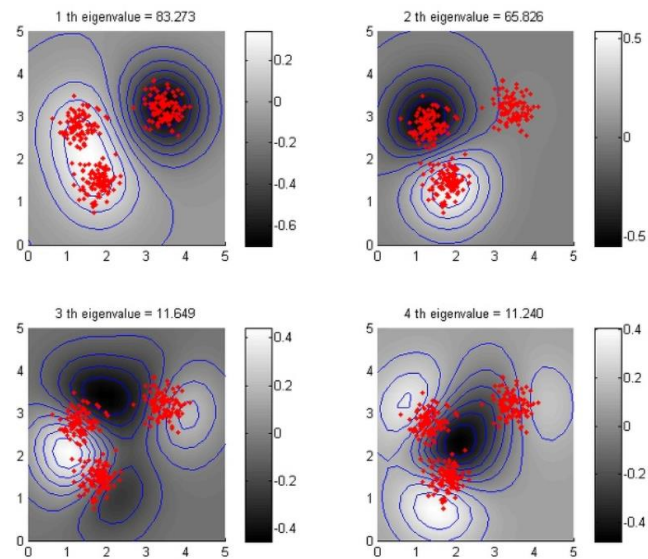
14 주차 조별보고서 (Default)	
작성일: 2019 년 12 월 6 일	작성자: 김영연
조 모임 일시: 2019 년 12 월 6 일 9 교시	모임장소: 학교 앞 카페
참석자: 위성조, 이충현, 최진성, 이재은, 김영연	조원: 위성조, 이충현, 최진성, 이재은, 김영연
구 분	내 용
학습 범위와 내용	<p>6.5절 공간 변환의 중요성</p> <p>6.6절 PCA, ICA, sparse 코딩</p> <p>6.7절 AUTO ENCODER</p> <p>6.8 절 매니폴드 개념과 isomap, LLE, T-SNE 매니폴드 학습기법</p>
논의 내용	<p>Q1.</p> <p>오토 인코더를 분류기로 활용하는 경우, CNN과 비교하여 어떤 차이가 있는지 알고 싶습니다.</p> <p>A1.</p> <p>영상처리 같은 경우에, CNN은 영상의 부분을 보고 특징을 추출한다. 그 특징이 쌓여서, 나중에 SoftMax로 가면 확률적으로 나온다. 오토 인코더는 입력 벡터를 출력 벡터로 복사하는 신경망인데 영상 전체의 중요한 부분까지 포함하게 하는 것이다. 또한 차원 축소를 통해 축소된 벡터로 classification한다.</p> <p>CNN 경우에도 결국 유사한 개념이나, 좀 더 공간적인 개념이 들어간다. CNN도 오토 인코더처럼 discriminant한 함수를 학습한다. 그러나 오토 인코더는 unsupervised pretraining이고 CNN은 supervised learning이다.</p>

Q2.

주성분 분석 중 데이터내에 비선형적 관계가 있으면 잘 작동하지 않는다고 한다. 이때의 해결방법에 대해 알고 싶다.

A2.

비선형적인 관계라면 Kernel PCA가 사용되어진다. Kernel PCA란 기존의 PCA에 Kernel trick을 적용시킨 것이다. Kernel trick은 기존 formulation의 내적 부분을 kernel function을 통해 만든 kernel matrix로 대체하는 것이다. PCA를 unsupervised learning에 적용할 경우, 주어진 데이터를 우리가 정한 basis에 projection시키고, 각 basis들에 project된 값이 주어진 데이터의 feature가 되는 구조이다. 이것에 kernel trick을 적용하였기 때문에 기존의 linear projection이 nonlinear한 projection으로 바뀌게 된다. 결과는 아래와 같다.



	<p>Q3.</p> <p>오토인코더를 이용하면 특징 추출과 분류를 따로 학습시키게 되므로 특징 추출과 분류를 한꺼번에 시키는 CNN 에 비해 성능이 더 좋을 것 같은데, 이 이론이 맞는지, 맞다면 대부분의 이미지 인식 네트워크에서 CNN 을 사용하는 특별한 이유가 있는지 알고 싶다.</p>
	<p>A3.</p> <p>AutoEncoder는 Unsupervised Pretraining 기법이다. 즉, 클래스 라벨로부터 역전파를 통해 학습하는 것이 아니라 입력 값을 출력 값으로 하여 재구성할 수 있는 특징을 찾아내는 과정이다. 이는 CNN도 유사한 개념이라고 할 수 있으나, CNN은 Supervised Training 형식이다.</p> <p>이전과 달리 vanishing gradient 문제를 해결한 ReLU의 등장과 Label된 데이터의 양이 증가하면서 Unsupervised Pretraining의 중요성이 감소하였고, 이후 CNN이 등장할 때에는 이미 데이터의 양이 충분했기 때문에 거의 대부분의 CNN이 Pretraining 없이 바로 Supervised Learning을 통한 학습이 가능하게 되었다.</p>
	<p>Q4.</p> <p>어떤 데이터셋에 적용한 차원 축소 알고리즘 성능은 어떻게 평가할 수 있나?</p>
	<p>A4.</p> <p>차원을 축소시키면 일부 정보가 유실된다. 그래서 훈련 속도는 빨라질 수 있으나 성능은 나빠진다. 또한 파이프라인이 조금 더 복잡해지고 유지관리가 어려워진다. 이러한 이유로 차원 축소를 고려하기 전에 원본 데이터의 훈련이 너무 느린지 먼저 훈련시켜봐야 한다. 그러나 어떤 경우에는 훈련 데이터의 차원을 축소시키면 잡음이나 불필요한 세부사항을 걸러내므로 성능을 높일 수 있다고 생각을 하는데 이는 훈련 속도가 빨라지기는 하지만 항상 더 좋거나 간단한 솔루션이 되지는 못한다. 따라서, 차원 축소 알고리즘의 성능은 전적으로 데이터셋에 달려 있다고 볼 수 있다.</p>

	<p>Q5.</p> <p>데이터셋의 차원을 축소시키고 나서 이 작업을 되돌리는 것이 가능한지 궁금합니다.</p> <hr/> <p>A5.</p> <p>결론적으로 말하면 완벽한 복원은 할 수 없지만, 어느 정도 복원이 가능하다</p> <p>대표적인 차원 축소 기법인 PCA(주성분분석)의 경우, 분산이 최대가 되며, 투영거리가 최소가 되는 초평면을 이용하여 차원을 축소시키게 되는데, 투영거리만큼의 데이터가 손실되므로, 완벽한 복원은 불가능하다. 그러나, 잘 수행된 차원 축소에서는 각 데이터를 복원하는데 가장 중요한 특징들을 추출하게 되므로, 복원된 데이터에는 기존 데이터를 다른 데이터와 구분하는데 필요한 특징들이 남아있게 된다.</p>
--	--

	<div data-bbox="701 202 1883 231" data-label="Text"> <p>Original High Dimension                      Latent Low Dimension                      Original High Dimension</p> </div> <div data-bbox="761 250 1816 948" data-label="Diagram"> <p style="text-align: center;"><b>Encoder                      Decoder</b> <b>(Dimension Reduction) (Reconstruction)</b></p> </div> <div data-bbox="553 1005 2033 1093" data-label="Text"> <p>Ex) 오토인코더의 경우 <math>z</math>영역까지 계속해서 차원을 축소시키지만, 이를 이용하여 원본 이미지에 근접한 이미지를 복원할 수 있음을 보여준다.</p> </div>
<p>질문 내용</p>	

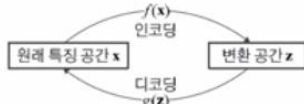
<첨부 개인 레포트>

<기계학습 14주차 레포트>

컴퓨터공학과

201402665 이충현

작성일: 2019.12.04

구분	내용
학습 범위	6.5절 공간 변환의 중요성 6.6절 PCA, ICA, sparse 코딩 6.7절 AUTO ENCODER 6.8절 매니폴드 개념과 isomap, LLE, T-SNE 매니폴드 학습기법
학습 내용	<p>&lt;6.5 절 공간 변환의 중요성&gt;</p> <p>공간변환 = 특징 공간을 극 좌표 공간으로 변환한 뒤 차원축소</p> <p>원래 공간을 다른 공간으로 변환 = 인코딩</p> <p>변환 공간을 원래 공간으로 변환 = 디코딩</p> <div data-bbox="1064 1034 1400 1204"><p><math>\hat{x} = g(f(x))</math></p><pre>graph LR; x[원래 특징 공간 x] -- "f(x) 인코딩" --&gt; z[변환 공간 z]; z -- "g(z) 디코딩" --&gt; x;</pre></div> <p>그림 6-17 공간 변환과 역변환</p> <p>데이터 압축 경우, 역변환으로 얻은 <math>\hat{x}</math>는 원래 신호 <math>x</math>와 가급적 같아야 한다.</p> <p>데이터 가시화의 경우, 2,3 차원의 <math>z</math> 공간으로 변환한다. 디코딩은 불필요</p>

### <6.6 절 선형 인자 모델>

선형 연산(scale linearity)을 이용한 공간 변환 기법

선형 연산을 사용함으로써 행렬 곱으로 인코딩, 디코딩 과정 표현가능.

$$f: \mathbf{z} = \mathbf{W}_{enc}\mathbf{x} + \boldsymbol{\alpha}_{enc}$$

$$g: \mathbf{x} = \mathbf{W}_{dec}\mathbf{z} + \boldsymbol{\alpha}_{dec}$$

A 는 노이즈나 데이터를 원점으로 이동하는 역할.

Z 에 확률 개념이 없고, a 를 생략하면 PCA-관찰 벡터 x 와 인자 z 는 결정론 적인 1:1 매핑 관계

주성분 분석(PCA) = z 와 a 가 가우시안 분포를 따른다 고 가정할 때 사용.

데이터를 원점 중심으로 옮기는 전처리를 먼저 수행->변환식 사용

주된 목적은 손실을 최소화하면서 저차원으로 변환하는 것. 훈련집합의 분산이 클수록 정보 손실이 적다고 판단됨.

최적화 문제 = 훈련집합 x 를 z 으로 변환하면서 분산을 최대화하는 q 개의 축을 찾는 것.

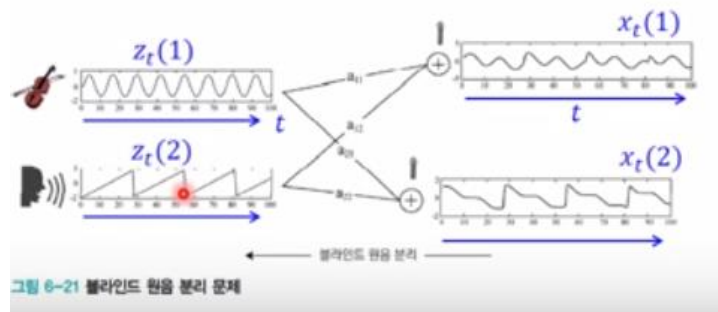
디코딩 과정 = w 는 정규직교 행렬.

$$\tilde{\mathbf{x}} = \mathbf{W}\mathbf{z}$$

독립 성분 분석(ICA) = z 가 비가우시안 분포를 따른다고 가정할 때 사용.

실제 세계에서는 여러 신호가 섞여 나타남. Ex) 뇌파와 장기 신호가 섞인 EEG, 잡음이 섞인 영상

문제 정의 = x 로부터 z 를 찾는 문제



과소 조건 문제 = 조건 제약이 적어서 많은 답이 나올 수 있는 상황(추가 조건 제시). 과소 적합이 문제  
따라서 독립성 가정과 비가우시안 가정을 세운다.

독립성 가정 = 원래 신호가 서로 독립이라는 가정

비가우시안 가정 = 원래 신호가 가우시안이라면 분리 불가능. 그러나 가우시안 이어도 성분이 다르거나 mean 이 다르면 분리가능.

문제 풀이 = 비가우시안 정도를 최대화하는 가중치를 구하는 전략.

$$\hat{\mathbf{w}}_j = \operatorname{argmax}_{\mathbf{w}_j} \check{G}(z_j) \quad (6.29)$$

- $\check{G}$ 는 비가우시안 정도를 측정하는 함수
- 주로 식 (6.31)의 점도를 사용

$$\text{kurtosis}(z_j) = \frac{1}{n} \sum_{i=1}^n z_{ji}^4 - 3 \left( \frac{1}{n} \sum_{i=1}^n z_{ji}^2 \right)^2 \quad (6.31)$$

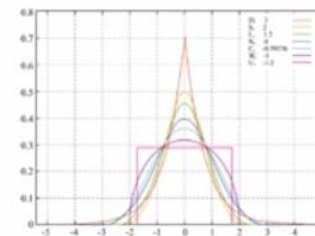


그림 6-23 여러 가지 분포의 점도 측정

학습 = 1)전처리 수행, 2)최적 가중치 구함



### ■ PCA와 ICA 비교

- ICA는 비가우시안과 독립성 가정, PCA는 가우시안과 비상관을 가정
- ICA는 4차 모멘트까지 사용, PCA는 2차 모멘트까지 사용
- ICA로 찾은 축은 수직 아님, PCA로 찾은 축은 서로 수직
- ICA는 주로 블라인드 원음 분리 문제를 푸는데, PCA는 차원 축소 문제를 풀

희소 코딩 = 기저함수 또는 벡터의 선형 결합으로 신호를 표현. 사전  $D$  를 구성하는 기저 벡터  $d_1, d_2, \dots$ 의 선형 결합으로 신호  $x$  를 표현한다. 단 개수를 최대한 적게 한다(sparse).

$$\left. \begin{array}{l} \mathbf{x} = \mathbf{D}\mathbf{a} \\ \text{이때 } \mathbf{D} = (\mathbf{d}_1 \ \mathbf{d}_2 \ \cdots \ \mathbf{d}_m) \end{array} \right\} \quad (6.32)$$

비지도 학습이 사전(기저 벡터)을 자동으로 알아냄 => 희소 코딩은 데이터에 맞는 기저 벡터를 사용하는 셈이다.

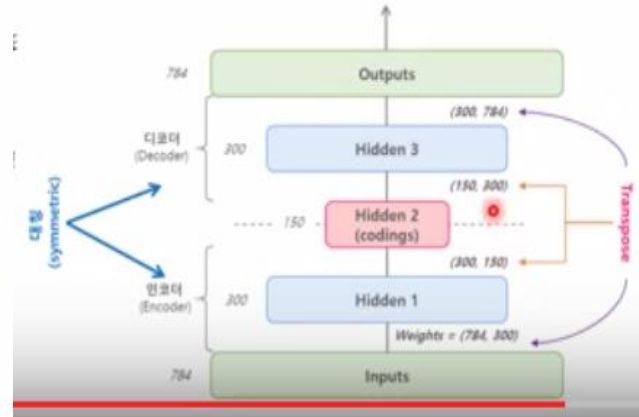
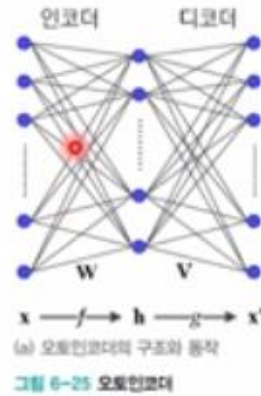
희소 코딩 구현 = 최적의 사전과 최적의 희소 코드를 알아야 함.

- $\phi$ 는 희소 코드의 희소성을 강제하는 규제항

$$\hat{\mathbf{D}}, \hat{\mathbf{A}} = \underset{\mathbf{D}, \mathbf{A}}{\operatorname{argmin}} \sum_{i=1}^n \|\mathbf{x}_i - \mathbf{D}\mathbf{a}_i\|_2^2 + \lambda \phi(\mathbf{a}_i)$$

### <6.7 절 오토 인코더>

오토 인코더 = 특징 벡터  $x$  를 입력 받아 동일한 또는 유사한 벡터  $x'$  를 출력하는 신경망

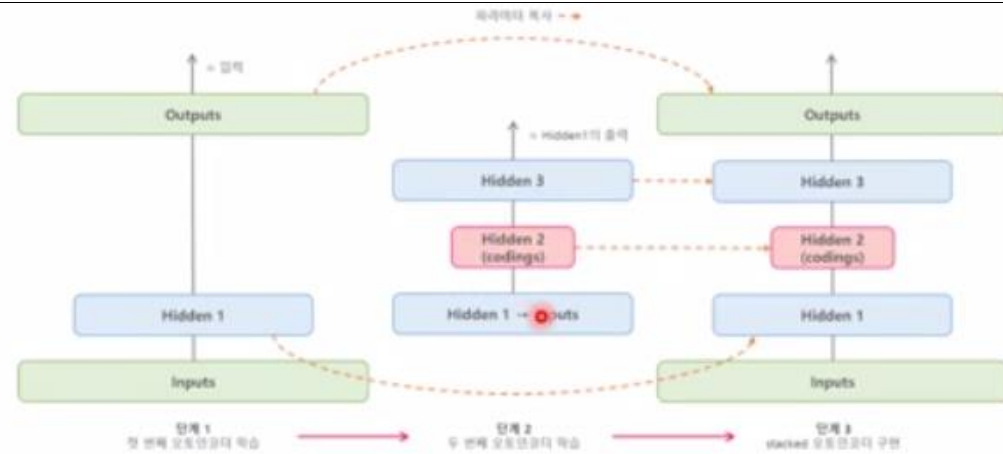


규제 오토인코더 =  $m > d$  인 상황에서도 단순 복사를 피할 수 있다. 충분히 큰 모델을 사용하되 적절한 규제 기법을 적용하는 현대 기계 학습 추세를 오토인코더로 따르는 셈이다.

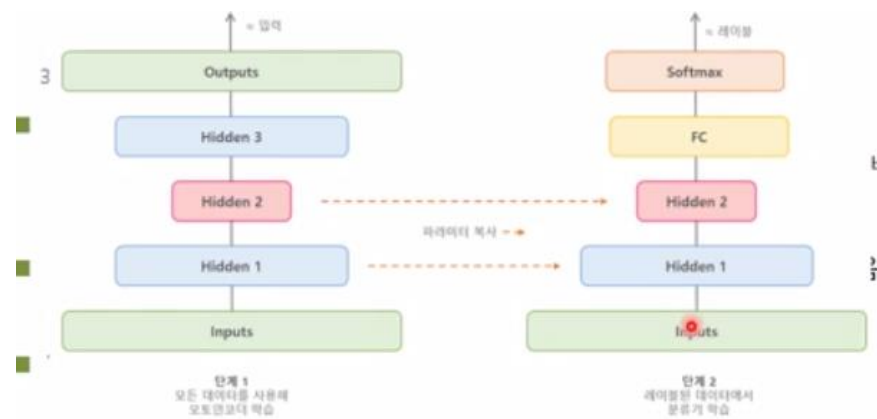
SAE (sparse autoencoder) = 은닉 벡터가 희소하도록 강제화

DAE (denoising autoencoder) = 잡음을 추가한 다음 원본을 복원하도록 학습하는 원리

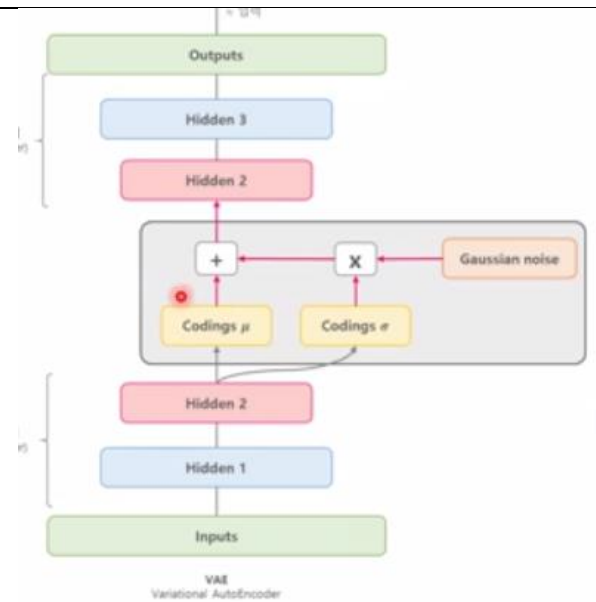
적층(stack)오토인코더 = 오토인코더는 알고 은닉층이 하나뿐인 신경망이라서 여러 층으로 쌓으면 용량이 커진다. 적층 오토인코더는 층별 예비학습을 이용하여 깊은 신경망을 만든다.



적층 오토인코더를 지도 학습에 활용하는 경우의 학습과정 = 층별 예비학습을 필요한 만큼 수행 -> 마지막 층의 출력을 입력으로 하여 MLP 를 학습한다. ->신경망 전체를 한꺼번에 추가로 학습한다.



Variational autoencoders (VAEs) -> GAN = 가우시안 필터 사용



### <6.8 절 매니폴드 학습>

매니폴드란 고차원 공간에 내재한 저차원 공간이다. 보통 매니폴드는 비선형 공간이지만 지역적으로 살펴보면 선형 구조이다.

Isomap = k-최근접 이웃을 구하여 그 거리를  $n \times n$  행렬  $M$  에 채운다.

LLE = 거리 행렬  $M$  대신에 가중치의 합(비율)  $W$  으로 표현한다.

t-SNE(stochastic neighbor embedding) = 매니폴드 공간 변환 기법 중에서 가장 뛰어나다. 원래 공간에서 유사도 측정한다.

t-distribution

### ■ Pdf of Student's *t*-distribution

$$f(t) = \frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi} \Gamma(\frac{\nu}{2})} \left(1 + \frac{t^2}{\nu}\right)^{-\frac{\nu+1}{2}}, \quad Z \text{에서 } \sigma^2 \text{ 대신 } S^2 \text{로 대체 } T = \frac{Z}{\sqrt{V/\nu}} = \frac{(\bar{X} - \mu)}{S} \frac{\sqrt{n}}$$

• where  $\nu$  is the number of *degrees of freedom* and  $\Gamma$  is the *gamma function*

•  $\nu = 1$

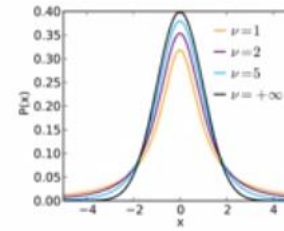
Distribution function:

$$F(t) = \frac{1}{2} + \frac{1}{\pi} \arctan(t).$$

Density function:

$$f(t) = \frac{1}{\pi(1+t^2)}.$$

See Cauchy distribution



$$\Gamma(z) = \int_0^{\infty} x^{z-1} e^{-x} dx$$

$$\Gamma(z+1) = z \int_0^{\infty} x^{z-1} e^{-x} dx = z\Gamma(z).$$

변환된 공간에서의 유사도는 스튜던트 t 분포로 측정한다. 원래 데이터와 변환된 데이터의 구조가 비슷해야 하므로 KL 다이버전스 사용. q 분포를 변경

학습 알고리즘 = p,q 의 KL 다이버전스를 최소화 하는 x'를 찾는 문제.

• 확률 분포  $P$ 와  $Q$ 가 비슷할수록 좋음

• 비슷한 정도를 측정하기 위해 식 (6.47)의 KL 다이버전스를 사용

$$J(\mathbb{X}') = KL(P \parallel Q) = \sum_{i=1}^n \sum_{j=1}^n p_{ij} \log \left( \frac{p_{ij}}{q_{ij}} \right)$$

•  $y_i$  update of symmetric t-SNE

### 질문 내용

1. 데이터셋의 차원을 축소시키고 나서 이 작업을 되돌리는 것이 가능한가?
2. 어떤 데이터셋에 적용한 차원 축소 알고리즘의 성능은 어떻게 평가할 수 있나?

## <기계학습 14주차 레포트>

컴퓨터/전자시스템 공학부

201500629 김영연

작성일: 2019.12.04

구분	내용
학습 범위	6.5절 공간 변환의 중요성 6.6절 PCA, ICA, sparse 코딩 6.7절 AUTO ENCODER 6.8절 매니폴드 개념과 isomap, LLE, T-SNE 매니폴드 학습기법
학습 내용	<ul style="list-style-type: none"><li>- 인코딩: 원래 공간을 다른 공간으로 변환하는 과정 (f)</li><li>- 디코딩: 변환 공간을 원래 공간으로 역변환 하는 과정 (g) <math display="block">X = g(f(x))</math></li><li>- 선형 인자 모델: 선형 연산을 이용한 공간 변환 기법 이를 인코딩 식과 디코딩 식으로 나타내면 다음과 같다. <math display="block">f: \mathbf{z} = \mathbf{W}_{enc}\mathbf{x} + \boldsymbol{\alpha}_{enc}</math><math display="block">g: \mathbf{x} = \mathbf{W}_{dec}\mathbf{z} + \boldsymbol{\alpha}_{dec}</math>알파는 데이터를 원점으로 이동하거나 잡음을 추가하는 등의 역할</li></ul> <p>-주 성분 분석이 필요한 이유: 데이터의 차원이 커지면 차원 내의 부피가 기하급수적으로 커짐 -&gt; 차원이 커지</p>

고 차원의 부피도 커지면 그만큼 데이터가 없는 빈공간이 많아짐(차원의 저주)-> 이러한 현상을 개선하기 위해 차원 축소가 필요하다.

주성분을 분석하기 위해서는 데이터를 원점 중심으로 옮기는 전처리를 먼저 수행해야 한다.

$$\left. \begin{array}{l} \mathbf{x}_i = \mathbf{x}_i - \boldsymbol{\mu}, \quad i = 1, 2, \dots, n \\ \text{이때 } \boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \end{array} \right\}$$

주성분 분석이 사용하는 변환식: 변환 행렬  $\mathbf{W}$ 는  $d \times q$ 로서 주성분 분석은  $d$ 차원의  $\mathbf{x}$ 를  $q$  차원의  $\mathbf{z}$ 로 변환( $q < d$ )

$$\mathbf{z} = \mathbf{W}^T \mathbf{x}$$

$$\text{이때 } \mathbf{W} = (\mathbf{u}_1 \ \mathbf{u}_2 \ \dots \ \mathbf{u}_q) \text{이고, } \mathbf{u}_j = (u_{1j}, u_{2j}, \dots, u_{dj})^T$$

주성분 분석은 변환된 훈련집합의 분산이 클수록 정보 손실이 적다고 판단

- 차원 축소에는 대표적인 두 가지 방법이 있다.

하나의 변수선택(총  $M$ 개의 설명변수 중에서  $N$ 개의 변수만을 뽑아 사용)이고 나머지 하나는 변수를 추출( $M$ 차원의 벡터를 넣어  $N$ 차원의 벡터를 출력하는 것  $\rightarrow$  PCA)하는 것이다.

- PCA: 데이터가 멀리 퍼진 방향, 즉 분산이 큰 방향으로 하나의 축을 잡고 그 축에 모든 데이터를 사영시키는 것

- 디코딩 과정:  $\tilde{\mathbf{x}} = \mathbf{W}\mathbf{z}$

- 독립성분 분석의 문제 공식화: 혼합신호  $\mathbf{x}$ 를 원래 신호  $\mathbf{z}$ 의 선형 결합으로 표현 가능

$$x_1 = a_{11}z_1 + a_{12}z_2$$

$$x_2 = a_{21}z_1 + a_{22}z_2 \quad z_1 \text{과 } z_2 \text{는 독립이라 가정}$$

$$\text{이를 행렬로 표현하면 } \mathbf{x} = \mathbf{A}\mathbf{z}$$

- ICA: 독립성 분해 모든 방향이 동일하게 분산

	<p>원래 신호의 비가우시안인 정도를 최대화하는 가중치를 구하는 전략을 사용</p> <p>ICA학습을 하기 위해서는 먼저 전처리를 수행해야 하고 <math>\hat{\mathbf{w}}_j = \underset{\mathbf{w}_j}{\operatorname{argmax}} \check{G}(\mathbf{z}_j)</math> 를 풀어 가중치를 구한다.</p> <p>- 오토인코더: 특징 벡터 <math>\mathbf{x}</math>를 입력 받아 동일한 또는 유사한 벡터 <math>\mathbf{x}'</math>를 출력하는 신경망</p>
질문 내용	주성분 분석 중 데이터내에 비선형적 관계가 있으면 잘 작동하지 않는다고 한다. 이때의 해결방법에 대해 알고 싶다.

## <기계학습 14주차 레포트>

컴퓨터공학과

201402033 위성조

작성일: 2019.12.04

구분	내용
학습 범위	<p>6.5 공간 변환의 이해</p> <p>6.6 선형 인자 모델</p> <p>6.7 오토인코더</p> <p>6.8 매니폴드 학습</p>



## 학습 내용

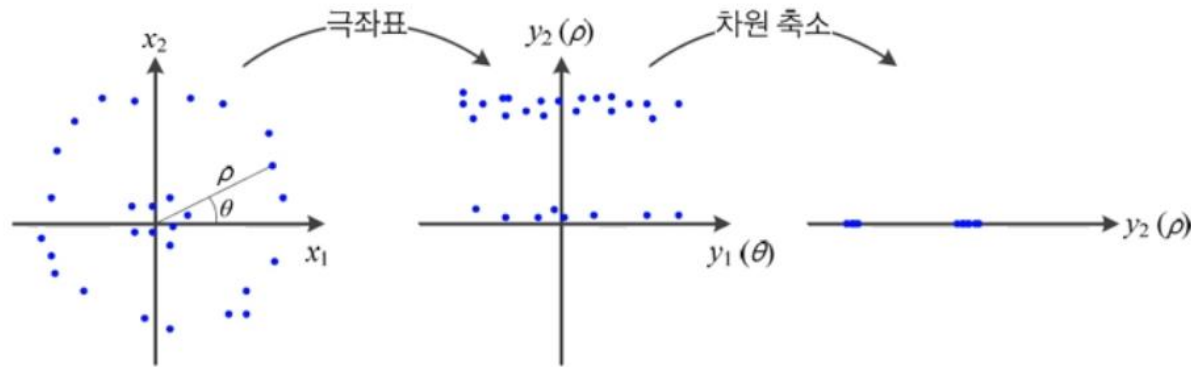


그림 6-16 공간 변환의 예

실제 문제에서는 비지도 학습을 통해 최적의 공간 변환을 자동으로 알아내야 함

원래 공간을 다른 공간으로 변환하는 인코딩, 변환 공간을 원래 공간으로 역변환 하는 디코딩.

데이터 압축의 경우, 역변환으로 얻은  $x'$ 는 원래 신호  $x$ 와 가급적 같아야 함

데이터 가시화에서는 2차원 또는 3차원의 공간으로 변환. 디코딩 불필요

선형 인자 모델 - 선형 연산을 이용한 공간 변환 기법

선형 연산을 사용하므로 행렬 곱으로 인코딩( $f$ ), 디코딩( $g$ ) 과정을 표현

$$f: \mathbf{z} = \mathbf{W}_{enc} \mathbf{x} + \boldsymbol{\alpha}_{enc} \quad g: \mathbf{x} = \mathbf{W}_{dec} \mathbf{z} + \boldsymbol{\alpha}_{dec}$$

$\mathbf{A}$ 는 데이터를 원점으로 이동하거나 잡음을 추가하는 등의 역할

$\mathbf{Z}$ 에 확률 개념이 없고,  $\boldsymbol{\alpha}$ 를 생략하면 주성분 분석

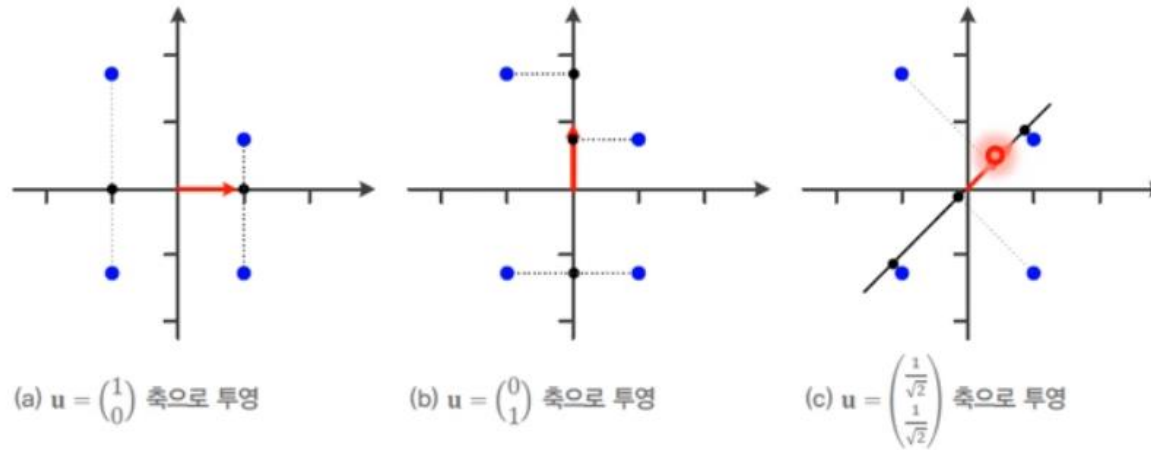
주성분 분석 (PCA : Principal Component Analysis)

데이터를 원점 중심으로 옮기는 전처리를 먼저 수행 :  $X_i = X_i - \mu$  ( $\mu$ : 평균)

변환 행렬  $\mathbf{W}$ 는  $d \times q$ 로서 주성분 분석은  $d$ 차원의  $\mathbf{x}$ 를  $q$ 차원의  $\mathbf{z}$ 로 변환( $q < d$ )

$\mathbf{W}$ 의  $j$ 번째 열 벡터와의 내적  $u_j^T \mathbf{x}$ 는  $\mathbf{x}$ 를  $u_j$ 가 가리키는 축으로 투영

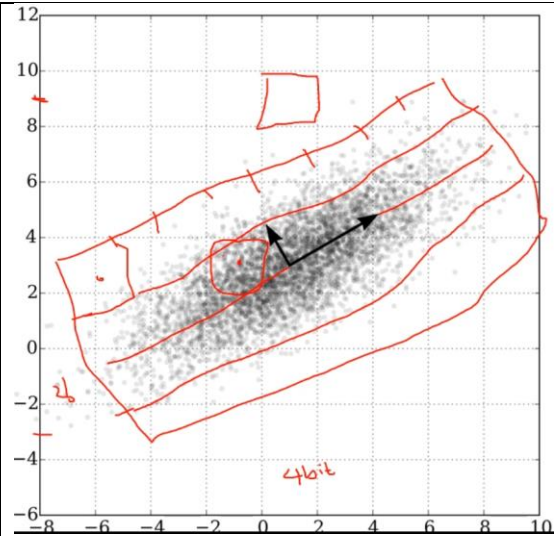
- 예, 2차원을 1차원으로 변환하는 상황( $d = 2, q = 1$ )



주성분 분석의 목적

손실을 최소화하면서 저차원으로 변환하는 것 - **변환된 훈련집합의 분산이 클수록** 정보 손실이 적다고 판단.

PCA 기반 데이터 압축



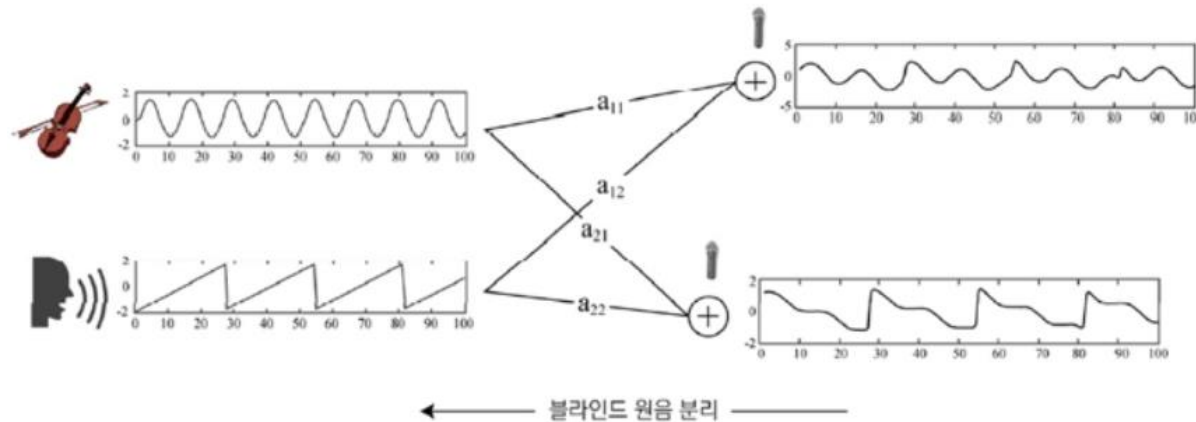
디코딩 과정

역변환은  $x = (W^T)^{-1}z$ 인데,  $W$ 가 정규직교 행렬이므로  $X' = Wz$  가 됨.

$q = d$ 로 설정하면  $W$ 가  $d \times d$ 이고  $X'$ 는 원래 샘플  $X$ 와 같게 됨 - 원래 공간을 단지 일정한 양만큼 회전하는 것에 불과  
실제로는  $q < d$ 로 설정하여 차원 축소를 피함

- 데이터 압축
- $q = 2$  또는  $q = 3$ 으로 설정하여 2차원 또는 3차원으로 축소하여 데이터 가시화
- 고유얼굴 기법:  $256 \times 256$  얼굴 영상( $d = 65536$ )을 7차원( $q = 7$ )으로 변환하여 얼굴 인식(정면 얼굴에 대해 96% 정확률) -  
> 상위 몇 개의 고유벡터가 대부분의 정보를 가짐

독립 성분 분석 (ICA : Independent Component Analysis)



마이크로 측정한 혼합 신호로부터 원음(음악과 목소리)를 복원할 수 있나? -> 블라인드 원음 분리 문제라 부르며, 독립 성분 분석 기법으로 해결 가능

혼합 신호  $x$ 를 원래 신호  $z$ 의 선형 결합으로 표현 가능( $z_1(t)$ 와  $z_2(t)$ 가 독립이라는 가정)

$$x_1 = a_{11}z_1 + a_{12}z_2, \quad x_2 = a_{21}z_1 + a_{22}z_2$$

행렬 표기로 쓰면  $x = AZ$

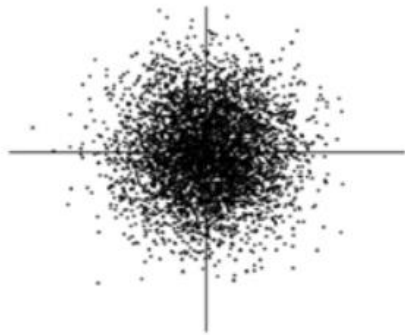
블라인드 원음 분리 문제란  $A$ 를 구하는 것.  $A$ 를 알면,  $z = Wx$ , 이때  $W = A^{-1}$ 를 이용해 원음 복원

정수 하나를 주고 어떤 두 수의 곱인지 알아내라는 문제와 비슷함 -> 추가 조건을 주면 유일해가 가능.

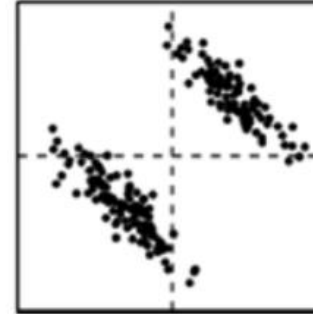
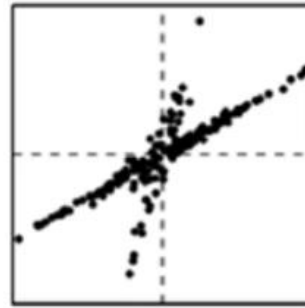
독립성 가정과 비가우시안 가정을 이용하여  $x = Az$ 의 해를 찾음.

독립성 가정 - 원래 신호가 서로 독립이라는 가정 (음악과 대화는 서로 무관하게 발생함)

비가우시안 가정 - 원래 신호가 가우시안이라면 혼합 신호도 가우시안이 되므로 분리할 실마리 없음. 비가우시안이면 실마리가 있음.



(a) 확률변수가 가우시안일 때



(b) 확률변수가 비가우시안일 때

ICA의 문제 풀이 - 원래 신호의 비가우시안인 정도를 최대화하는 가중치를 구하는 전략 사용

PCA와 ICA의 비교

PCA	ICA
가우시안과 비상관 가정	비가우시안과 독립성 가정
2차 모멘트까지 사용	4차 모멘트까지 사용
PCA로 찾은 축은 서로 수직	ICA로 찾은 축은 수직 아님
주로 차원 축소 문제 해결	주로 블라인드 원음 분리 문제 해결

희소 코딩 - 기저함수 또는 기저벡터의 선형 결합으로 신호를 표현

푸리에 변환 또는 웨이블릿 변환 등

희소 코딩이 다른 변환 기법과 다른 점

비지도 학습이 사전(기저벡터)를 자동으로 알아냄 (푸리에 변환은 삼각함수를 사용함)

→ 희소 코딩은 데이터에 맞는 기저 벡터를 사용하는 셈

사전의 크기를 과잉 완벽하게 책정 ( $m > d$ )

희소 코드  $\mathbf{a}$ 를 구성하는 요소 대부분이 0값을 가짐

희소 코딩 구현 - 최적의 사전과 최적의 희소 코드를 알아내야 함

$$\hat{\mathbf{D}}, \hat{\mathbf{A}} = \underset{\mathbf{D}, \mathbf{A}}{\operatorname{argmin}} \sum_{i=1}^n \|\mathbf{x}_i - \mathbf{D}\mathbf{a}_i\|_2^2 + \lambda \phi(\mathbf{a}_i)$$

,  $\phi$ 는 희소 코드의 희소성을 강제하는 규제항

오토인코더 - 특징 벡터  $\mathbf{x}$ 를 입력받아 동일한 또는 유사한 벡터  $\mathbf{x}'$ 를 출력하는 신경망  
단순 복사하는 단위 행렬은 무의미

병목 구조 오토인코더의 동작 원리

$m < d$  인 구조 (ex, 256\*256 영상을 입력 받아 256\*256 영상을 출력하는 경우  $d=65536$ 인데  $m=1024$ 로 설정)

은닉층의  $h$ 는 훨씬 적은 메모리로 데이터 표현. 필요한 경우, 디코더로 원래 데이터 복원

$h$ 는  $\mathbf{x}$ 의 핵심 정보를 표현 -> 특징 추출, 영상 압축 등의 응용

여러 형태의 오토인코더

은닉 노드 개수에 따라  $m < d$ ,  $m = d$ ,  $m > d$  구조 / 활성화함수에 따라 선형과 비선형 구조

오토인코더의 학습

주어진 데이터는 훈련집합  $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  알아내야 하는 매개변수는  $f$ 와  $g$ 라는 매핑 함수 즉 가중치집합  $\theta = \{W, V\}$

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^n L(\mathbf{x}_i, g(f(\mathbf{x}_i)))$$

$$L(\mathbf{x}_i, g(f(\mathbf{x}_i))) = \|\mathbf{x}_i - g(f(\mathbf{x}_i))\|_2^2$$

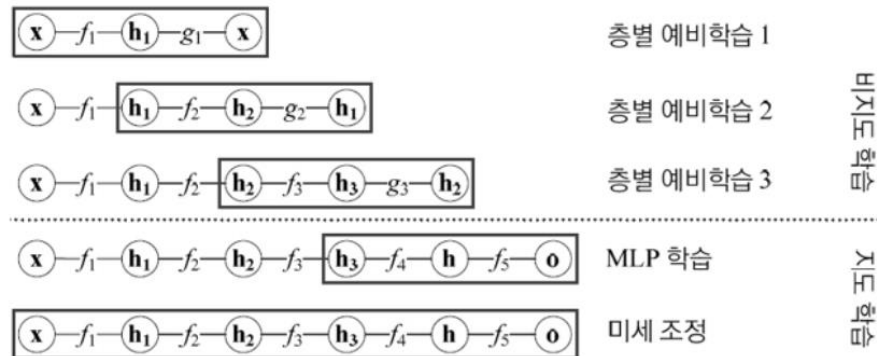
규제 오토인코더

여러 규제 기법을 적용 -  $m > d$ 인 상황에서도 단순 복사를 피할 수 있음  
 SAE (sparse autoencoder) - 은닉 벡터  $h_i$ 가 희소하도록 강제화(0이 아닌 요소의 개수를 적게 유지)  
 DAE (denoising autoencoder) - 잡음을 추가한 다음 원본을 복원하도록 학습하는 원리  
 CAE (contractive autoencoder) - 인코더함수  $f$ 의 야코비안 행렬의 프로베니우스 놈을 작게 유지  
 CAE는 공간을 축소하는 효과

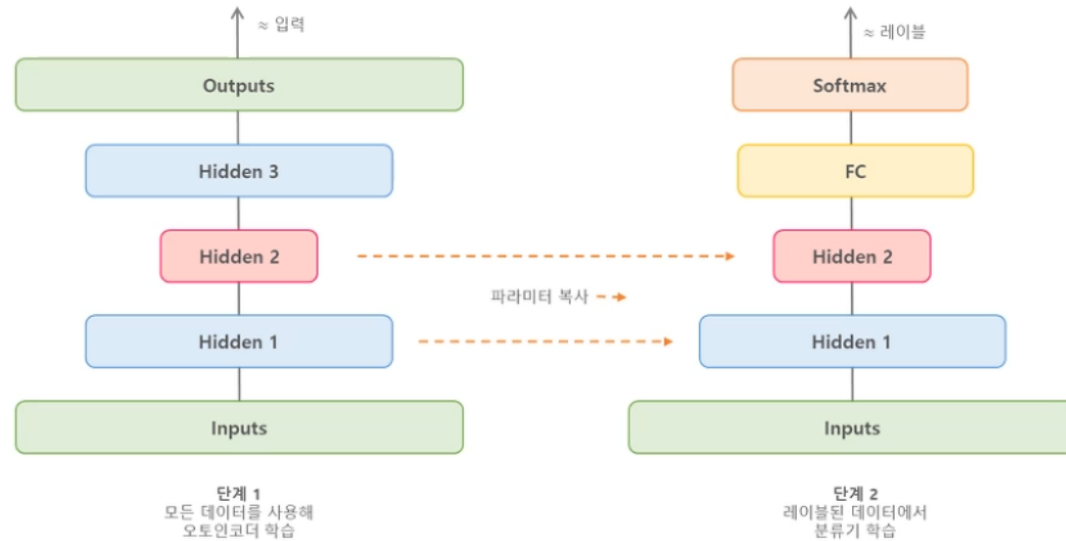
적층 오토인코더

은닉층이 하나인 경우 표현력에 한계가 있다. -> 여러 층으로 쌓으면 용량이 커짐

층별 예비학습을 이용하여 깊은 신경망을 만듦



적층 오토인코더를 지도학습(분류)에 활용하는 경우의 학습 과정



매니폴드 - 고차원 공간에 내재한 저차원 공간

도로가 매니폴드에 해당

자동차 위치를 3차원 데이터로 나타낼 수 있으나, 기준점에서의 거리 라는 1차원(저차원) 공간, 즉 매니폴드로 표현할 수 있음.

보통 매니폴드는 비선형 공간이지만 지역적으로 살펴보면 선형 구조

매니폴드 가정 - 고차원 공간에 주어진 실제 세계의 데이터는 고차원 입력 공간  $R^d$ 에 내재한 훨씬 저차원인  $d_M$ 차원 매니폴드의 인근에 집중되어 있다.

매니폴드를 어떻게 구할까

IsoMap = 최근접 이웃 그래프 구축

1. 각 점은 k-최근접 이웃을 구하여 거리를  $n \times n$ 행렬 M에 채움
2. 빈 곳은 최단 경로의 shortest path 길이로 채움

M의 고유 벡터를 계산하고, 큰 순서대로  $d_{low}$ 개의 고유 벡터를 선택



- 이들 고유 벡터가 새로운 저차원 공간 형성
- i번째 샘플의 k번째 좌표는  $\sqrt{\lambda_k} v_k^i$ 로 변환

M이 너무 크다는 문제점

LLE (locally linear embedding)

거리 행렬 M대신에 함수  $\epsilon$ 를 최소로 하는 가중치 행렬 W를 사용함.

$$\epsilon(\mathbf{W}) = \sum_{i=1}^n \left\| \mathbf{x}_i - \sum_{\mathbf{x}_j \in \{\mathbf{x}_i \text{의 이웃}\}} w_{ij} \mathbf{x}_j \right\|_2^2$$

t-SNE (stochastic neighbor embedding)

현재 t-SNE는 매니폴드 공간 변환 기법 중에서 가장 뛰어난

원래 공간에서 유사도 측정

변환된 공간에서의 유사도는 스튜던트 t 분포로 측정

$\mathbf{y}_i$ 와  $\mathbf{y}_j$ 는 변환된 공간에서의 점

$$q_{ij} = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|_2^2)^{-1}}{\sum_{k \neq i} (1 + \|\mathbf{y}_i - \mathbf{y}_k\|_2^2)^{-1}}$$

원래 데이터와 변환된 데이터의 구조가 비슷해야 하므로, 확률 분포 P와 Q가 비슷할수록 좋음

비슷한 정도를 측정하기 위해 아래의 KL 다이버전스를 사용

$$J(\mathbb{X}') = KL(P \parallel Q) = \sum_{i=1}^n \sum_{j=1}^n p_{ij} \log \left( \frac{p_{ij}}{q_{ij}} \right)$$

	<p>Transductive 학습 모델</p> <p>훈련집합 이외의 샘플을 처리할 능력이 없는 모델</p> <p>IsoMap, LLE, t-SNE 모두 트랜스덕티브 모델</p> <p>데이터 가시화라는 목적에 관한 한 PCA나 오토인코더와 같은 귀납적 모델보다 성능이 뛰어남</p> <p>귀납적 모델 (inductive model, bottom-up)</p> <p>훈련집합 이외의 새로운 샘플을 처리할 능력이 있는 모델</p> <p>IsoMap, LLe, t-SNE를 제외한 지금까지 공부한 모든 모델</p>
<b>질문 내용</b>	<ol style="list-style-type: none"> <li>1. 오토인코더를 분류기로 활용하는 경우, CNN 과 비교하여 어떤 차이가 있는지 알고 싶습니다.</li> <li>2. 오토인코더를 이용하면 특징 추출과 분류를 따로 학습시키게 되므로 특징 추출과 분류를 한번에 학습시키는 CNN 에 비해 성능이 더 좋을 것 같은데, 이 이론이 맞는지, 맞다면 대부분의 이미지 인식 네트워크에서 CNN 을 사용하는 특별한 이유가 있는지 알고 싶습니다.</li> </ol>

## <기계학습 14주차 레포트>

컴퓨터/전자시스템 공학부

201502469 이재은

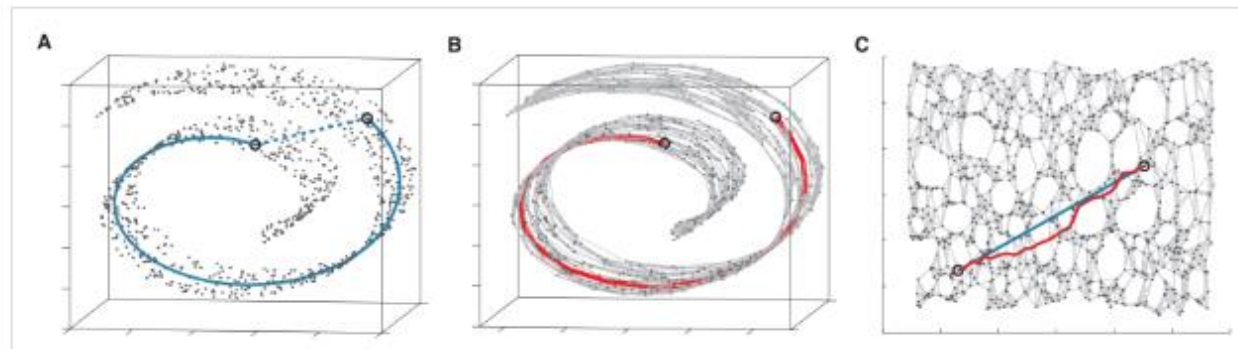
작성일: 2019.12.04

구분	내용
<b>학습 범위</b>	<p>6.5절: 기계 학습에서 공간 변환의 중요성을 강조한다.</p> <p>6.6절: 선형 인자 모델로서 PCA, ICA, 희소 코딩을 소개한다.</p> <p>6.7절: 오토인코더를 소개하고 규제 오토인코더로서 SAE, DAE, CAE를 설명한다.</p> <p>6.8절: 매니폴드 개념을 소개하고 IsoMap, LLE, t-SNE라는 매니폴드 학습 기법을 설명한다.</p>

<p><b>학습 내용</b></p>	<ul style="list-style-type: none"> <li>- 입력의 합은 출력의 합과 같다: Super position</li> <li>- PCA(Principal Component Analysis): 데이터의 분산(variance)을 최대한 보존하면서 서로 직교하는 새 축을 찾아, 고차원 공간의 표본들을 선형 연관성이 없는 저차원 공간으로 변환하는 기법이다.</li> <li>- PCA 단계: 1) 데이터를 원점 중심으로 옮기는 전처리를 먼저 수행 2) 각각의 축으로 투영</li> <li>- PCA 목적: 손실을 최소화하면서 저차원으로 변환하는 것 //분산이 클수록 정보 손실이 적다.</li> <li>- PCA의 학습 알고리즘: 1) 훈련집합으로 공분산 행렬을 계산한다. 2) 식 (6.22)를 풀어 d개의 고유값과 고유 벡터를 구한다. 3) 고유값이 큰 순서대로 주성분들을 나열한다. 4) q개의 주성분들을 선택하여 식(6.20)에 있는 행렬 <b>W</b>에 채운다.</li> <li>- PCA를 이용해서 데이터 압축이 어떻게 되는가?</li> <li>- ICA(Independent Component Analysis): 다변량의 신호를 통계적으로 독립적인 하부 성분으로 분리하는 계산 방법이다.</li> <li>- ICA는 실제로 그렇게 많이 사용되지 않는다.</li> <li>- PCA와 ICA 비교 <table border="1"> <thead> <tr> <th>PCA</th><th>ICA</th></tr> </thead> <tbody> <tr> <td>가우시안과 비상관</td><td>비가우시안과 독립성 가정</td></tr> <tr> <td>2차 모멘트까지 사용</td><td>4차 모멘트까지 사용</td></tr> <tr> <td>찾은 축은 서로 수직이다</td><td>찾은 축은 수직이 아니다</td></tr> <tr> <td>차원 축소 문제</td><td>블라인드 원음 분리 문제</td></tr> </tbody> </table> </li> <li>- Non-negative matrix factorization: 행렬이 있으면 이 행렬을 두 행렬의 곱으로 표현하는데, 행렬의 element 들이 모두 양의 값으로 decomposition 하는 것 //음원 분리에 사용</li> </ul>	PCA	ICA	가우시안과 비상관	비가우시안과 독립성 가정	2차 모멘트까지 사용	4차 모멘트까지 사용	찾은 축은 서로 수직이다	찾은 축은 수직이 아니다	차원 축소 문제	블라인드 원음 분리 문제
PCA	ICA										
가우시안과 비상관	비가우시안과 독립성 가정										
2차 모멘트까지 사용	4차 모멘트까지 사용										
찾은 축은 서로 수직이다	찾은 축은 수직이 아니다										
차원 축소 문제	블라인드 원음 분리 문제										

$$\begin{bmatrix} & W \\ & \\ & \\ & \\ & \end{bmatrix} \times \begin{bmatrix} & H \\ & \\ & \\ & \\ & \\ & \\ & \\ & \end{bmatrix} \approx \begin{bmatrix} & V \\ & \\ & \\ & \\ & \\ & \\ & \\ & \end{bmatrix}$$

- 희소코딩(Sparse Coding): 기저함수 또는 기저 벡터의 선형 결합으로 신호를 표현하는 것이다.
- 오토인코더: 자기 스스로 코딩한다. 대표적으로 규제 오토인코더, 적층 오토인코더가 있다.
- SAE(Sparse Auto-Encoder): 말 그대로 듽성듬성한 오토인코더를 말하는 것이다. 가장 큰 특징은, 히든 유닛의 수가 Input 데이터의 차원의 수(Input unit의 수)에 비해 상당히 많은 경우에 Sparsity Parameter을 제어하여, Hidden Unit의 Activation을 제어할 수 있는 오토인코더이다.
- 오토인코더는 얇은 신경망, 적층인코더는 깊은 신경망
- 매니폴드: 고차원 공간에 내재한 저차원 공간
- IsoMap: manifold 에서의 점들 간의 거리를 nearest neighbor graph 에서의 점들 간의 최단 경로로 정의한다.



위와 같은 스위스롤 데이터를 isomap은 manifold 에서의 이웃 간의 정보를 보존하는 2차원 평면으로 학습할 수 있다.

	<div data-bbox="589 209 1805 683" data-label="Figure"> </div> <p>- t-SNE: 현재 매니폴드 공간 변환 기법 중에서 가장 뛰어나다.</p>
질문 내용	

## <기계학습 14주차 레포트>

컴퓨터공학과

201403474 최진성

작성일: 2019.12.04

구분	내용
학습 범위	기계학습 6장 비지도 학습 6.5 공간 변환의 이해 6.6 선형 인자 모델 6.7 오토인코더 6.8 매니폴드 학습
학습 내용	기계학습 6장 비지도 학습  6.5 공간 변환의 이해 차원 $n$ 에 해당하는 특징 공간을 $m$ 차원으로 옮겨서 최적의 공간 변환을 통해 데이터 군집 분류에 도움이 되게 하는 것.  Encoding – 원래 공간을 다른 공간으로 변환 Decoding – 다른 공간을 원래 공간으로 변환 <ul style="list-style-type: none"><li>● 데이터 압축의 경우 역 변환으로 얻은 <math>x'</math>는 <math>x</math>와 가급적 같아야 함</li><li>● 데이터 가시화에서는 2-3차원의 <math>z</math>공간으로 변환하나 디코딩은 불필요하다.</li></ul> 6.6 선형 인자 모델 -> 선형 연산을 이용한 공간 변환 기법(행렬 곱으로 Encoding, Decoding 과정 표현) $f: z = W_{enc}x + \alpha_{dec}$

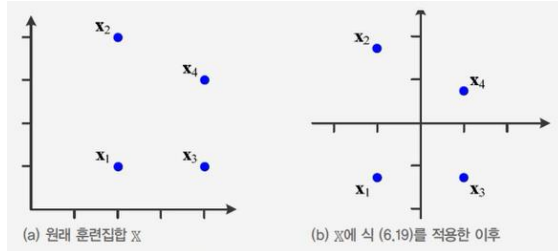
$$g : x = W_{dec}z + \alpha_{enc}$$

- $\alpha$ 는 데이터를 이동시키거나 잡음을 추가하는 역할
- 인자  $z$ 와  $\alpha$ 에 따라 여러가지 모델이 존재함.
  - ➔  $z$ 에 확률 개념이 없고  $\alpha$  생략 -> PCA(관찰 벡터  $x$ 와 인자  $z$ 는 결정론적인 1:1 매핑 관계)
  - ➔  $z$ 와  $\alpha$ 가 가우시안 분포를 따른다고 가정할 경우 PCA(Probabilistic PCA)
  - ➔  $z$ 가 비가우시안 분포를 따른다고 가정할 경우 ICA

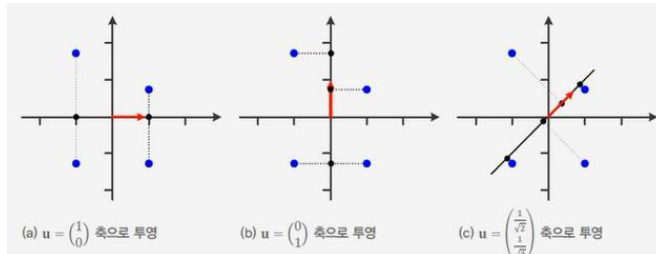
### 주성분 분석

- 손실을 최소화 하면서 저차원으로 변환하는 것이 목적
  - ➔ 데이터가 저차원으로 이동하면서 같은 점으로 수렴하는 것을 방지함 -> 데이터의 분산이 클수록 정보 손실이 적음.

1. 데이터를 원점 중심으로 옮기는 전처리를 먼저 수행한다.



2. 데이터를 다른 차원으로 변환한다. (데이터를 잘 표현할 수 있는 축 찾기)



\* 주성분 분석의 학습 알고리즘

1. 훈련집합으로 공분산 행렬  $\Sigma$ 를 계산한다.
2. 식을 풀어 d개의 고윳값과 고유 벡터를 구한다
3. 고윳값이 큰 순서대로  $u_1, u_2, \dots, u_d$ 를 나열한다( = 주성분 나열)
4. q개의 주성분  $u_1, u_2, \dots, u_q$ 를 선택하여 행렬 W에 채운다.

디코딩 과정

역변환은  $x = (W^T)^{-1}z$ 인데 W가 정규직교 행렬이므로 식  $\tilde{x} = Wz$ 가 됨.

q = d로 설정하면 W가 d \* d이므로  $\tilde{x} = x$ (원래 공간을 일정한 양만큼 회전)

실제로는  $q < d$ 로 설정하여 차원 축소를 꾀함.

Ex : 데이터 압축, 얼굴 인식, 데이터 가시화 등.

독립 성분 분석

실제 세계에서는 많은 신호가 섞여 나타남. (ex : 음악과 대화 파장이 섞이는 등)

→ 블라인드 원음 분리 문제.

$z_1(t)$ 와  $z_2(t)$ 를 원래 신호로, 혼합 신호를  $x_1(t), x_2(t)$ 로 표기한다.

t 시간에 획득한  $x_t = (x_1(t), x_2(t))^T$ 를 훈련 샘플로 취합하여 원음  $z_1(t), z_2(t)$ 를 찾는 문제.

$x = Az$ 라고 하였을 때 A를 구하는 것. -> 과소 조건 문제(ex : 정수 하나를 주고 어떤 두 수의 곱인지 알아내라)

보통 추가 조건을 이용하여 해를 찾아낸다.

독립성 가정

→ 원래 신호가 서로 독립이라는 가정(음악과 대화는 무관하게 발생한다)

비가우시안 가정



➔ 원래 신호가 가우시안이라면 혼합 신호도가 분류하기 어려운 형태인 원형과 같이 나오기 때문에[ 분리할 실마리가 없는 반면, 비가우시안이면 분리가 가능한 실마리가 존재함.

ICA의 경우

원래 신호의 비가우시안인 정도를 최대화하는 가중치를 구하는 전략을 사용한다.

전처리 수행 후 훈련 집합  $X$ 의 평균이 0이 되도록 이동시킨다.

이후 화이트닝 변환을 적용한 뒤, 식을 풀어 최적의 가중치를 구한다. -- 
$$\mathbf{x}'_i = \left( \mathbf{D}^{-\frac{1}{2}} \mathbf{V}^T \right) \mathbf{x}_i, i = 1, 2, \dots, n$$

#### ICA VS PCA

	ICA	PCA
가우시안의 유무	비가우시안 가정	가우시안 가정
독립 여부	독립성 가정	비상관 가정
모멘트의 횟수	4차	2차
찾은 축의 수직 여부	수직	비수직
용도	블라인드 원음 문제	차원 축소 문제

## 희소 코딩

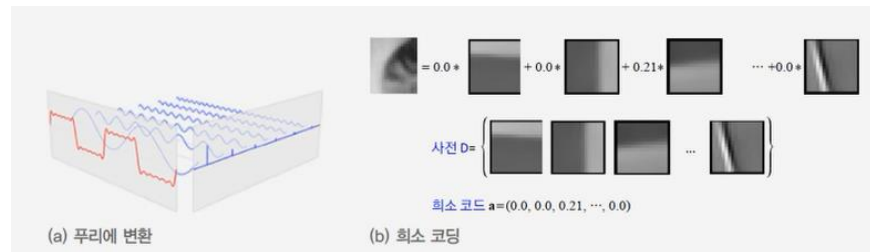
기저함수 또는 기저 벡터의 선형 결합으로 신호를 표현

- 푸리에 변환, 또는 웨이블릿 변환 등

사전  $d$ 를 구성하는 기저 벡터  $d_1, d_2, \dots, d_m$ 의 선형 결합으로 신호  $x$ 를 표현

$x = Da$ , 이때  $D = (d_1, d_2, \dots, d_m)$

### 푸리에 변환과 희소 코딩



- 비지도 학습이 기저 벡터를 자동으로 알아냄(푸리에 변환의 경우 삼각함수 사용)
  - ➔ 희소 코딩은 데이터에 맞는 기저 벡터 사용
- 기저 벡터의 크기를 과잉 완벽하게 책정( $m > d$ )
- 희소 코드  $a$ 를 구성하는 요소 대부분이 0 값을 가진다.

구현 - 최적의 기저 벡터와 희소 코드를 알아내야 함. (희소 코드의 희소성을 강제하는 규제항  $\phi$  포함.)

$$\hat{D}, \hat{A} = \underset{D, A}{\operatorname{argmin}} \sum_{i=1}^n \|x_i - Da_i\|_2^2 + \lambda \phi(a_i)$$

## 6.7 오토인코더

특징 벡터  $x$ 를 입력받아 동일한, 또는 유사한 벡터  $x'$ 를 출력하는 신경망

➔ 단, 단순히 복사하기만 하는 단위 행렬은 오토인코더로 정의하지 않는다.

여러 가지의 규제 기법을 적용하여 유용한 신경망으로 활용하고 있음.

병목 구조 오토 인코더

- $m < d$ 인 구조
- 은닉층의  $h$ 는 훨씬 적은 메모리로, 필요하면 디코더로 원래 데이터를 복원하여 표현
- $h$ 는  $x$ 의 핵심 정보를 표현한다.(→ 특징 추출, 영상 압축 등의 응용)

은닉 노드의 개수에 따라  $m < d, m = d, m > d$  구조

활성함수에 따라 선형(영상 압축 등)과 비선형(영상 특징 추출 등)이 나뉘어져 있음.

훈련 집합  $X$ 를 통해 데이터를 입력받아서 학습, 알아내야 하는 매개변수는  $f$ 와  $g$ 라는 매칭 함수

최적화 문제로 갈 경우  $\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \sum_{i=1}^n L(x_i, g(f(x_i)))$

$\sum_{i=1}^n L(x_i, g(f(x_i))) = \|x_i - g(f(x_i))\|_2^2$ 이므로  $\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \|x_i - g(f(x_i))\|_2^2$

Input – Hidden1 – Hidden2(coding) – Hidden3 – Output

-----Encoding-----||-----Decoding-----

보통 Encoding과 Decoding은 대칭을 이룬다. \* Hidden1과 3은 같은 값을 가진다.

규제 오토 인코더

- 여러 규제 기법을 사용
- ➔  $m > d$ 인 상황에서도 단순 복사한 결과를 피할 수 있음.

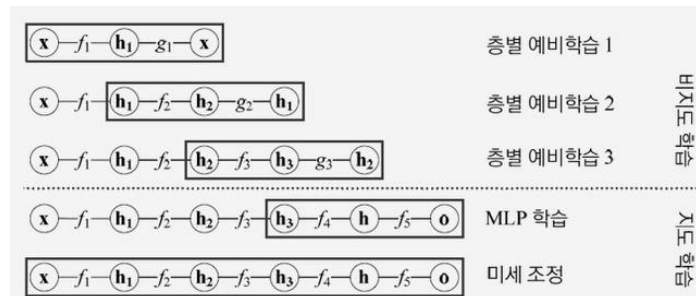
- ➔ 충분히 큰 모델을 사용하되 적절한 규제 기법을 적용하는 현대 기계 학습 추세
- SAE(Sparse AutoEncoder)
  - ➔ 은닉 벡터  $h_i$ 가 희소하도록 강제화(0이 아닌 요소의 개수를 적게 유지함)
  - ➔  $\phi(h_i)$  규제항을 넣어 벡터  $h_i$ 가 희소하도록 강제.
- DAE(Denoising AutoEncoder)
  - ➔ 잡음을 추가한 다음 원본을 복원하도록 학습 시킴
  - ➔ 특징 벡터  $x_i$ 에 적절한 양의 잡음을 추가한  $\tilde{x}_i$ 를 입력으로 사용함.
- CAE(Contractive AutoEncoder)
  - ➔ 인코더함수  $f$ 의 야코비안 행렬의 프로베니우스 shja을 작게 유지함.
  - ➔ 공간을 축소하는 효과를 가짐.

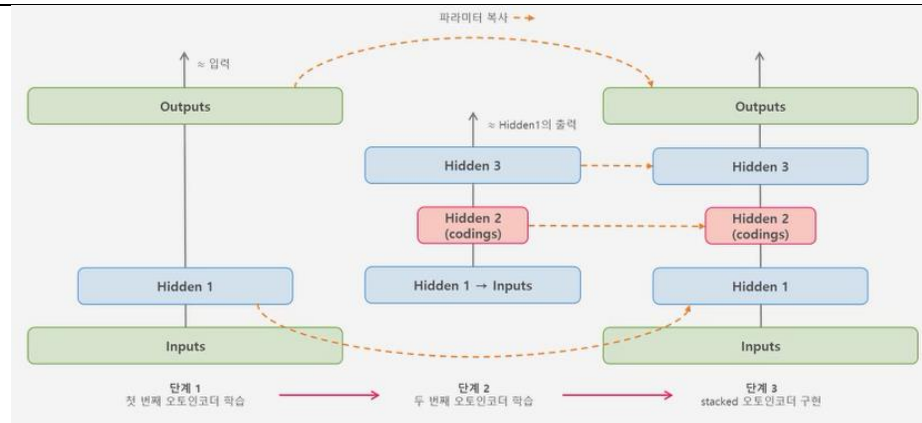
#### 적층 오토인코더

오토 인코더는 얇은 신경망이기 때문에 표현력에 한계가 있음

- ➔ 여러 층으로 쌓아서 용량을 키움.

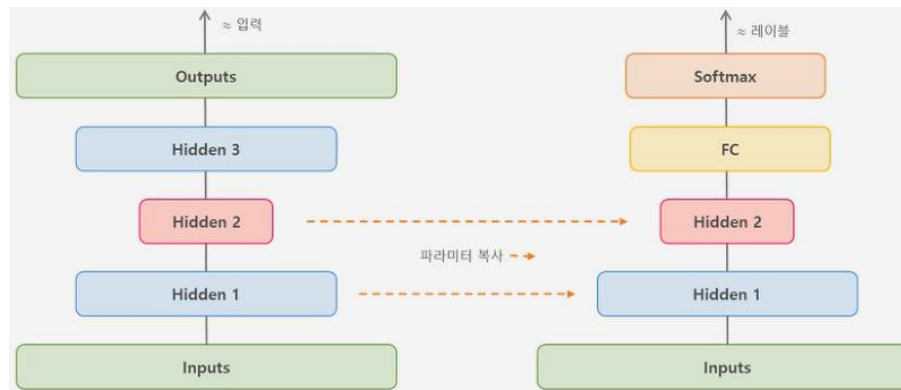
#### 깊은 신경망을 통한 적층 오토인코더의 학습 방법





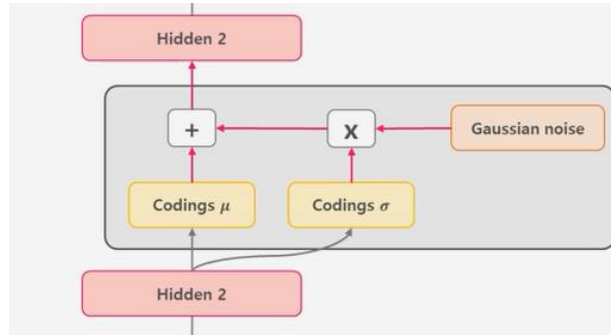
### 학습 과정

1. 층별 예비학습을 필요한 만큼 수행(X만 가지고 비지도 학습)
2. 마지막 층의 출력을 입력으로 하여 MLP 학습(X와 Y를 가지고 지도 학습)
3. 신경망 전체를 한꺼번에 추가로 학습하여 미세 조정



당시에는 기술이 부족했기 때문에 층별 예비학습을 통하여 MLP 학습을 하는 방향으로 나아갔지만, 현재는 여러 기술 향상으로 인해 층별 예비학습이 없어도 DMLP 학습을 하는데 문제가 되지 않는다.

VAEs(Variational AutoEncoder) -> GAN(잡음을 가우시안 모델을 이용하여 추가)



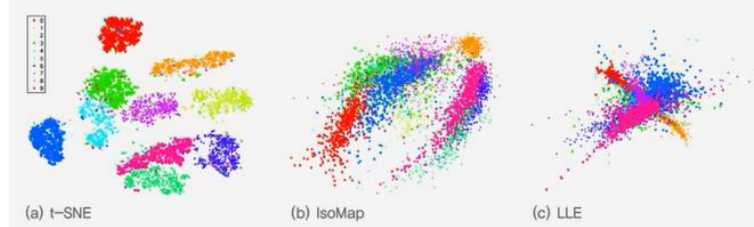
### 6.8 매니폴드 학습

오토인코더는 데이터 구조를 간접적으로 표현한데 비해 매니폴드 학습은 데이터의 비선형 구조를 직접적으로 반영함.

#### 매니폴드

고차원 공간에 내재한 저차원 공간 – 보통 매니폴드는 비선형 공간이지만 지역적으로 살펴보면 선형구조이다.

여러 개의 매니폴드 분류 방법



#### IsoMap

- 최근접 이웃 그래프를 구축하는 형식
  - ➔ 각 점은 k-최근접 이웃을 구하여 거리를  $n \times n$  행렬인  $M$ 에 채운 뒤 빈 곳은 최단 경로의 길이로 채운다
- $M$ 의 고유 벡터를 계산하고 큰 순서대로  $d_{low}$ 개의 고유 벡터를 선택한다
  - ➔ 이들 고유 벡터가 새로운 저차원 공간을 형성한다.
- $M$ 의 크기가 방대한 문제점이 발생.

#### LLE(Locally Linear Embedding)

- 거리 행렬  $M$  대신, 함수  $\epsilon$ 을 최소화 하는 가중치 행렬  $W$ 를 사용
- $x_i$ 를 k-최근접 이웃의 선형 결합으로 근사화하는 셈
- 저차원 공간에서는 변환된 저차원 공간의 점 간의 거리를 최소화 하는  $x'$ 를 찾아야 한다.
- 고차원 원래 공간에서의 점의 거리와 저차원 변환 공간에서의 점의 거리의 비율을 비슷하게 유지함으로써 원 데이터  $x$ 와 변환된  $x'$ 가 유사한 구조를 지닌다.

#### t-SNE(Stochastic Neighbor Embedding)

- 현재 매니폴드 공간 변환 기법 중에서 가장 뛰어난.
- 원래 공간에서 유사도를 선 측정 한 뒤 t-분산을 이용, 분포는 카이제곱 분포를 이용한다.
- 변환된 공간에서의 유사도는 스튜던트 t 분포로 측정.
- 원래 데이터와 변환된 데이터의 구조가 비슷해야 하므로 KL 다이버전스를 사용

	<p>→ 확률분포 <math>P</math>와 <math>Q</math>가 비슷할수록 좋음.</p> <ul style="list-style-type: none"> <li>● 학습 알고리즘은 목적함수 <math>J</math>를 최소화 하는, 즉 <math>P</math>와 <math>Q</math>의 KL 다이버전스를 최소로 하는 <math>X'</math>를 찾는다.</li> </ul> <p>→ 경사하강법을 이용함.</p> <p>Transductive 학습 모델</p> <ul style="list-style-type: none"> <li>● 훈련집합 이외의 새로운 샘플을 처리할 능력이 없는 모델.</li> <li>● IsoMap, LLE, t-SNE는 모두 Transductive 학습 모델에 속한다.</li> <li>● 데이터 가시화에 특화됨(PCA나 AutoEncoder보다 더 뛰어난 성능을 보임)</li> </ul> <p>귀납적 모델</p> <ul style="list-style-type: none"> <li>● 훈련집합 이외의 새로운 샘플을 처리할 능력이 있는 모델</li> <li>● IsoMap, LLE, t-SNE를 제외한 지금까지 공부한 모든(PCA, AutoEncoder를 포함하여) 모델</li> </ul>
질문 내용	