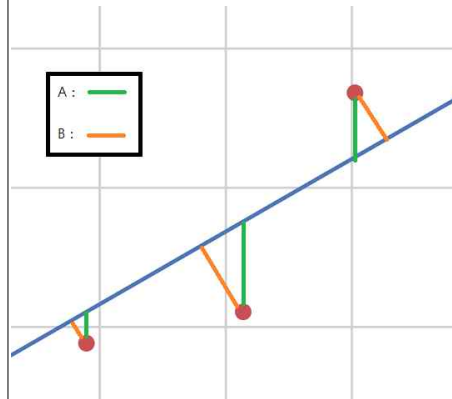


3주차 조별보고서 (Default)

작성일: 2019년 9월 20일		작성자: 김영연		
조 모임 일시: 2019년 9월 20일 9교시		모임장소: 학교 앞 카페		
참석자: 위성조, 이충현, 최진성, 이재은, 김영연		조원: 위성조, 이충현, 최진성, 이재은, 김영연		
구	분	내		용
학습 범위와 내용	1.3	데이터의 중요성과 희소성	1.6	현대 기계 학습에서 데이터 확대와 가중치 감소
	1.4	선형 회귀	1.7	기계 학습의 유형
	1.5	모델 선택의 중요성과 방법	1.8	기계 학습의 역사와 사회적 의미
논의 내용		저희 Default 조는 개인 레포트/질문을 작성한 뒤에, 금요일에 모여 수업시간에 교수님이 말씀하신 손실함수 계산방식에 대한 논의를 제일 먼저 하였습니다. 그 후에 서로의 질문을 맡아 찾아보고 그에 대한 내용을 서로 얘기하는 방식으로 회의를 진행했습니다.		

Q1. 수업 중 손실함수 계산방식에 대한 논의



A1.

1. 먼저 A를 사용하는 것이 효율적이라는 의견은 다음과 같다.

- 손실함수의 계산에서 주로 B대신에 A를 사용하는 이유는 손실함수는 데이터와 모델 사이의 차이를 측정하고, 이를 통해 모델을 어떤 식으로 바꿔야 하는지 결정하는 데 도움이 되기 위한 것이지, 실제 데이터와 모델 사이의 정확한 차이를 구하기 위해서 사용되는 것이 아니므로, 점과 직선 사이의 거리를 구하기 위해 피타고라스 정리를 이용해 제곱과 루트 계산을 하는 것보다 y값의 차이만을 계산하면 되는 비용 효율적인 방법을 택한 것이 아닌가 생각한다.

2. B를 사용하는 것에 대한 의견은 다음과 같다.

실제 값과 예측 값에 대한 오차를 표현할 때, 오차를 B로 사용하면 과연 더 줄어들지 의문이 들었다. 그러나 이는 실제 값과 예측 값에 대한 x값이 같아야 하는데 사선 방향으로 표시되면 x값이 2개가 존재하므로 표시하면 안 된다.

Q2. 진성: 과소적합과 과잉적합은 데이터 집합에 따라 발생할 수 있는 모델의 문제점을 보여주고 있는데, 이를 해결할 방법이 무엇인지에 대해 궁금하다.

A2.

feature를 많이 반영하면 high variance가 되어 발생하는 과잉적합을 해결하기 위해서는 크게 3가지 방법이 있다. 첫번째로는 정규화로 학습데이터의 일부분만 학습을 시키고, 나머지는 학습을 시키지 않고 테스트에 활용하는 방법이다. 두번째는 교차검증이다. 검증용 데이터를 고정하지 않고 랜덤하게 변경해가며 사용하는 기법이다. 여기서 검증용데이터란 아래의 그림과 같이 훈련 데이터를 나누어 사용하는 것 중 한 종류이다.



세번째는 가지치기이다. 가지치기란 가지치기하듯이 특정 가지 이후를 잘라내어 단순하게 만듦으로써 overfitting을 해결하는 것이다.

반면에, feature를 적게 반영하여 high bias가 되어 발생하는 과소적합을 해결하기 위해서는 무조건 더 많은 데이터를 늘려서는 안 된다. 그것은 오히려 에러를 더 증가시킬 수 있기 때문에, feature를 더 많이 반영해서 분산을 높여야 한다. 또는, 분산이 높은 기계학습모델

Decision Tree, KNN, SVM등을 사용하는 방법 등이 있다.

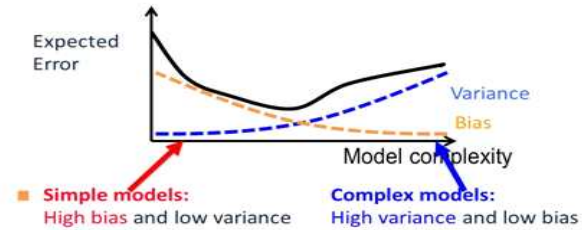
위의 내용을 논의하며, bias와 variance의 trade-off 관계에서 각 학습 모델에 맞는 적절한 지점을 찾는 것이 중요함을 알 수 있다.

Q3. 재은: 과대적합 문제를 해결하기 위해서는 훈련 데이터를 더 많이 모으거나, 정규화를 거치거나, 훈련 데이터의 잡음을 줄이는 등의 방법이 쓰입니다. 그에 반해, 과소적합 문제는 파라미터가 더 많은 모델을 선택하거나 과대적합이 되기 전까지 학습을 시킴으로써 해결합니다. 그렇다면 과대적합과 과소적합 중 더 안 좋은 문제가 무엇인가요? 각각의 문제에서 해결하는 방식에서는 어떤 방식이 더 효율적인지 알고 싶습니다.

A3.

과소적합 및 과대적합 모두 피해야 할 문제임은 분명하나, 굳이 구분하자면 과대적합이 과소적합보다는 나을 수 있다. 그 이유로는 과대적합의 경우 적어도 훈련 데이터셋에 대해서는 높은 성능을 보여주기 때문이다. 과대적합과 과소적합 문제에서 비용문제를 제외한다면 언제나 효율적인 해결책은 훈련 데이터량을 늘리는 것이지만, 비용 문제로 데이터량을 늘리기 힘든 경우, 모델의 복잡도를 우선 조정해 보는 것이 효율적이라고 판단된다.

Model complexity



DECISION TREES

CS446 Fall '15

60

위 그래프에서 볼 수 있듯이, 모델의 복잡도에 따라 에러율이 달라지는데, 평균적으로 bias와 variance 가 만나는 지점에서 에러율이 가장 낮아지므로, 과소 및 과대적합 모두에 적용할 수 있는 방식이므로, 먼저 모델의 복잡도를 변경해 보는 것이 나을 것이다.

그러나, 모델의 복잡도를 변경한 뒤 훈련을 진행하고 테스트를 해야 하므로, 시간이 많이 걸리기에, 돈과 시간 모두 부족한 상황이라면, 다른 방식을 고민해 봐야 할 것 같다.

Q4 영연: 고차원 공간에 내제된 manifold를 찾는 방법은 무엇인지 궁금합니다.

A4.

학습 데이터 셋에 존재하는 수많은 이미지를 고차원 공간 속에 매핑 시키면 유사한 이미지는 특정 공간에 함께 있을 것이다. 그리고 그 점들의 집합을 잘 아우르는 부분공간이 존재하

	<p>는데 그것이 매니폴드라고 한다. 매니폴드는 다음과 같은 가정을 가지고 있다.</p> <ul style="list-style-type: none"> -고차원 데이터의 밀도는 낮지만, 이들의 집합을 포함하는 저차원의 매니폴드가 있다. -이 저차원의 매니폴드를 벗어나는 순간 밀도는 급격히 낮아진다. <p>매니폴드 공간은 본래 고차원 공간의 부분공간이기 때문에 차원수가 상대적으로 작아진다. 이는 데이터 차원 축소가 가능하다고 볼 수 있다. 따라서 차원 축소가 잘 되었다는 것은 매니폴드 공간을 잘 찾았다는 뜻이다. 본래 고차원 공간에서 각 차원들을 잘 설명하는 새로운 특징(feature)을 축으로 하는 공간을 찾았다는 뜻으로 해석할 수도 있다. 매니폴드를 잘 찾으면 이미지들의 의미적인 유사성을 잘 보존할 수 있다. 또한 유사한 데이터를 획득하여 학습에 없는 데이터를 획득할 가능성도 있다.</p>
	<p>Q5. 총현: 수업 중에 의사결정트리(Decision Tree)이론에서 대해 언급했다. 트리를 만들려면 데이터를 모으고 각 데이터의 특징을 추출한다고 들었다. 그리고 트리가 모이지면 숲(Forest)가 되어진다. 그러면 예를 들어서 50개의 작은 의사결정 트리가 모여져서 하나의 큰 의사결정 숲이 된다고 하면, 큰 의사결정 숲이 50개의 작은 의사결정 트리보다 더 효율적일까?</p> <p>A5.</p> <p>의사결정트리는 이해하고 해석하기 쉬운데다가 동시에 숫자형/범주형 데이터를 다룰 수 있지만, 새로운 데이터에 대한 일반화 성능이 좋지 않게 overfitting 되기 쉽다는 점이 있다.</p> <p>또한 의사결정트리가 많이 발생하게 된다면 어떤 것이 최적의 의사결정트리인지 정하기 힘들어진다는 단점이 있다.</p>

	<p>Random Forest는 전체 데이터가 아닌 샘플의 결과물을 랜덤한 순서로 각 트리의 입력 값으로 넣어 학습하는 방식이기 때문에 앙상블 학습 방법이라고도 불리며, 이렇게 구성된 다수의 결정 트리로부터 분류, 또는 평균 예측치를 출력함으로써 동작한다. 즉, 의사결정트리의 단점을 상쇄하기 위해서 고안된 알고리즘이므로 많은 데이터와 많은 의사결정트리가 필요한 작업에는 효율이 뛰어나지만, 데이터가 적고 의사결정트리 또한 많이 필요하지 않은 곳에서는 오히려 데이터의 결과 값이 한 쪽에 너무 치우치거나 몇 개의 의사결정트리를 사용하는 것보다 효율이 떨어질 수 있다.</p> <p>따라서, 위로 미루어 보아 무조건적으로 큰 의사결정트리가 작은 의사결정트리보다 더 효율적이라고 확신할 수 없다 생각되어진다.</p>
질문 내용	<p>1. 과소적합과 과대적합에 대해 논의하던 도중 feature라는 용어를 알게 되었습니다. 과소적합을 해결할 때 무조건 데이터를 증가시키는 것이 아니라 feature를 증가시킨다는 말이 나오는데, 이 말이 그 모델의 특징을 더 많이 나타내는 데이터를 증가시킨다는 것이 맞나요? 그게 맞다면 그 feature가 유용한 feature인지 아닌지 구분은 어떻게 하는지 궁금합니다.</p> <p>2. 가중치 감쇠에서 가중치를 감소시켜 노이즈 및 아웃라이어의 영향을 줄인다는 개념은 이해하겠는데, 그 방식으로 제시되는 L1, L2 정규화의 경우 손실함수의 값을 증가시키는 방향으로 작용한다는 것을 알게 되었습니다. 손실함수의 값이 클수록 모델과 데이터 사이의 괴리가 크다는 것을 의미하고, 이는 모델의 더 큰 변화를 야기할 것 같은데, 어떻게 손실함수 증가가 가중치의 감소를 이끌어 내는지 잘 모르겠습니다.</p>

기타	
----	--

참고

조 운영 지침

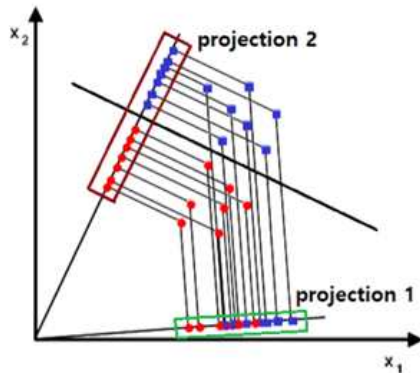
1. 매주 1회 정해진 시간과 장소에 모여서 1시간 정도의 조모임을 갖는다.
2. 조장은 모임 전에 학습할 범위를 조원들에게 통보한다.
3. 각 조원은 학습 범위 내의 교재와 강의 자료를 공부한 후에 이해한 내용과 이해하지 못한 내용을 각각 간단하게 정리하여 개별보고서를 작성한다. (1-2쪽으로 충분함) 작성한 개인 보고서는 모임 전에 모든 조원에게 전송한다.
4. 그 모임의 회의 진행은 순번을 정하여 돌아가면서 진행하고 해당 순번은 조별모임 한 후에 조별보고서를 작성하여 다음 수업시간 전에 과목 웹 페이지에 게시를 한다.
5. 조별 모임에 참석하지 않는다든지 보고서를 작성하지 않는다든지 혹은 지각 등의 조의 단합을 저해하는 조원은 조원들 스스로 학기 초에 정한 규정에 의하여 처리할 수 있다. (벌금 부과나 조 퇴출 등) 이러한 규정들은 조가 결정된 후에 서로 조별로 협의하여 규정을 만들어 제출하며 규정은 계속 개정할 수도 있다. (규정을 소급적용할 수는 없다.)
6. 조별모임을 원하지 않는 사람이나 퇴출된 학생은 다른 조에 동의를 얻어서 합류하거나 보고서 작업을 혼자 진행한다. (조원의 최대 숫자는 학기 초에 정해진다.)
7. 개인 보고서와 조별 보고서 모두 “자료조사” 혹은 교재 내용을 요약 정리하는 것에 중점이 있는 것이 아니라 자신이 혹은 조원들이 잘 모르겠는 것들 이해되지 않는 것들이 무엇인지를 파악하는 데 중점을 둔다.
8. 작성된 조별 보고서는 수업시간 혹은 과목 홈페이지 게시판을 통하여 설명될 것이다.

첨부 개인 레포트

이름		김영연	학번	201500629
구분	내용			
학습 범위	1.3 데이터에 대한 이해 1.4 간단한 기계학습의 예 1.5 모델 선택의 중요성과 방법 1.6 현대 기계 학습에서 데이터 확대와 가중치 감소 1.7 기계 학습의 유형 1.8 기계 학습의 역사와 사회적 의미			
학습 내용	1. Entropy -열역학적 계의 유용하지 않은 에너지의 흐름을 설명할 때 이용되는 상태 함수. -통계역학적으로 주어진 거시적 상태에 대응하는 미시적 상태의 수의 로그 -머신러닝에서의 entropy : 머신러닝에서 분류의 정확도 기준은 데이터 분포의 불순도를 나타내는 'entropy'라는 개념을 이용해서 계산을 한다. 데이터 불순도가 높을수록 entropy의 값이 낮아지고, 불순도가 낮을수록 entropy 값이 높아진다. 2. 데이터 베이스의 품질 -주어진 응용에 맞는 충분히 다양한 데이터를 충분한 양만큼 수집à 추정 정확도가 높아진다. 3. Manifold -Manifold란 고차원 공간에 내재한 저차원 공간을 의미한다. -manifold는 보통 비선형 구조를 가지는데, 특정 점을 중심으로 인근만 살펴보면 선형구조에 가깝다. 예를 들면, 지구는 구의 형태를 지니나 지도를 그릴 때는 평면으로 간주하는 이치와 같다.			

4. Manifold가정

- 고차원의 공간에 주어진 실제 세계의 데이터는 고차원 입력공간에 에 내재한 훨씬 저차원인 manifold의 인근에 집중되어 있다.



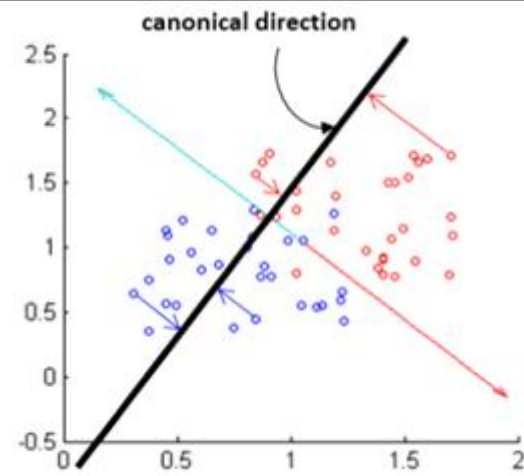
5. fisher's linear discriminant

- classification의 일종으로 n차원의 데이터를 직선상으로 투사시켜 차원을 낮추어 선형 분류 문제로 바꿔주는 방법
fisher교수의 linear discriminant를 하려면 원점을 통과하는 직선을 그어 각 데이터 점들로부터 수선을 그려야 한다.

위의 그림에서 보면 projection1에서는 투영된 점들이 섞여 있으나 projection2에서 보면 점들이 명확하게 구분되어 있음을 알 수 있고 이런 작업은 성공적이라고 할 수 있다.

- canonical direction: 두 그룹의 data가 적게 겹치게 되는 선

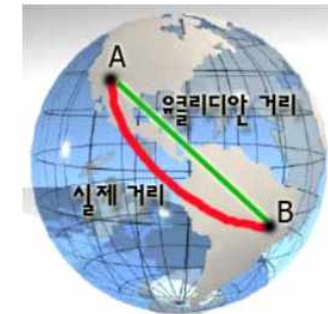
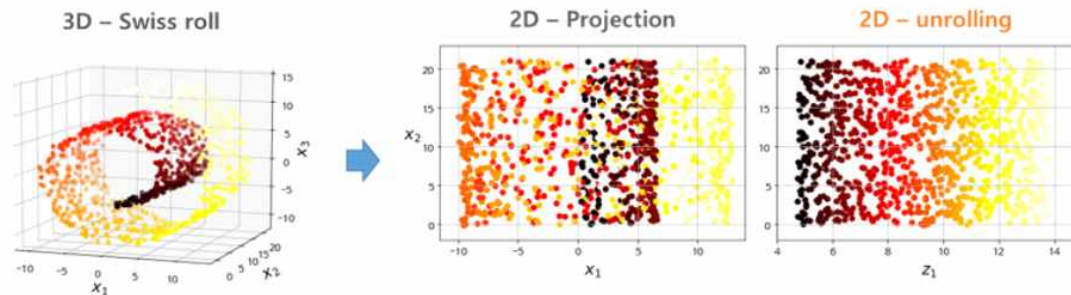
Canonical direction은 두 그룹의 '중점'은 거리가 멀고, 각 그룹 내의 요소들은 가까이 몰려 있어서 variance가 적어야 한다.



질문 내용 고차원 공간에 내제된 manifold를 찾는 방법은 무엇인지 궁금합니다.

이름		이충현	학번	201402665
구분	내용			
학습 범위	1.3 데이터의 중요성과 희소성 1.4 선형 회귀 1.5 모델 선택의 중요성과 방법 1.6 현대 기계 학습에서 데이터 확대와 가중치 감소 1.7 기계 학습의 유형 1.8 기계 학습의 역사와 사회적 의미			
학습 내용	<p>1.3 데이터의 중요성과 희소성</p> <ul style="list-style-type: none"> ● 과학기술의 발전 과정 데이터 수집 -> 모델 정립 -> 예측 상황에 맞는 다양한 데이터를 충분히 수집하면 예측 정확도가 높아진다. EX) Iris, MNIST, ImageNet, Kaggle dataset ● <u>Fisher's linear discriminant에 대한 개념</u> 고차원의 데이터 집합을 선형 결합에 의하여 직선으로 projection 시켰을 때 서로 다른 클래스를 갖고 있는 그룹들이 '직선 위'에서 잘 구분이 되어야 한다. 따라서 구분을 지어 줄 직선을 찾는 게 관건이다. Projection이 될 직선은 그룹을 잘 구분시켜 주는 직선에 대한 직교 방향벡터가 된다. 결론적으로 각 그룹의 중심 거리가 멀어야 하고 각 그룹의 분산이 작아야 한다. ● <u>매니폴드학습</u> 매니폴드 학습이란 고차원상에 뒤틀린 상태로 놓여 있는 매니폴드로부터 원래의 저차원 유클리디안 공간을 구하는 것이 			

다. 매니폴드 학습은 매니폴드 가정 또는 매니폴드 가설에 의해, 고차원인 실제 데이터셋이 더 낮은 저차원 매니폴드에 가깝게 놓여 있다고 가정한다. 매니폴드 가정은 분류나 회귀 같은 작업을 하기전에 학습 데이터셋을 저차원의 매니폴드 공간으로 표현하면 간단하게 해결할 수 있다는 가정을 할 수 있다. 그러나 저차원 매니폴드가 오히려 결정 경계를 찾는 것이 더 어려울 수 있다. 따라서, 모델을 학습시키기 전에 데이터셋의 차원을 감소시키면 학습 속도는 빨라지지만 모델의 성능은 더 낮거나 간단해지는 것은 아니고 데이터셋이 어떠한 모양을 하고 있느냐에 따라 달라진다.



수업 시간에 언급했듯이, 지구를 지도로 보면 2차원이지만 실제로는 구 형태의 3차원 매니폴드에 있다.

- 데이터 가시화

4차원 이상의 초공간은 한꺼번에 가시화 불가능

대표적으로 고차원 공간을 저차원으로 변환하는 t-SNE기법이 있다.

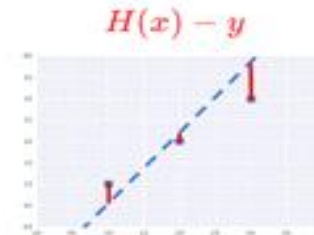
1.4 선형회귀

Cost

How fit the line to our (training) data

$$\frac{(H(x_1) - y_1)^2 + (H(x_2) - y_2)^2 + (H(x_3) - y_3)^2}{3}$$

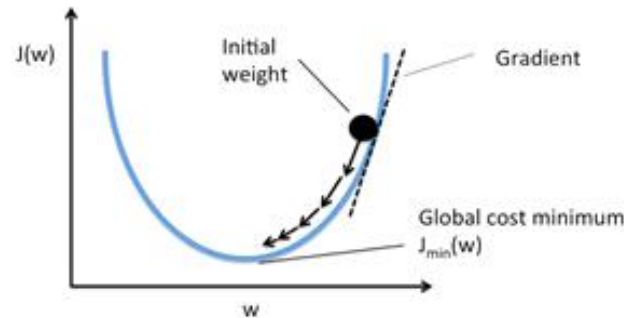
$$\text{cost}(W) = \frac{1}{m} \sum_{i=1}^m (Wx_i - y_i)^2$$



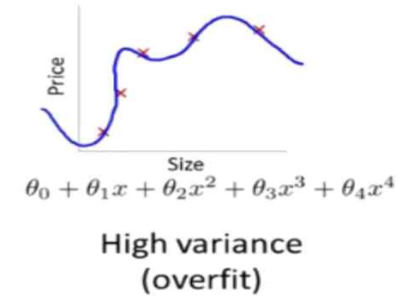
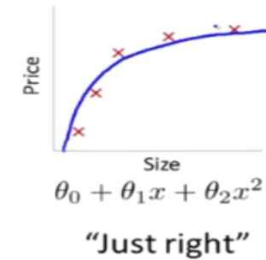
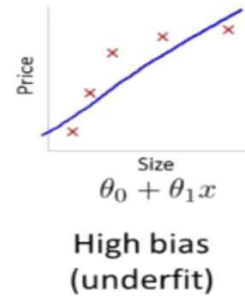
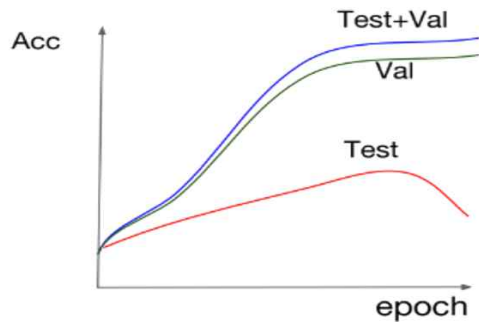
- 여기서 '학습이 잘 되었다'='Error가 적다'='Loss(Cost)가 적다'='y와 데이터(점)간의 거리가 짧다' 라고 볼 수 있다.

학습의 궁극적인 목표는 Loss를 최소화시키기 위함이다. Figure2에서 직선상 거리가 음수가 나올 수 있는데 음수인 거리는 존재하지 않으므로 제곱을 시키고 평균값을 구한다. 이렇게 학습이 마치면 예측에 사용하게 된다. 그리고 그 예측과 실제 값의 차이가 최소가 되면 잘 학습되었다고 본다.

- 손실을 최소화하기 위한 방법에는 경사하강법이 있다. 경사하강법이란 쉽게 말해 Cost Function의 기울기가 최소가 되는 값을 찾는 알고리즘으로 임의의 한 점에서 기울기를 구한 Cost function, 다음 기울어진 방향으로 계속 이동하여 이 최소가 되는 최적의 Parameter를 찾으면 된다.



1.5 모델 선택의 중요성과 방법



- 과소적합은 모델의 용량이 작아 오차가 클 수밖에 없는 현상이다. 해결 방법으로는 파라미터가 많은 복잡한 모델 선택, 모델의 제약을 줄이기가 있다. 과잉적합은 모델이 훈련 데이터에 너무 맞아떨어져 일반성이 떨어진다. 훈련 데이터에 너무 맞춰져 있기 때문에 그 외의 다양한 변수에는 대응하기 힘들다. 해결 방법으로는 훈련 데이터를 더 많이 수집, 차원을 줄이거나 모델의 제약, 드롭-아웃(보통 0.5로 맞추다) 등 다양한 방법을 이용해서 정규화 한다.
- 위의 그래프를 보듯이, 훈련 데이터와 테스트 데이터 말고 별도의 검증(평가) 집합이 있다. 검증 집합은 주로 사용하고 있는 데이터로 사용된다. 따라서 이상적인 훈련은 테스트 집합과 검증 집합 둘 다 사용되는 경우이다. 검증 세트는 반드시 필요하다. 그 이유는 모델을 개발할 때는 항상 모델의 설정을 튜닝하기 때문이다. 예를 들어 층 수, 층의 유닛 수(네트워크 가중치와 구별을 위해 하이퍼파라미터 사용)를 선택한다.

	<p>1.6 규제</p> <ul style="list-style-type: none"> ● 데이터 수집은 비용이 많이 드므로 인위적으로 데이터를 확대한다. 훈련집합에 있는 샘플을 회전 또는 와핑을 통해 변형한다. ● 가중치 감쇠는 개선된 목적함수를 이용하여 가중치를 작게 조절하는 규제 기법이다. 학습 데이터셋의 양을 늘리는 것은 오버피팅(overfitting) 문제를 해결할 수도 있지만, 학습 데이터를 추가로 확보하는 것은 일반적으로 어려운 일이다. 그렇기 때문에, 사용하는 함수의 복잡도를 조정하는 것을 더 선호한다. 구체적으로는 차수를 조정해서 다항식의 복잡도를 조절할 수 있는 것을 확인했다. <p>1.7 기계학습 유형</p> <ul style="list-style-type: none"> ● 지도학습, 비지도 학습, 강화 학습, 준지도 학습 <p>1.8 기계 학습의 역사와 사회적 의미</p> <p>미래의 직업 변화 및 기계가 사람을 지배할지 모른다는 두려움</p>
질문 내용	<p>수업 중에 의사결정트리(Decision Tree)이론에서 대해 언급했다. 트리를 만들려면 데이터를 모으고 각 데이터의 특징을 추출한다고 들었다. 그리고 트리가 모아지면 숲(Forrest)가 되어진다고 했다. 그러면 예를 들어서 50개의 작은 의사결정 트리가 모여져서 하나의 큰 의사결정 숲이 된다고 하면, 큰 의사결정 숲이 50개의 작은 의사결정 트리보다 더 효율적일까??</p>

이름		위성조	학번	201402033
구분	내용			
학습 범위	1.3 데이터에 대한 이해 1.4 간단한 기계 학습 1.5 모델 선택 1.6 규제 1.7 기계 학습 유형			
학습 내용	<p>과학 기술의 발전 과정</p> <p>데이터 수집 -> 모델 정립 -> 예측 -> 데이터 수집의 사이클을 가진다. // 테스트 해 봐야 하는 모델이 여러 개인 경우, 가장 간단한 모델부터 시작해 보는 것이 좋다.</p> <p>데이터베이스의 중요성</p> <p>얼굴 인식 응용에서, 정면을 기준으로 한 얼굴 사진만으로 훈련한 경우, 기울어진 얼굴 사진에 대해서는 인식률이 매우 떨어질 것임. 따라서, 충분히 다양한 데이터를 충분한 양만큼 확보하는 것이 매우 중요함</p> <p>Fisher's linear discriminant</p> <p>분류하려는 집단들이 각각의 평균은 최대한 멀리, 그리고 각 분산은 최대한 작도록 분류 기준을 생성하는 것이 좋다.</p> <p>데이터베이스의 왜소한 크기</p> <p>발생 가능한 모든 경우의 수 보다, 실제 데이터 수가 매우 작은 경우, 실제로 데이터가 발생하는 공간은 매우 작은 부분이라는</p>			

점에 착안하여, 발생 가능성이 매우 낮은 곳은 제외한다

데이터 가시화

4차원 이상의 초공간은 한꺼번에 가시화가 불가능하므로, 2개씩 조합하여 여러 개의 그래프를 그리거나, 고차원 공간을 저차원으로 변환하는 기법들을 이용한다.

목적 함수(Objective function) 혹은 비용 함수(Cost function) -> 평균제곱오차(MSE, Mean Squared Error)

기계 학습은 이를 이용하여, 수치적 방법으로 최적해를 찾아감.

Gradient descent = local minimum은 찾을 수 있으나, 그 값이 Global minimum인지는 알 수 없다.

과소적합 - 데이터에 비해 모델의 용량이 작아 어쩔 수 없이 오차가 큰 상황

Ex) Xor 문제는 1차원 직선 모델로 완벽하게 해결할 수 없다

과잉적합 - 데이터에 비해 모델의 용량이 너무 큰 경우, 잡음까지 학습하여, 훈련 데이터에 대하여 높은 성능을 보이나, 테스트 데이터에 대해서는 성능이 좋지 않음.

적절한 용량의 모델 선택 작업이 필요함.

바이어스와 분산

일반적으로 용량이 작은 모델은 바이어스는 크고 분산은 작으며, 용량이 큰 모델의 경우 바이어스가 작고 분산이 크다.

낮은 바이어스와 낮은 분산을 갖는 모델을 찾는 것이 최종 목표이나, 바이어스와 분산은 트레이드 오프 관계에 있으므로, 균형을 잘 찾는 것이 중요하다.

모델 선택

	<p>훈련집합과 테스트집합과 다른 별도의 검증집합을 가진 상황 - Ideal한 상황이므로 각 집합의 목적에 맞게 사용한다.</p> <p>비용 문제로 별도의 검증집합이 없는 상황 - 훈련집합을 등분하여 학습과 평가 과정을 여러 번 반복한 후 평균 사용</p> <p>부트스트랩 - 훈련집합에서 난수를 이용하여 샘플링한 후 훈련시킨다.(대치 허용)</p> <p>현실에서는 모델의 종류가 매우 많으므로, 경험에 기반하여 큰 틀을 선택한 후, 모델 선택 알고리즘으로 세부 모델 선택</p> <p>현대 기계 학습의 전략 - 용량이 충분히 큰 모델을 선택한 후, 선택한 모델이 정상을 벗어나지 않도록 여러 규제 적용</p> <p>데이터 확대</p> <p>데이터를 더 많이 수집하면 일반화 능력이 향상됨, 그러나 비용이 많이 들기에, 인위적으로 샘플을 변형하여 데이터를 확대한다. (부류 소속이 변하지 않게 주의)</p> <p>가중치 감쇠</p> <p>개선된 목적함수를 이용하여 가중치를 작게 조절하는 규제 기법 - 네트워크의 복잡도에 제한을 두어 과적합되지 않게 한다.</p> <p>지도 방식</p> <p>지도 학습/비지도 학습/강화 학습/준지도 학습/오프라인 학습과 온라인 학습/결정론적 학습과 스토캐스틱 학습/ 분별 모델과 생성 모델</p>
질문 내용	<p>가중치 감쇠에서 가중치를 감소시켜 노이즈 및 아웃라이어의 영향을 줄인다는 개념은 이해하겠는데, 그 방식으로 제시되는 L1, L2 정규화의 경우 손실함수의 값을 증가시키는 방향으로 작용한다는 것을 알게 되었다.</p> <p>손실함수의 값이 클수록 모델과 데이터 사이의 괴리가 크다는 것을 의미하고, 이는 모델의 더 큰 변화를 야기할 것 같은데, 어떻게 손실함수 증가가 가중치의 감소를 이끌어 내는지 잘 모르겠다.</p>

이름	이재은	학번	201502469
구분	내용		
학습 범위	<p>1.3절: 데이터의 중요성과 희소성을 강조한다.</p> <p>1.4절: 선형 회귀를 이용하여 기계학습을 직관적으로 설명한다.</p> <p>1.5절: 모델 선택의 중요성과 방법, 과소적합과 과잉적합을 설명한다.</p> <p>1.6절: 현대 기계 학습에서 매우 중요한 규제 기법으로 데이터 확대와 가중치 감소를 간략히 기술한다.</p> <p>1.7-8절: 기계학습의 유형과 역사를 살펴본다.</p>		
학습 내용	<p># [알파고조 1주차 발표 keyword 中] Convolution 연산과 관련된 Filtering에 대하여</p> <p>Convolution 연산은 아래의 식과 같이 f함수와 g함수를 반전 이동한 값을 곱한 다음, 구간에 대해 적분하여 새로운 함수를 구하는 수학 연산자 합성곱이다.</p> $(f * g)(t) = \int_{-\infty}^{\infty} g(\tau) f(t - \tau) d\tau$ <p>이 연산자가 이미지 프로세싱에서는</p> <div style="text-align: center;"> <p>Input: $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$ Filter: $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ Convolution: $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}$ Feature Map: $\begin{bmatrix} 4 & 1 & 1 & 0 \\ & & & \end{bmatrix}$</p> <p>Calculation: $1*1 + 0*0 + 0*0 + 1*0 + 1*1 + 1*1 + 0*1 + 1*1 + 0*1 = 4$</p> </div>		

	<p>입력데이터에 필터와 Convolution 한 연산 결과로 Feature Map의 결과 데이터 값을 얻으며, 이 연산처리는 간격(Stride)을 지정해 주어 얻고자 하는 Feature Map의 크기에 맞게 조정할 수 있다. 수업시간에 자주 언급되는 CNN은 전통적인 neural network 앞에 여러 계층의 Convolutional 계층을 추가한 것이다. Convolutional 계층을 통해서 입력 받은 이미지에 대한 특징(Feature)을 추출해 그 특징을 기반으로 기존의 뉴럴 네트워크를 이용하여 분류하는 인공지능망으로 합성곱 연산이 매우 중요함을 알 수 있다.</p> <p># 기계 학습에서의 데이터는 생성과정을 모르고 주어진 훈련집합만으로 결과 값을 근사 추정한다. 따라서, 충분히 다양한 데이터를 충분히 수집해야 추정 정확도가 높아진다.</p> <p># 목적함수란 기계학습 모델의 출력 값과 사용자가 원하는 출력 값의 차이, 즉 오차를 말한다. 신경망을 학습할 때 학습 상태에</p> $J(\theta) = \frac{1}{n} \sum_{i=1}^n (f_{\theta}(\mathbf{x}_i) - y_i)^2$ <p>대해 측정하는 하나의 지표로 사용되어진다. 대표적으로 평균제곱오차 가 있다. 처음에는 최적 매개변수 값을 알 수 없고, 훈련을 하면 할수록 오차 값이 적어지며 학습 모델의 성능이 높아진다.</p> <p># 과대적합(Overfitting)은 모델이 훈련 데이터에는 너무 잘 맞지만 일반성이 떨어질 수 있다. 훈련데이터에 대해서는 높은 성능을 보여줄 수 있지만, 그 데이터 값이 아닌 다른 변수가 들어왔을 때는 모델의 복잡도가 필요이상으로 높기 때문에 대응하기 힘들어진다. 반대개념인 과소적합(Underfitting)은 모델이 너무 단순해서 데이터를 제대로 학습하지 못할 때 발생한다.</p>
질문 내용	<p>과대적합 문제를 해결하기 위해서는 훈련 데이터를 더 많이 모으거나, 정규화를 거치거나, 훈련 데이터의 잡음을 줄이는 등의 방법이 쓰입니다. 그에 반해, 과소적합 문제는 파라미터가 더 많은 모델을 선택하거나 과대적합이 되기 전까지 학습을 시킴으로써 해결합니다. 그렇다면 과대적합과 과소적합 중 더 안좋은 문제가 무엇인가요? 각각의 문제에서 해결하는 방식에서는 어떤 방식이 더 효율적인지 알고 싶습니다.</p>

이름	최진성	학번	201403474
구분	내용		
학습 범위	1.3 데이터에 대한 이해 1.4 간단한 기계학습의 예 1.5 모델 선택		
학습 내용	<p>1.3 데이터에 대한 이해</p> <p>과학기술의 발전은 다음의 과정을 따른다</p> <p>데이터의 수집 -> 수집된 데이터를 토대로 모델 정립 -> 모델을 이용하여 앞으로의 데이터를 예측</p> <p>단, 모델이 잘못되었을 경우 자신이 수집한 데이터나 앞으로의 데이터를 예측하는데 실패할 가능성이 높다</p> <p>Ex : 튀코브라헤 – 천동설</p> <p>그렇기 때문에 기계의 경우, 기존에 사용하던 지식 기반의 학습을 이용하게 되면 잘못된 모델이 생성되어 제대로 된 학습을 하지 못할 가능성이 높다.</p> <p>Ex : 원형의 물체 가운데에 구멍이 있다면 모두 다 단추라고 할 수 있는가?</p> <p>데이터베이스 안의 데이터가 많아질수록 데이터베이스의 품질은 높아지고 앞으로의 데이터를 예측할 때의 정확도가 높아진다.</p> <p>Fisher Linear Discriminant</p> <p>데이터 집합을 여러 변수들의 선형 결합으로 표시하였을 때 서로 다른 그룹을 잘 구분할 수 있는 '직선'을 찾는 방법이다.</p>		

이를 Projection이라고 하는데, 데이터 집합들에 직선을 놓고, 그 직선에 데이터들을 정사영 하였을 때 다음과 같은 조건을 따르면 그 직선은 데이터 집합을 잘 분리하였다고 할 수 있다.

1. 두 그룹의 중심 거리가 멀어야 한다.
2. 두 그룹의 분산이 작아야 한다.

CNN -> Convolutional Neural Network

*Fully Connected가 아닌 Convolutional Connected

4차원 데이터를 Fully Connected로 연결하여 평면화 하여 저장할 경우 평면화 과정에서 공간정보(ex : 사진)가 손실될 가능성이 매우 높다.

CNN은 공간정보를 유지한 상태로 학습이 가능하다.

Manifold Learning

매니폴드란 유클리드 공간과 닮았으나 사실은 위상공간인 형태로, 겉으로 보기에는 유클리드 공간과 크게 다를 바가 없다는 것이 특징이다. 하지만 유클리드 공간 상에서 가까운 거리지만 실제 유클리드 거리를 구할 경우 위상공간의 특징으로 인해 상당히 거리가 멀게 나오는 경우가 많다.(지구 전도에서의 가장 가까운 진행 방향과 실제 지구에서의 가장 가까운 진행 방향이 다른 경우)

매니폴드 학습이란 이를 가정하고 진행하는 학습으로, 고차원 상태에서 흩뿌려진 데이터들로부터 저차원의 유클리디안 거리를 구하는 것이다. 매니폴드 학습을 이용하는 이유는 데이터의 압축과 시각화에 최적화 되어 있고 차원의 저주를 해결할 수도 있기 때문이다. 하지만 매니폴드의 구성에 따라 저차원의 형태로 변경하였을 때 성능이 낮아지거나 Projection하는 직선을 찾기가 더 어려울 수 있기 때문에 매니폴드 학습에 앞서 어떠한 형태를 취하고 있는지 확인해봐야 한다.

* 데이터 가시화

4차원 이상의 공간은 한 번에 가시화 할 수 없기 때문에 저차원의 공간을 여러 개 조합하는 방법을 사용하여 여러 개의 그림을 그리는 방법을 사용한다.

Ex : t-SNE

1.4 간단한 기계학습의 예

선형회귀의 경우

기계학습에서의 선형회귀는 비용이 적고 오차가 없을 것을 우선으로 한다. 하지만 대부분의 선형 회귀의 경우에는 단순한 직선이 나 곡선이 아닌, 횡수를 거듭할수록 더 복잡한 선이 나타나는게 특징이다. 그렇기 때문에 선형회귀는 '목적 함수(=비용 함수 = 평균 제곱 오차)를 이용하여 선을 최대한 단순하게 하면서 오차를 줄일 수 있는 진행 경로를 찾는다.

이 때 손실을 최소화하기 위해서 사용하는 방법으로 경사하강법이 있는데, 이 방법은 여러 곳에서도 쓰이는 방법이지만 선형 회귀의 경우에는 목적 함수의 비용이 최소화되는 방향으로 계속 이동한다. 이렇게 비용이 최소가 되는 지점을 찾아서 데이터의 선형 회귀 학습을 진행한다.

1.5 모델 선택

과소적합

비선형 모델을 사용할 때 데이터 집합의 위치에 따라 선의 형태가 달라지는데, 데이터 집합의 양이 적을 경우 횡수를 거듭할 수록 데이터의 양에 비해 모델이 너무 지나치게 복잡해져 오차가 크게 발생하게 되는데, 이를 과소적합이라고 한다.

과잉적합

과잉적합은 모델이 데이터 집합에만 너무 완벽하게 맞추어져 있는 나머지 새로운 데이터 집합이 추가될 경우 해당 데이터 집합을 포함하는 모델을 재구성하는데 드는 비용이 너무 크게 발생하는 문제이다. 이는 데이터 집합이 많을수록, 데이터 모델이 복잡할수록 더 많은 비용이 발생한다.

질문 내용	과소적합과 과잉적합은 데이터 집합에 따라 발생할 수 있는 모델의 문제점을 보여주고 있는데, 이를 해결할 방법이 무엇인지에 대해 궁금하다.
-------	--