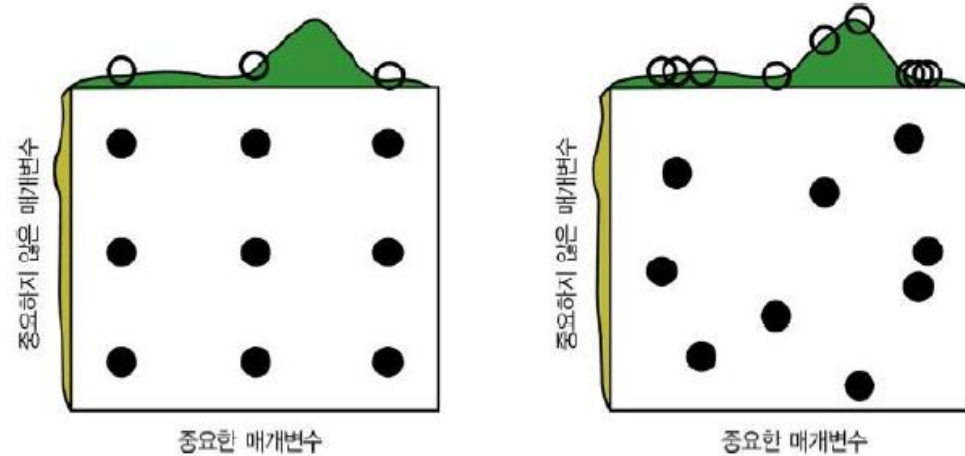


12 주차 조별보고서 (Default)	
작성일: 2019 년 11 월 22 일	작성자: 이충현
조 모임 일시: 2019 년 11 월 22 일 9 교시	모임장소: 학교 앞 카페
참석자: 위성조, 이충현, 최진성, 이재은, 김영연	조원: 위성조, 이충현, 최진성, 이재은, 김영연
구 분	내 용
학습 범위와 내용	<p>5.3절 딥러닝에서의 규제의 필요성과 원리</p> <p>5.4절 많이 쓰이는 규제 기법</p> <p>5.5 절 2 차 미분 정보를 활용한 SGD 를 보완하는 기법</p>
프로젝트 Proposal	<p>주제: KERAS 딥러닝 라이브러리를 사용해서, DCGAN 모델을 이용한 포켓몬 이미지 생성하기.</p> <p>목적: 컴퓨터 비전 분야에서 CNN 을 이용한 지도학습이 많이 채택되어왔다. 하지만 비교적 비지도 학습 CNN 은 주목받지 못했다. 따라서 DCGAN 이라는 네트워크를 사용해 지도 학습과 비지도 학습 간의 성과 차이를 줄일 수 있나 테스트를 해볼 것이다.</p> <p>DCGAN 소개: 다른 비지도 학습 방법들과 비교하기 위해서, 이미지 분류 과제에 학습되었던 식별자를 이용한다(D). GAN 에 의해 학습된 필터들을 시각화 하여 특정한 물체를 그리도록 학습된 특정한 필터를 보여줄 것이다. 생성자(G)가 연산적인 특성으로 인해 쉽게 조작하여 다양한 의미의 샘플을 생성하는 것을 보여준다.</p>

	<p>내용: 포켓몬 데이터셋은 Kaggle 사이트에 올려진 것을 사용 예정. 구현 환경은 Google Colab에서 GPU환경에서 돌릴 예정. 이미지 사이즈도 신경 쓸 필요가 있음.</p> <div data-bbox="994 480 1592 852" data-label="Image"> </div> <p>[포켓몬 이미지 예시]</p>
<p>논의 내용</p>	<p><b>Q1. 격자 학습보다 임의 학습이 더 좋은 효과를 보인다면, 격자 학습이 효과가 더 좋을 경우와 격자 학습을 써야하는 상황은 무엇일까?</b></p> <p><b>A1.</b></p> <p>[Bergstra 2012]에 따르면,</p>



(전략)... With grid search, nine trials only test  $g(x)$  in three distinct places. With random search, all nine trials explore distinct values of  $g$ . ...(후략)

위의 그림에서 격자 탐색의 경우, 각 매개변수에 대해서 3개의 다른 값만 시험해 볼 수 있지만,

임의 탐색은 각 매개변수에 대해 9개의 다른 값을 시험해 볼 수 있으므로 그 성능이 더 낫다고 한다.

위의 결과에 따라, 하이퍼파라미터를 정하는 효과적인 방법이 나오기 전에, 모든 변수의 모든 경우의 수를 탐색할 수 없는 경우 임의 탐색이 더 효율적임을 알 수 있다. 따라서 격자 탐색이 효과적인 경우는 그 반대의 경우, 모든 변수의 모든 경우의 수를 탐색가능한 경우, 혹은 파라미터의 가능한 값 범위가 매우 한정적이어서, 랜덤값으로는 변수값이 다른 경우를 많이 생성하지 못 할 때, 더 효율적일 것으로 생각된다.

	<p><b>Q2. metric과 Norm의 차이에 대하여 알고 싶습니다.</b></p> <p><b>A2.</b></p> <p>Metric은 학습을 통해 목표를 얼마나 잘 달성했는지를 나타내는 척도이다. 여기서는 한 가지 손실 함수만을 다룬다. 훈련을 계속하다 보면 손실 값이 줄어들면서 척도 값도 줄어들지만, 정말 수렴할 때 즈음에 이르러서는 손실 값이 줄어도 척도 값이 줄지 않기도 한다. 이는 둘의 계산식이 다르기에 일어나는 현상이다. 그래서, 훈련 막바지에 검증 데이터에서 손실 값이 줄어드는 것을 모니터링 하지 않고 척도 값이 줄어드는 것을 모니터링 한다.</p> <p>Norm은 벡터의 길이 혹은 크기를 측정하는 방법이다. Norm이 측정한 벡터의 크기는 원점에서 좌표까지의 거리 혹은 크기라고 한다. <math>L_p = (\sum_i^n  x_i ^p)^{\frac{1}{p}}</math> 으로 나타낼 수 있고, p는 Norm의 차수를 의미한다. p가 1이면 L1 Norm이고 p는 2이면 L2 Norm이다. N은 벡터의 요소 수이다. Norm은 각 요소별로 요소 절대 값을 p번 곱한 값의 합을 p제곱근한 값이다.</p>
질문 내용	<p>하이퍼 매개변수를 최적화할 때의 핵심은 하이퍼 매개변수의 '최적 값'이 존재하는 범위를 조금씩 줄어 나가는 것입니다. 이때, 하이퍼 매개변수의 범위는 대략적으로 지정하는 것이 효과적이라 로그 규모 간격을 사용한다 합니다. 그렇다면, 실제로 로그 규모 간격이 쓰이는 경우는 어떤 케이스가 있는지 궁금합니다.</p>

<첨부 개인 레포트>

-201402033 위성조

구분	내용
학습 범위	5.3 규제 의 필요성과 원리 5.4 규제 기법 5.5 하이퍼 매개변수 최적화 5.6 2차 미분을 이용한 최적화
학습 내용	과잉적합에 빠지는 이유 대부분 가지고 있는 데이터에 비해 훨씬 큰 용량의 모델을 사용 – 훈련집합을 단순히 암기하는 과잉적합에 주의 를 기울여야 함.)  현대 기계 학습의 전략 – 충분히 큰 용량의 모델을 설계한 다음, 학습 과정에서 여러 규제 기법을 적용  Ill-posed problem 적절한 가정을 투입하여 문제를 풀 -> 입력과 출력 사이의 매핑은 매끄럽다는 사전 지식  [Deep Learning] 책의 정의 – 일반화 오류를 줄이려는 의도를 가지고 학습 알고리즘을 수정하는 방법 모두  명시적 규제 : 가중치 감쇠나 드롭아웃처럼 목적함수나 신경망 구조를 직접 수정하는 방식 암시적 규제 : 조기 멈춤, 데이터 증대, 잡음 추가, 앙상블처럼 간접적으로 영향을 미치는 방식

$$\underbrace{J_{regularized}(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

규제항은 훈련집합과 무관하며, 데이터 생성 과정에 내재한 사전 지식에 해당

규제항은 매개변수를 작은 값으로 유지하므로 모델의 용량을 제한하는 역할(수치적 용량을 제한)

규제항  $R(\theta)$  - 큰 가중치에 벌칙을 가해 작은 가중치를 유지하려고 주로 L2놈이나 L1놈을 사용

놈

메트릭은 두 점 사이의 관계에 중점을 두나, 놈은 0과 한 점 사이의 관계에 중점을 둠

L0 놈은 벡터  $x$ 의 0이 아닌 성분의 개수 - sparse representation

규제항을 추가한 목적함수의 매개변수를 갱신하는 수식

$$\begin{aligned}\Theta &= \Theta - \rho \nabla J_{regularized}(\Theta; \mathbb{X}, \mathbb{Y}) \\ &= \Theta - \rho(\nabla J(\Theta; \mathbb{X}, \mathbb{Y}) + 2\lambda\Theta) \longrightarrow \underline{\Theta = (1 - 2\rho\lambda)\Theta - \rho\nabla J} \\ &= (1 - 2\rho\lambda)\Theta - \rho\nabla J(\Theta; \mathbb{X}, \mathbb{Y})\end{aligned}$$

-최종해를 원점 가까이 당기는 효과(가중치를 작게 유지함)

$\hat{\mathbf{w}} = (\mathbf{X}^T \mathbf{X} + 2\lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$  로 최적해를 구할 수 있으나,

(normal equation) 일반적으로  $X$ 가 너무 크므로 계산이 어려워, 경사하강법을 이용하여 문제를 해결한다.

$$\left. \begin{aligned} \mathbf{U}^1 &= \mathbf{U}^1 - \rho \frac{\partial J}{\partial \mathbf{U}^1} \\ \mathbf{U}^2 &= \mathbf{U}^2 - \rho \frac{\partial J}{\partial \mathbf{U}^2} \end{aligned} \right\} (3.21) \longrightarrow \begin{aligned} \mathbf{U}^1 &= (1 - 2\rho\lambda)\mathbf{U}^1 - \rho \frac{\partial J}{\partial \mathbf{U}^1} \\ \mathbf{U}^2 &= (1 - 2\rho\lambda)\mathbf{U}^2 - \rho \frac{\partial J}{\partial \mathbf{U}^2} \end{aligned}$$

가중치 규제 적용 전과 적용 후

(L1 높은  $\|\Theta\|_1 = |\theta_1| + |\theta_2| + \dots$ )

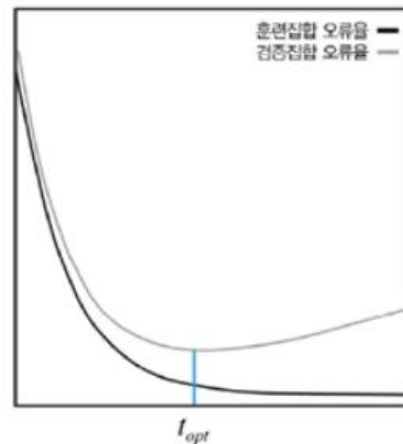
$$\Theta = \Theta - \rho \nabla J_{\text{regularized}}(\Theta; \mathbb{X}, \mathbb{Y})$$

$$= \Theta - \rho (\nabla J(\Theta; \mathbb{X}, \mathbb{Y}) + \lambda \text{sign}(\Theta))$$

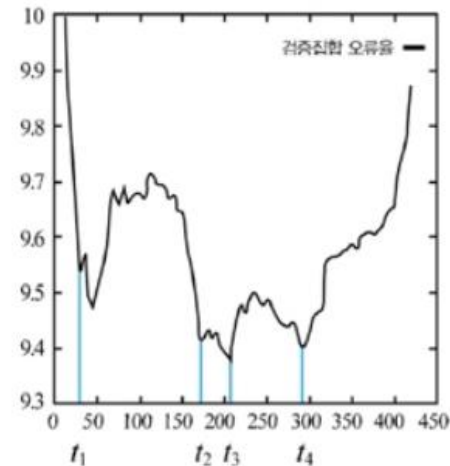
$$= \Theta - \rho \nabla J(\Theta; \mathbb{X}, \mathbb{Y}) - \rho \lambda \text{sign}(\Theta)$$

조기 멈춤

일정 시간이 지나면 과적합 현상이 나타남 -> 일반화 능력 감소, 단순히 훈련 데이터를 암기하기 시작



(a) 개념적인 도표



(b) 실제 데이터에 나타나는 지그재그 현상

그림 5-23 학습 시간에 따른 성능 추이

실제 데이터에서는 성능 증가 추이가 지그재그 모양을 그리므로

단순히 에러율이 높아질 때 멈추는 알고리즘으로는 좋은 결과를 얻기 힘들다

따라서 개선된 알고리즘에서는 참을성 인자  $p$ 를 도입하여, 에러율이 높아지더라도  $p$ 번 진행하고,  $p$ 번의 연산 안에

에러율이 낮아지는 경우, 다시 연산을 최대 p번 진행하는 형식으로 운용한다.

#### 데이터 확대

과잉적합을 방지하는 가장 확실한 방법은 큰 훈련집합을 사용하는 것이나, 데이터 수집은 비용이 많이 든다, 따라서 데이터 확대(인위적으로 변형하여 확대)를 이용한다. / augmentation



모핑을 이용한 확대 -> 비선형 변환으로서 어파인 변환에 비해 훨씬 다양한 형태의 확대  
자연영상 확대, 잡음 섞기.

#### 드롭아웃

입력층과 은닉층의 노드 중 일정 비율을 임의로 선택하여 제거 -> 남은 부분 신경망을 학습  
많은 부분 신경망을 만들고, 예측 단계에서 앙상블 결합하는 기법으로 볼 수 있음  
-실제로는 가중치 공유를 사용

#### 앙상블 기법

서로 다른 여러 개의 모델을 결합하여 일반화 오류를 줄이는 기법  
현대 기계 학습은 앙상블도 규제로 여김

학습 모델에는 두 종류의 매개변수가 있음

내부 매개변수 = 신경망의 경우 엣지 가중치, 학습 알고리즘이 최적화함

하이퍼 매개변수 = 모델의 외부에서 모델의 동작을 조정함/ 은닉층의 개수, CNN 마스크 크기와 보폭, 학습률, 모멘텀과 관련된 매개변수 등



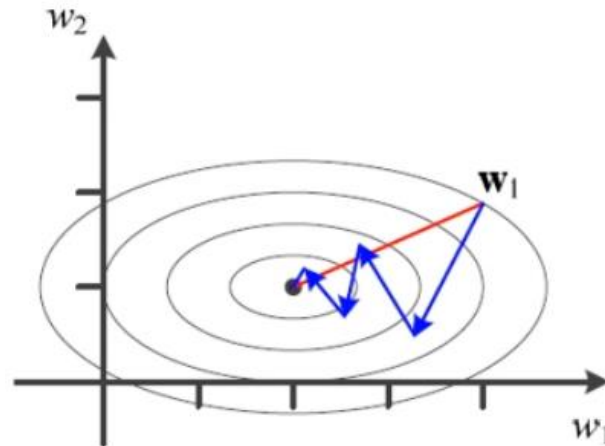
하이퍼 매개변수 최적화

격자 탐색과 임의 탐색 - 임의 탐색은 난수로 매개변수 조합을 생성함

차원의 저주 문제 발생 - 매개변수가 m개이고, 각각이 q개 구간이라면  $q^m$ 개의 점을 조사해야 함.

임의 탐색이 우월

경사 하강의 경우 그레디언트(1차 미분)를 사용함



그레디언트를 이용한 경우 빨간 선이 가리키는 방향은 알 수 없다.

뉴턴 방법

$$\frac{\partial J(w + \delta)}{\partial \delta} \approx J'(w) + \delta J''(w) = 0$$

$$\delta = -\frac{J'(w)}{J''(w)} = -(J''(w))^{-1}J'(w) = \delta = -H^{-1}\nabla J$$

H는 헤시안 행렬(변수들의 이차미분행렬)

매개변수의 개수를 m이라 할 때  $O(m^3)$ 이라는 과도한 계산량 -> 켈레 그레이디언트 방법이 대안으로 제시됨

질문 내용	

-201402665 이충현

구분	내용
학습 범위	5.3절 딥러닝에서의 규제와 필요성과 원리 5.4절 많이 쓰이는 규제 기법 5.5절 2차 미분 정보를 활용한 SGD를 보완하는 기법
학습 내용	<p>&lt;5.3 절 규제의 필요성과 원리&gt;</p> <ul style="list-style-type: none"> <li>● 대부분 가지고 있는 데이터에 비해 훨씬 큰 용량의 모델을 사용한다. 훈련집합을 단순히 암기하는 광잉적합에 주의를 기울여야 한다. 현대 기계 학습의 전략은 충분히 큰 용량의 모델을 설계한 다음, 학습과정에서 여러 규제 기법을 적용한다.</li> <li>● 규제는 모델 용량에 비해 데이터가 부족한 경우의 불량 문제를 푸는 데 사용. 적절한 가정을 투입하여 해결한다 --&gt;단 입출력 사이의 매핑은 매끄럽다는 사전 지식.</li> <li>● 티호노프의 규제 기법은 매끄러움을 가정에 기반을 둔 식</li> </ul>

$$\underbrace{J_{regularized}(\Theta)}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta)}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

- 규제 활용 1)optical flow = 동영상의 프레임에서 픽셀의 움직임 예측, 칼라로 매핑해서 방향벡터로 구분가능. 2)nonlinear image registration = 의료영상, inform 시 어떻게 변하는지, 비교분석 가능하게 해준다.

#### <5.4절 규제 기법>

- 명시적 규제 = 가중치 감소나 드롭아웃처럼 목적함수나 신경망 구조를 직접 수정하는 방식  
암시적 규제 = 조기 멈춤, 데이터 확대, 잡음 추가, 앙상블처럼 간접적으로 영향을 미치는 방식

##### 1) 가중치 벌칙

$$\underbrace{J_{regularized}(\Theta; \mathbf{X}, \mathbf{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta; \mathbf{X}, \mathbf{Y})}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

규제항은 훈련집합과 무관하며 매개변수는 작은 값으로 유지하므로 모델의 용량을 제한하는 역할. 큰 가중치에 벌칙을 가해 작은 가중치를 유지하려고 주로 L1, L2norm사용.

Norm = 선형적, 0일 때 0 벡터이어야 한다.

선형 회귀에 적용.  $\mathbf{Xw} = \mathbf{y}$ 에서 가중치 감소를 적용한 목적함수는,

$$J_{regularized}(\mathbf{w}) = \|\mathbf{Xw} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{w}\|_2^2 = (\mathbf{Xw} - \mathbf{y})^T (\mathbf{Xw} - \mathbf{y}) + \lambda \|\mathbf{w}\|_2^2$$

미분하여 0으로 놓으면,

$$\frac{\partial J_{regularized}}{\partial \mathbf{w}} = \mathbf{X}^T \mathbf{Xw} - \mathbf{X}^T \mathbf{y} + 2\lambda \mathbf{w} = \mathbf{0} \Rightarrow (\mathbf{X}^T \mathbf{X} + 2\lambda \mathbf{I}) \mathbf{w} = \mathbf{X}^T \mathbf{y}$$

L1 norm = 매개변수를 갱신하는 식

$$\Theta = \Theta - \rho \nabla J - \rho \lambda \text{sign}(\Theta)$$

L1 norm의 희소성 효과(0이 되는 매개변수가 많음) = 선형 회귀에 적용하면 특징 선택 효과

## 2) 조기 멈춤

일정 시간이 지나면 과잉적합 현상이 나타나서 일반화 능력을 저하시킨다. 검증집합의 오류가 최저인 점에서 학습을 멈춘다.

## 3) 데이터 확대

과잉적합을 방지하는 가장 큰 확실한 방법은 큰 훈련집합을 사용하는 것이다. 데이터를 인위적으로 변형하여 확대한다.

학습 기반 = 데이터에 맞는 '비선형 변환 규칙'을 학습하는 셈이다.

## 4) 드롭아웃

입력층과 은닉층의 노드 중 일정 비율을 임의로 선택하여 제거한다. '망각'의 개념. 예측 단계에서 앙상블 결합하는 기법으로 볼 수 있다.

앙상블 효과 모방. 가중치에 생존 비율(1-드롭아웃 비율)을 곱하여 전방 계산. 학습 과정에서 가중치가 (1-드롭아웃 비율)만큼만 참여했기 때문이다.

## 5) 앙상블 기법

서로 다른 여러 개의 모델을 결합하여 일반화 오류를 줄이는 기법.

- 서로 다른 예측기를 학습하는 일

- 서로 다른 구조의 신경망 여러 개를 학습 또는 서로 다른 초기값과 하이퍼 매개변수를 설정하고 학습.

- 훈련 집합을 여러 번 샘플링하여 서로 다른 훈련집합을 구성

## 6) 하이퍼 파라미터의 최적화

차원의 저주 문제 발생. 임의 탐색이 우월함.

### <5.5절 2차 미분을 이용한 방법>

- 1차 미분은 현재 위치에서 지역적인 기울기 정보만 알려준다.

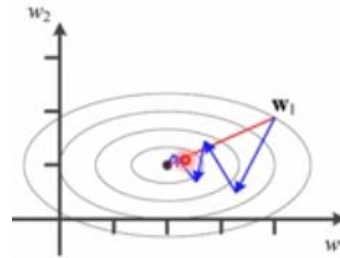


그림 5-31 1차 미분을 사용하는 경사 하강법을 더 빠르게 할 수 있는가

뉴턴 방법은 2차 미분 정보를 활용하여 빨간 경로를 알아냄.

- 기계학습이 사용하는 목적함수는 2차 함수보다 복잡한 함수이므로 한 번에 최적해에 도달하기엔 불가능하다. 뉴턴 방법을 사용해야 한다.
- 켈레 그래디언트 방법 = 직선 탐색은 경사 하강법이 학습률을 직선 탐색으로 찾는 상황이다.

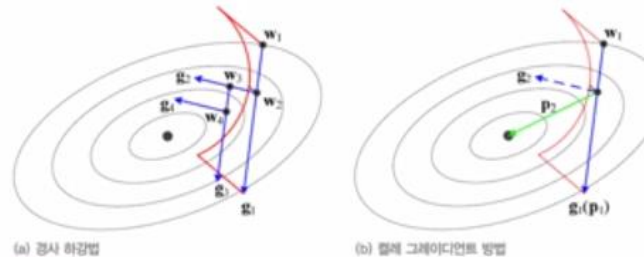
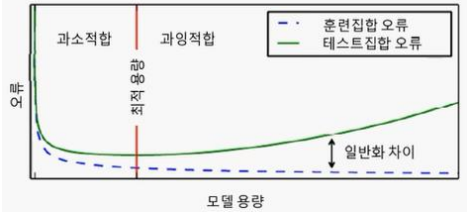


그림 5-33 경사 하강법과 켈레 그래디언트 방법의 비교

질문 내용

구분	내용
<b>학습 범위</b>	<p>기계학습 5장 딥 러닝 최적화</p> <p>5.3 규제와 필요성과 원리</p> <p>5.4 규제 기법</p> <p>5.5 하이퍼 매개변수 최적화</p> <p>5.6 2차 미분을 이용한 방법</p>
<b>학습 내용</b>	<p>기계학습 5장 딥 러닝 최적화</p> <p>5.3 규제의 필요성과 원리</p> <p>과잉 적합에 빠지는 이유 -&gt; 대부분 가지고 있는 데이터에 비해 훨씬 큰 용량의 모델을 사용하기 때문.</p> <p>➔ 훈련집합을 단순히 '암기' 하는 형식이 되어버림.</p> <p style="text-align: center;">학습 모델의 용량과 일반화 능력의 관계</p>  <p>현대 기계 학습은 충분히 큰 용량의 모델을 설계한 다음, 학습 과정에서 여러 규제 기법을 적용한다.</p> <p>규제</p> <p>모델 용량에 비해서 데이터가 부족한 경우의 불량 문제를 해결하기 위한 수단</p>

적절한 가정을 투입하여 문제를 풀

- 입력과 출력 사이의 매핑은 매끄럽다는 사전 지식.
- ➔ '수치적 용량'을 제한하거나 비지도 학습을 이용.
- 오래 전부터 수학과 통계학에서 연구해온 주제.

Optical Flow : 인접한 Frame이 있을 때 Pixel의 움직임이 해당 개체를 어떻게 이동시키는가?

Nonlinear Image Registration : Deform을 시킬 때 해당 Field가 어떻게 변하느냐를 추적.

티호노프의 규제 기법

$$\underbrace{J_{regularized}(\Theta)}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta)}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

↓

$$\underbrace{J_{regularized}(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

(관련 변수가 드러나도록 할 경우)

\* 일반화 오류를 줄이려는 의도를 가지고 학습 알고리즘을 수정하는 방법 모두를 규제라고 할 수 있다.

#### 5.4 규제 기법

- \* 명시적 규제 : 가중치 감소나 드롭아웃처럼 목적함수나 신경망 구조를 직접 수정 방식
- \* 암시적 규제 : 조기 멈춤, 데이터 증대, 잡음 추가, 앙상블처럼 간접적으로 영향을 끼치는 방식

가중치 벌칙

티호노프의 규제 기법 중 관련 변수가 드러나도록 할 경우에 해당

규제항은 훈련 집합과 무관하며 데이터 생성 과정에 내재한 사전 지식  
매개변수를 작은 값으로 유지하게 하므로 모델의 용량을 제한 -> 수치적 용량 제한  
MLP와 DMLP에 적용됨.

$\lambda R(\theta)$ 를 어떤 것으로, 얼마나 설정할 것인가? - 보통 L2놈이나 L1놈을 사용

\* 놈 : 벡터의 크기, 혹은 길이. L1 -> 절댓값의 합, L2 -> 자승의 합(유클리드 놈)

규제식 L2놈의 Gradient 계산

$$J_{regularized}(\theta; X, Y) = J(\theta; X, Y) + \lambda \|\theta\|_2^2 \rightarrow \nabla J_{regularized}(\theta; X, Y) = \nabla J(\theta; X, Y) + 2\lambda\theta$$
$$\therefore \theta = (1 - 2\rho\lambda)\theta - \rho\nabla J$$

\*  $\lambda = 0$ 으로 두면 규제를 적용하지 않은 원래 식  $\theta = \theta - \rho\nabla J$ 이 됨. -> 가중치 감쇠는  $\theta$ 에  $(1 - 2\rho\lambda)$ 를 곱해주기만 함.

- 최종 해를 원점 가까이 당기는 효과.

선형 회귀에 적용할 경우

$$\hat{\mathbf{w}} = (\mathbf{X}^T \mathbf{X} + 2\lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y}$$

공분산 행렬  $X^T X$ 의 대각 요소가  $2\lambda$ 만큼씩 증가

-> 역행렬을 곱하는 셈이 되므로 가중치를 축소하여 원점으로 당기는 효과

규제식 L1놈의 Gradient 계산

$$J_{regularized}(\theta; X, Y) = J(\theta; X, Y) + \lambda \|\theta\|_1 \rightarrow \nabla J_{regularized}(\theta; X, Y) = \nabla J(\theta; X, Y) + \lambda \text{sign}(\theta)$$
$$\therefore \theta = \theta - \rho \nabla J(\theta; X, Y) - \rho \lambda \text{sign}(\theta)$$



→ 부호만 고려한다

- L1놈의 희소성 효과 - 0이 되는 매개변수가 많음

→ 선형 회귀에 적용하면 특징을 선택하는 효과를 보여준다.

#### 조기 멈춤

일정 시간( $T_{apt}$ )이 지나면 과잉 적합 현상이 나타남 -> 일반화 능력이 저하 -> 훈련 데이터를 단순히 암기하는 문제가 발생

조기 멈춤은 검증집합의 오류가 최저인 점  $T_{apt}$ 에서 학습을 멈춘다.

실제 학습은 최저값 갱신이 한 번만 이루어 지지 않으므로 여러 최저값들의 parameter를 저장한 후 가장 낮은 최저값을 기록하는 parameter  $T_{apt}$ 를 선택하여 멈춘다.

#### 데이터 확대

과잉적합을 방지하는 가장 확실한 방법으로, 큰 훈련집합을 사용한다.

-> 데이터 수집은 비용이 많이 든다는 한계가 있음.

- 데이터를 인위적으로 변형하여 확대함 : 자연계에서 벌어지는 잠재적인 변형을 프로그램으로 흉내

Ex : 모핑을 이용한 변형, 자연 현상 확대, 잡음 추가 등

#### 드롭아웃

배치 정규화가 나오기 전까지 많이 사용된 방법으로, 입력층과 은닉층의 노드 중에서 임의로 선택하여 제거하는 방식

- 많은 부분 신경망을 자체적으로 제작하여 예측 단계에서 앙상블 결합 기법을 이용.

→ 계산 시간과 메모리 공간 측면 등, 비용에 대한 부담 문제가 발생.

\* 실제로는 하나의 가중치를 공유하여 사용

#### 양상블

서로 다른 여러 개의 모델을 결합하여 일반화 오류를 줄인다.

서로 다른 예측기를 학습하는 일.

Ex : 배깅(훈련 집합을 여러 번 샘플링 하여 서로 다른 훈련집합을 구성), 부스팅( $i$ 번째 예측기의 오류를  $i+1$ 번째에서 잘 인식하도록 의도적으로 구성)

이전에는 규제 기법으로 보지 않았으나 현대에 와서 규제에 대한 정의가 넓어짐에 따라 양상블도 규제에 포함

#### 5.5 하이퍼 매개변수 최적화

\* 내부 매개변수 : 에지 가중치로서 보통  $\theta$ 로 표기, 학습 알고리즘이 최적화 함

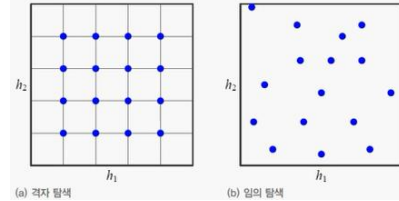
\* 하이퍼 매개변수 : 모델의 외부에서 모델의 동작을 조정. Ex : 은닉층의 개수, CNN 마스크 크기와 보폭, 학습률 등.

#### 하이퍼 매개변수 선택

표준 문헌이 제시하는 기본값을 사용

- 보통 여러 후보 값 또는 범위를 제시하며, 후보 값 중에서 주어진 데이터에 최적인 값을 선택함.
- 하이퍼 매개변수 조합을 생성하는 방법에 따라 수동 탐색, 격자 탐색, 임의 탐색으로 나뉜다.

격자 탐색과 임의 탐색



보통 격자 탐색보다는 임의 탐색이 더 유리하다.

로그 규모 간격

어떤 매개변수는 로그의 규모를 사용해야 한다.

Ex : 학습률 범위가 0.0001 ~ 0.1일 때

- 등간격 : 0.0001, 0.0002, 0.0003, ..., 0.0998, 0.0999, 0.1
- 로그 규모 : 0.0001, 0.0002, 0.0004, 0.0008, ..., 0.0256, 0.0512 ...

매개변수가 m개이고 각각이 q개 구간이라면  $q^m$ 개의 점을 조사해야 함 -> 차원의 저주

5.6 2차 미분을 이용한 방법

\* 1차 미분을 이용한 방법 : 경사 하강법 – 현재 기계 학습의 주류 알고리즘

경사 하강법을 더 빠르게 할 수 있는가에 대해서 연구하다가 발견

뉴턴 방법

테일러 급수를 이용한 방법.

2차 함수에 뉴턴 방법을 적용했으므로 3차 항 이상을 무시한 식을 사용했음에도 최적의 경우를 제시함

식

$$\delta = -\frac{J'(w)}{J''(w)} = -(J''(w))^{-1}J'(w)$$

→  $\delta = -H^{-1}\nabla J$ (변수가 여러 개일 경우로 확장할 경우)

- 기계 학습이 사용하는 목적 함수는 2차 함수보다 더 복잡한 형태이므로 한 번에 최적해에 도달하기는 힘들다
- 반복하는 뉴턴 방법을 사용해야 함.
- $H$ 를 구해야 함.
- 매개 변수의 개수를  $m$ 이라 할 때  $O(m^3)$ 라는 과다한 계산량 요구
- 켈레 그래디언트 방법이 대안으로 제시됨.

켈레 그래디언트 방법

- 직선 탐색 방법

경사 하강법을 이용하여 얻은 해에 특정 값을 이용하여 해까지 일직선으로 도달하게 함

켈레 그래디언트 방법의 식

$$P_t^T H P_{t-1} = 0$$

- $P_t$ 와  $P_{t-1}$ 를 켈레라고 부름.

유사 뉴턴 방법

$H$ 를 직접 구하는 대신  $H$ 의 역행렬을 근사하는  $M$ 을 사용한다.

처음에는 단위 행렬  $I$ 로 시작하여 그래디언트 정보를 이용하여 점점 개선함.(LFGS)

기계 학습에서는  $M$ 을 저장하는 메모리를 적게 쓰는 L-BFGS를 주로 사용

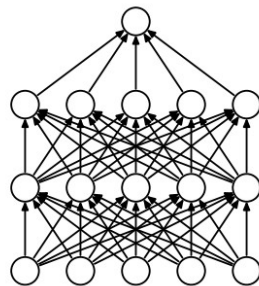
질문 내용	1. 격자 학습보다 임의 학습이 더 좋은 효과를 보인다면, 격자 학습이 효과가 더 좋을 경우와 격자 학습을 써야하는 상황은 무엇일까? 2. 매개 변수의 학습률 간격을 설정할 때 로그 규모 간격을 이용하는 경우는 어떤 경우가 있는가?
-------	--

-201502469 이재은

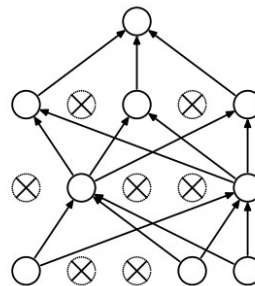
구분	내용
학습 범위	5.3절: 딥러닝에서 규제 of 필요성과 원리를 간략히 기술한다. 5.4절: 많이 쓰는 규제 of 기법으로서 가중치 벌칙, 조기 멈춤, 데이터 확대, 드롭아웃, 앙상블을 설명한다. 5.5절: 하이퍼 매개변수 of 중요성과 최적값을 찾는 방법을 소개한다. 5.6절: 2차 미분 정보를 활용하여 스토캐스틱 경사 하강법을 보완하는 기법을 기술한다.
학습 내용	<ul style="list-style-type: none"> <li>- 딱 적당한 용량을 찾아서 설계하기 보다는, 충분히 큰 용량 of 모델을 설계한다음 여러 규제 기법을 적용하는 것이 현대 기계학습 of 전략이다.</li> <li>- 규제: 일반화 오류를 줄이려는 의도를 가지고 학습 알고리즘을 수정하는 방법 모두</li> <li>- 규제항을 넣지 않으면 복잡한 함수로 mapping을 하는 반면에, 넣으면 차수들의 계수들이 큰 값을 못 갖기 때문에 비교적 부드러운 곡선을 만들어 낸다.</li> <li>- 규제 of 활용: Optical Flow, Nonlinear Image Registration</li> <li>- Optical Flow? 광학 흐름 또는 광학 흐름은 관찰자와 장면 사이의 상대적인 움직임으로 인해 시각적 장면에서 물체, 표면 및 가장자리 of 명백한 움직임 of 패턴</li> <li>- 규제 기법: 가중치 벌칙, 조기 멈춤, 데이터 확대, 드롭아웃, 앙상블 기법 // 명시적 규제와 암시적 규제로 나뉜다. -</li> <li>- 희소특성(sparse representation): 값이 0이 아닌 요소만 저장하는 텐서 of 표현이다. 다시말해, 원-핫 인코</li> </ul>

딩을 통해서 나온 원-핫 벡터들은 표현하고자 하는 단어의 인덱스의 값만 1이고, 나머지 인덱스에는 전부 0으로 표현되는 벡터 표현 방법이다. 이렇게 벡터 또는 행렬(matrix)의 값이 대부분이 0으로 표현되는 방법을 희소 표현(sparse representation)이라고 한다. 즉, 원-핫 벡터는 희소 벡터(sparse vector)이다.

- 드롭아웃(drop out) : 신경망 모델이 복잡해질 때 뉴런의 연결을 임의로 삭제하는 것이다. 훈련할 때 임의의 뉴런을 골라 삭제하여 신호를 전달하지 않게 한다. 테스트할 때는 모든 뉴런을 사용한다.



(a) Standard Neural Net



(b) After applying dropout.

- 앙상블기법: 앙상블 기법은 서로 다른 모델들을 학습해서 개별 모델들에서 나온 출력의 평균을 내어 추론하는 학습 방식이다. 드롭아웃은 학습할 때 뉴런을 무작위로 학습해 매번 다른 모델들을 학습시킨다는 측면에서 앙상블 기법과 유사하다.
- 배치 정규화: 활성화함수의 활성화값 또는 출력값을 정규화(정규분포로 만든다)하는 작업
- 배치 정규화 효과: 학습 속도가 개선된다 (학습률을 높게 설정할 수 있기 때문), 가중치 초기값 선택의 의존성이 적어진다 (학습을 할 때마다 출력값을 정규화하기 때문), 과적합(overfitting) 위험을 줄일 수 있다 (드롭아웃 같은 기법 대체 가능), Gradient Vanishing 문제 해결
- 하이퍼 매개변수 (hyper-parameters): 하이퍼 매개변수란 가중치(weight)같이 모델이 스스로 설정 및 갱신하는 매개변수가 아닌, 사람이 직접 설정해주어야 하는 매개변수를 말한다. 신경망에서는 뉴런의 수,

	<p>배치(batch)의 크기, 학습률(learning rate), 가중치 감소시의 규제 강도(regularization strength) 등이 있다. 이러한 하이퍼 매개변수 값에 따라 모델의 성능이 크게 좌우되기도 한다.</p> <p>하이퍼 매개변수 값은 매우 중요하지만, 사람이 결정해야하는 것이기에 값을 결정하기까지 많은 시행착오를 필요로 한다.</p>
질문 내용	Metric과 norm의 차이에 대해 자세히 알고 싶습니다.

-201500629 김영연

구분	내용
학습범위	<p>5.3 규제의 필요성과 원리</p> <p>5.4 규제 기법</p> <p>5.5 하이퍼 매개변수 최적화</p> <p>5.6 2차 미분을 이용한 최적화</p>

학습  
내용

- 과잉적합에 빠지는 이유: 가중치가 많아도 학습에 쓸 데이터는 충분하지 않은 경우가 많다. 문제의 크기에 비하여 정보가 턱없이 적게 주어진다면 학습 알고리즘은 매개변수 값을 조정하는 일을 반복하다가 결국 주어진 데이터를 단순히 암기하는 상태가 될 수 있다. 이렇게 되면 학습 데이터와 똑 같은 샘플이 들어오면 암기한 것으로 맞추는데 샘플이 조금만 달라진다면 틀릴 가능성이 높아진다.
- 불량 문제: 주어진 데이터를 근사화하는 함수를 구하거나 주어진 데이터로 방정식을 푸는 등의 문제에서 모델 용량에 비해 데이터가 부족한 경우
- 불량문제를 해결하는 방법: 티호노프 방법- 티호노프 방법은 입력과 출력 사이의 매핑은 매끄럽다는 성질을 사용하

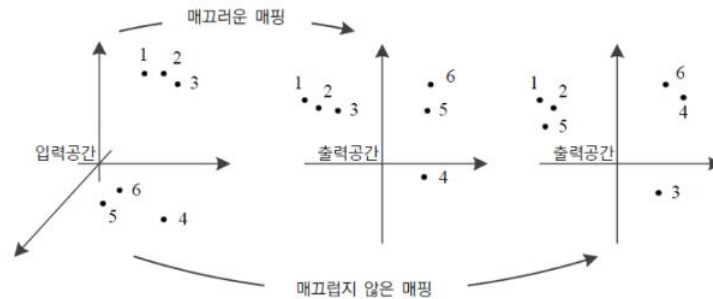


그림 5-20 사전 지식으로서 매끄러움의 특성

였다.

가운데 매핑을 보면, 입력 공간에서 서로 거리가 가까운 샘플은 출력 공간에서도 가깝다는 사실을 알 수 있다. 따라서 매끄러운 함수라고 할 수 있다. 그러나 오른쪽의 매핑을 보면 입력공간에서 서로 가까운 샘플이 출력 공간에



서 면 경우이므로 매끄럽지 않다고 할 수 있다.

$$\underbrace{J_{\text{regularized}}(\Theta)}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta)}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

규제항은 해의 매끄러운 정도를 나타내고, 덜 매끄러울수록 큰 값을 부여하여 벌칙을 강화한다.

- 가중치 벌칙

$$\underbrace{J_{\text{regularized}}(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{R(\Theta)}_{\text{규제 항}}$$

규제항 R은 훈련집합과 무관하며, 데이터 생성 과정에 내재한 사전 지식에 해당하고, 매개변수 값을 작은 값으로 유지하므로 모델의 용량을 제한하는 역할을 한다.

- L2 놈: 규제항 R로 가장 널리 쓰이는 것은 L2놈이며, 이를 사용하는 규제 기법을 가중치 감쇠 기법이라고 한다.

$$\underbrace{J_{\text{regularized}}(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{규제를 적용한 목적함수}} = \underbrace{J(\Theta; \mathbb{X}, \mathbb{Y})}_{\text{목적함수}} + \lambda \underbrace{\|\Theta\|_2^2}_{\text{규제 항}}$$

그래디언트 식

$\nabla J_{regularized}(\Theta; \mathbb{X}, \mathbb{Y}) = \nabla J(\Theta; \mathbb{X}, \mathbb{Y}) + 2\lambda\Theta$  그레디언트를 이용하여 매개변수를 갱신하는 식은 다음과 같다.

$$\Theta = (1 - 2\rho\lambda)\Theta - \rho\nabla J$$

람다를 0으로 설정하면 규제를 적용하지 않은 원래 식  $\Theta = \Theta - \rho\nabla J$ 이 된다.  $\rho$ 는 학습률이고 람다는 L2 놈 계수인데 학습률은 보통 1보다 훨씬 작은 수를 이용하므로  $2\rho\lambda$ 는 1보다 작은 수가 된다. 그리고  $2\rho\lambda$ 만큼 매개변수를 축소 한 후에  $-\rho\nabla J$ 를 더하는 효과를 준다. 이러한 과정을 반복하면 최종해를 원점에 당기는 효과를 준다.

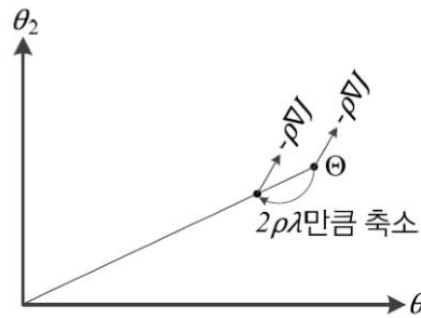
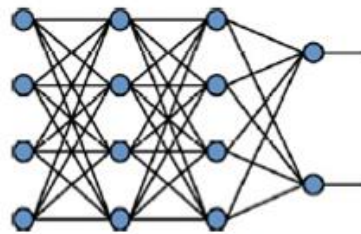


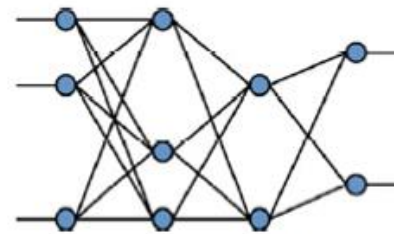
그림 5-21 L2 놈을 사용한 가중치 감쇠 기법의 효과

- 조기 멈춤: 학습을 오래 시킬수록 더 최적점에 접근할 수 있다. 하지만 어떤 지점을 넘어서면 훈련데이터를 암기하여 일반화 능력이 떨어지기 시작한다. 따라서 일반화 능력이 최고인 지점, 즉 검증집합의 오류가 최저인 지점을 만나면 그곳에서 멈추는 전략이 매우 효율적이다. 이 전략을 조기 멈춤이라고 한다.

- 데이터 확대: 과잉적합을 방지하는 가장 확실한 방법은 충분히 큰 훈련집합을 사용하는 것이다. 하지만 데이터를 늘리는 일은 현실적으로 불가능하거나 어려운 경우가 대부분이다. 비용을 적게 들이며 데이터 양을 늘리는 한가지 현실적인 방안은 현재 가진 데이터를 인위적으로 변형하는 것이다. 데이터 확대는 잠재적인 변형을 프로그램으로 구현하여 샘플의 수를 강제로 늘리는 기법이라 정의할 수 있다.
- 드롭아웃: 입력층과 은닉층의 노드 중 일정 비율을 임의로 선택하여 제거하는 작업이다. 선택된 노드는 자신에게 들어오는 에지와 자신에게서 나가는 에지까지 모두 제거한다. 드롭아웃 기법에서는 제거하고 남은 부분 신경망으로 학습을 진행한다. 따라서 서로 다른 부분 신경망을 아주 많이 만드는 셈이다.



(a) 원래 신경망(4-4-4-2 구조)



(b) 드롭아웃된 3개의 신경망 예시

그림 5-27 드롭아웃된 신경망

- 앙상블 기법

서로 다른 여러 개의 모델을 결합하여 일반화 오류를 줄이는 기법

앙상블 기법에서 서로 다른 예측기를 제적하는 방법

1. 모델평균: 주로 여러 모델의 출력으로부터 평균을 구하거나 투표하여 최종 결과를 결정하는 방법

2. 베깅: 성능 측정을 위한 부트스트랩 기법을 앙상블 기법으로 확장한 것이다.

앙상블 기법은 계산 시간과 메모리를 추라고 지불하여 성능 향상을 얻고자 할 때 적용하면 효과적이다.

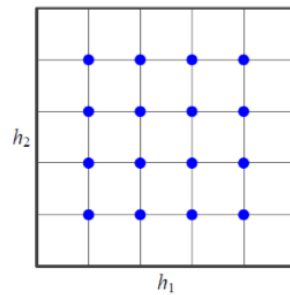
- 하이퍼 매개변수 최적화

모델의 외부에서 모델의 동작을 조정하는 매개변수

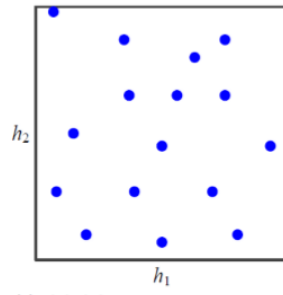
하이퍼 매개변수는 학습을 시작하기 전에 미리 설정해야 하는데 적절한 값을 선택하는 일이 매우 중요하다. 학습률이라는 하이퍼 매개변수의 값을 너무 크게하면 오버슈팅이 일어나 수렴하지 못할 수 있고, 너무 작으면 느리게 수렴하는 문제가 있을 수 있다.

격자 탐색: 각각의 매개변수를 일정한 간격으로 나눈 다음, 교차점에 해당하는 값을 사용한다.

임의 탐색: 난수를 이용하여 매개변수 조합을 생성한다.



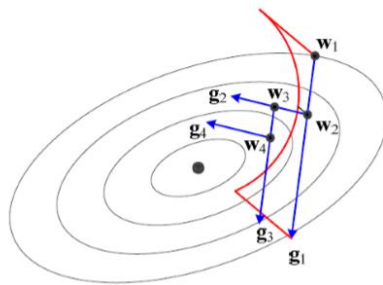
(a) 격자 탐색



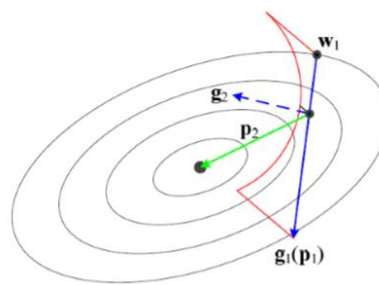
(b) 임의 탐색

- 켈레 그래디언트 방법

직선 탐색: 경사하강법이 학습률을 직선 탐색으로 찾는 상황을 예시



(a) 경사 하강법



(b) 켈레 그래디언트 방법

질문

내용