**2020년도 2학기 기초인공지능 정리**

고려대학교 빅데이터융합과

2019512004 박연정

**Chapter 01. 인공지능의 개념**

1. 인공지능의 개념
   1. Intelligence의 의미
      1. Intelligence는 인간이 가진 자질
      2. 어떤 사람이나 어떤 것이 사물을 배우고 이해할 수 있게 하고, 문제를 해결하고 결정을 내릴 수 있게 해주는 기관이 있어야 함

> Intelligence를 '배우고 이해하는 능력, 문제를 해결하고 결정을 내리는 능력'으로 정의

* 1. 개발자에게 맞는 '인공지능'의 정의
     1. 개발자 관점의 인공지능 = 사람처럼 행동하도록 만들어진 장치(또는 소프트웨어)
     2. 과거의 인공지능 붐에는 사람의 눈앞에 제공되는 것은 자동 제어의 결과물

1. 인공지능의 발전
   1. 암흑기: AI의 탄생(1943년 ~ 1956년)
   2. AI의 융성 : 큰 기대의 시대(1956년 ~ 1960년대 후반)
   3. 이행되지 않은 약속 : 현실의 직면(1960년대 후반~1970년대 초반)
   4. 전문가 시스템의 기술 : 성공의 열쇠(1970년대 초반~1980년대 중반)
   5. 기계가 학습하는 법 : 신경망의 재탄생(1980년대 중반~)
   6. 2000~2010년: 통계 기반 머신러닝과 분산 처리 기술의 발전
   7. 2010년 이후: 심층 신경망 기반 이미지 인식 성능 향상과3 차 인공지능 붐

**Chapter02. 규칙 기반 모델**

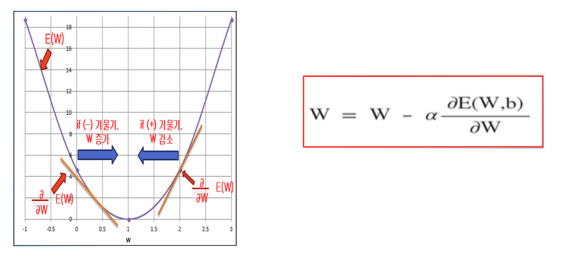
1. 규칙 기반 모델
   1. 조건 분기 프로그램과 규칙 기반 시스템
      1. 규칙(조건 설정)을 사용해 조건 분기 프로그램을 실행하는 시스템을 규칙 기반 시스템
   2. 규칙 설계
      1. 조건 분기의 기반이 되는 규칙은 사람이 미리 결정하며, 규칙을 설계해 나가는 단계에서 문제와 해법을 명확히 함
2. 지식 기반 모델
   1. 지식 기반 모델
      1. 규칙 기반 모델의 규칙이 늘어나거나 변하는 경우, 규칙을 변경했을 때 일일이 수정해야 하는 번거로움 발생
      2. 규칙이나 조건을 설정하는 데이터 세트와 데이터 세트를 처리하거나 출력하는 프로그램 분리하며 이렇게 분리된 데이터 세트를 지식 기반(Knowledge base)이라 함
3. 전문가 시스템
   1. 전문가 시스템의 기본적 특성
      1. 전문가 체계는 좁고 전문화된 영역에서 인간 전문가 수준에서 수행하도록 구축  
         > 전문가 시스템의 가장 중요한 특징은 고품질의 성능
   2. 규칙 기반 전문가 시스템의 추론
      1. 규칙 기반의 전문가 시스템에서 도메인 지식은 IF-THEN 생산 규칙 집합으로 표현되고 데이터는 현재 상황에 대한 일련의 사실 집합으로 표현
      2. 추론엔진은 지식기반에 저장된 각 규칙과 데이터베이스에 포함된 사실 비교
      3. 규칙의 IF(조건) 부분이 사실과 일치할 경우, 규칙이 실행되고 그 THEN(조치) 부분 실행
      4. 데이터베이스 및 지식기반에서의 서신은 상황이나 개념을 나타내기 위해 사용
      5. 규칙 IF 파트를 사실에 일치시키면 추론 체인 생성됨
      6. 추론 체인은 전문가 시스템이 규칙을 적용하여 결론을 도출하는 방법을 나타냄

**Chapter03. 데이터의 표현**

1. 패턴인식, 머신러닝, 딥러닝
   1. 패턴 인식(Pattern Recognition)
      1. 인공지능과 인지과학(Cognitive Science) 분야에 속하는 문제 중 하나
      2. 계산이 가능한 기계적인 장치(컴퓨터)가 어떠한 대상을 인식하는 문제
   2. 머신러닝(Machine Learning, 기계 학습)
      1. 인공지능의 한 분야, 컴퓨터가 학습할 수 있도록 하는 알고리즘과 기술을 개발하는 분야
      2. 초기 머신러닝은 사람의 신경세포를 흉내내는 방식과 수학적인 방식으로 나뉘어 발전
      3. Supervised learning
         * 입력 값(x)과 정답(t, label)을 포함하는 training data를 이용하여 학습하고, 학습된 결과를 바탕으로 미지의 데이터(Test data)에 대한 미래 값을 예측(Prediction)하는 방법
      4. Unsupervised learning
         * 비도지도학습(Unsupervised learning)은 Training data에 정답은 없고, 입력데이터만 사용
      5. 강화 학습(Reinforcement learning)
         * 행동심리학에서 영감을 받았으며, 어떤 환경 안에서 정의된 에이전트가 현재의 상태를 인식하여, 선택 가능한 행동들 중 보상을 최대화하는 행동 혹은 행동 순서를 선택하는 방법
   3. 딥러닝(Deep Learning)
      1. 기계학습의 한 분야로, 여러 비선형 변환기법의 조합을 통해 높은 수준의 추상화를 시도하는 기계학습 알고리즘의 집합
      2. 딥러닝은 기계학습 알고리즘 중 하나인 인공신경망 모델을 적극 이용

**Chapter04 데이터 특징 공간과 문제 해결**

1. 데이터 특징 공간
   1. 인공지능 데이터 분석 시스템
      1. 주어진 데이터를 비교해 원하는 결과를 도출
      2. 다수의 데이터에서 어떤 결과를 예측하는 경우: “어떤 2개의 항목 비교”
   2. 선형문제와 비선형문제
      1. 점의 분포를 선형 함수의 제약과 조건을 이용해 구할 수 있는 문제를 선형 계획 문제
      2. 선형 함수를 이용해 점의 분포를 여러 개 그룹으로 나눌 수 있을 때를 “선형 분리할 수 있다” 라고 함
2. Linear regression
   1. 정의
      1. Training data를 이용하여 데이터의 특성과 상관관계 등을 파악하고 그 결과를 바탕으로 미지의 데이터(Test data)가 주어졌을 때 그 결과를 연속적인 값(숫자)으로 예측하는 것  
         예) 공부 시간과 시험성적 간의 관계
   2. 학습
      1. Training data의 특성을 가장 잘 표현할 수 있는 가중치 W(기울기) 바이어스 b(y절편)를 찾는 것이 학습
   3. 오차(error), 가중치(weight) w, 바이어스(bias) b
      1. Training data의 정답(t)과 직선 y = Wx + b 값의 차이인 오차(error)
      2. 오차가 작다면 직선의 가중치와 바이어스 값이 잘 설정된 것 것이기 때문에 미래 값 예측도 정확할 수 있다고 예상할 수 있음
   4. 손실함수(cost function)
      1. 손실 함수는 Training data의 정답(t)과 입력(x)에 대한 계산 값 y의 차이를 모두 더해 수식으로 나타낸 것
      2. 손실 함수 E(W, b)가 최소값을 갖도록 (W, b)를 구하는 것이 최종 목적
3. Linear regression – Gradient decent algorithm
   1. 경사하강법(Gradient descent algorithm)
      1. W에서의 직선의 기울기인 미분 값을 이용하여 그 값이 최소가 되는 방향으로 손실함수 최소값을 찾는 방법
      2. W에서의 편미 𝜕E(W) / 𝜕W) 해당 W에서 기울기를 나타냄



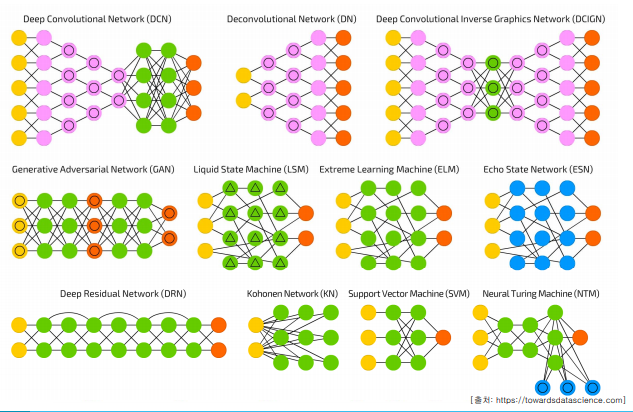
1. Logistic Regression
   1. Classification
      1. Training data 특성과 관계 등을 파악한 후에 미지의 입력 데이터에 대해서 결과가 어떤 종류의 값으로 분류될 수 있는지를 예측
   2. 시그모이드(Sigmoid) 함수
      1. 출력 값 y가 1또는 0 미만을 가져야 하는 분류 모델에서 함수 값으로 0~1 사이의 값을 가지는 sigmoid 함수를 사용
   3. 손실함수, W, b
      1. 최종 출력 값 y는 Sigmoid 함수에 의해 논리적으로 1 또는 0 값을 가지기 때문에 연속 값을 갖는 선형 회기 때와는 다른 손실 함수 필요
      2. 최종 출력 값 y는 sigmoid 함수에 의해 0~1 사의 값을 찾는 확률적인 분류 모델

**Chapter05. 기계학습 이해**

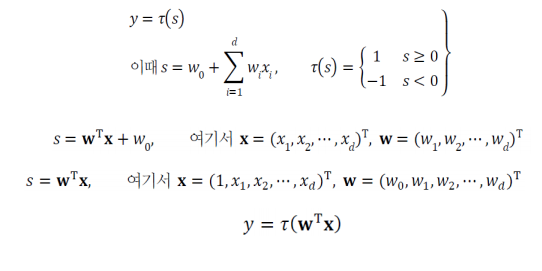
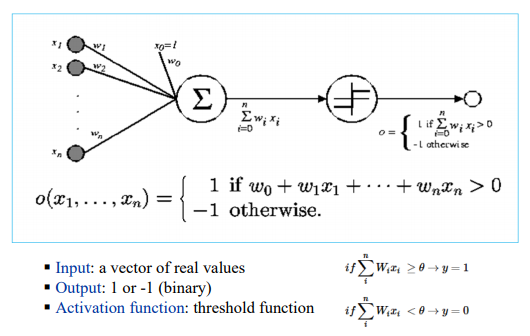
* 1. 인공지능
     1. 기계로부터 만들어진 지능이며, 컴퓨터 공학에서 이상적인 지능을 갖춘 존재
     2. 주로 미리 정의된 규칙의 모음을 이용해서 지능을 흉내내는 컴퓨터 프로그램을 개발
  2. 패턴 인식(Pattern Recognition)
     1. 인공지능과 인지과학(Cognitive Science) 분야에 속하는 문제 중 하나
     2. 계산이 가능한 기계적인 장치(컴퓨터)가 어떠한 대상을 인식하는 문제
  3. 머신러닝(Machine Learning, 기계 학습)
     1. 인공지능의 한 분야, 컴퓨터가 학습할 수 있도록 하는 알고리즘과 기술을 개발하는 분야
     2. 초기 머신러닝은 사람의 신경세포를 흉내내는 방식과 수학적인 방식으로 나뉘어 발전
  4. 딥러닝(Deep Learning)
     1. 기계학습의 한 분야로, 여러 비선형 변환기법의 조합을 통해 높은 수준의 추상화를 시도하는 기계학습 알고리즘의 집합
     2. 딥러닝은 기계학습 알고리즘 중 하나인 인공신경망 모델을 적극 이용
  5. 머신러닝
     1. 지도학습
     2. 비지도학습
     3. 강화학습

**Chapter06. 신경망과 딥러닝**

1. 신경망
   1. 신경망 모델



* 1. Perceptron 구조
     1. 입력층과 출력층을 가짐
     2. 입력층은 연산을 하지 않으므로 퍼셉트론은 단일 층 구조라고 간주
     3. 출력층은 한 개의 노드
  2. Perceptron 동작



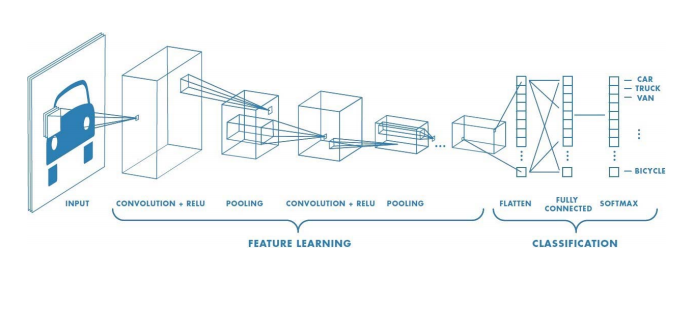
* + 1. 해당하는 특징 값과 가중치를 곱한 결과를 모두 더하여 s를 구하고, 활성함수 𝜏를 적용
    2. 활성함수 𝜏로 계단함수를 사용하므로 최종 출력 y는 +1 또는 -1
  1. perceptron 학습
     1. 전체 학습 데이터에 대한 전체 오차를 최소화하는 방향으로 연결 강도를 갱신  
        > Gradient Descent Method 이용
  2. Multi-layer perceptron
     1. 퍼셉트론은 선형 분류기라는 한계 - 선형 분리 불가능한 상황에서는 일정한 양의 오류
     2. 핵심 아이디어 - Hidden layer
     3. Sigmoid activation function - 연속적인 출력 값
     4. Backpropagation 알고리즘 사용
        + 여러 층이 순차적으로 이어진 구조
        + 역방향으로 진행하면서 한 번에 한층씩 그래디언트 계산하고 가중치 갱신
  3. MLP(Multi-layer perceptron) 기술적 문제 및 해결책
     1. MLP 시대의 한계
        + 2 계층을 넘는 MLP는 잘 학습되지 않음
        + Overfitting, Local optima가 많은 탐색공간
        + 매우 느리고 고차원 데이터 처리가 잘 되지 않음
        + High dimensionality, Slow learning time
     2. 해결책
        + Vanishing gradient 문제임이 알려짐 > ReLU
        + Dropout, Convolution (weight sharing), Max pooling
        + GPU, Big data

1. 딥러닝
   1. 딥러닝 알고리즘 분류
      1. Unsupervised Learning을 기반으로 한 방법  
         e.g., Deep Belief Network, Deep Auto-encoder
      2. Convolutional Neural Network의 다양한 변형들
      3. 시계열 데이터를 위한 Recurrent Neural Network: e.g. Long-Short Term Memory (LSTM)
      4. Deep Q-Network
   2. Deep MLP 구조와 동작
      1. 깊은 MLP(DMLP, deep MLP)의 구조
      2. 은닉층을 깊게(Deep) 할수록 정확도가 높아진다는 의미에서 딥(Deep)이란 용어 사용
   3. Activation function

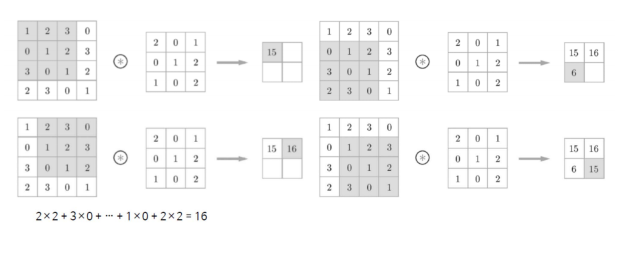
|  |  |  |
| --- | --- | --- |
| Sigmoid  Function | - 로지스틱 함수라고도 하며, X 값의 변화에 따라 0에서 1까지의 값을 출력  - 입력 값이 -5보다 작거나 보다 클 경우, 그래디언트 값이 지나치게 작아지고, 여난이 무거워 학습이 느려짐 |  |
| Tanh | - 하이퍼볼릭탄젠트는 시그모이드 함수의 크기와 위치를 조절한 함수  - 범위 : [-1, 1]  - 그래프의 모양은 시그모이드 함수와는 달리 0을 기준으로 대칭이므로 시그모이드를 사용할 때보다 학습 수렴 속도가 빠름 |  |
| Rectified  Linear  Unit  (ReLU) | - x가 양수면 그래디언트가 1로 일정하므로 그래디언트가 죽는 현상을 피할 수 있음  - 미분이 편해서 계산의 복잡성을 낮출 수 있음  - 실제로 시그모이드나 하이퍼볼릭탄젠트 함수 대비 학습 수렴 속도가 6배 빠름 |  |

**Chapter07. Convolutional Neural Network**

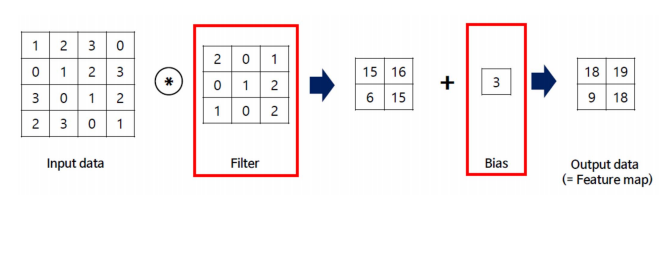
1. Deep Learning? Deep Neural Network
   1. Deep Learning 발전 과정
      1. Deep Neural Network 침체기 (1986~2006)
      2. 침체기로부터의 탈출
      3. Deep Belief Network (DBN, Hinton et al.,2007)
      4. ReLU(Rectified Linear Unit, Nair&Hinton 2010; Glorot et al.,2011)
      5. Drop-out (Dahl et al., 2013)
2. CNN 개념
   1. 개념
      1. 합성곱(Convolution) 연산을 사용하는 ANN(Artificial Neural Network)의 한 종류
      2. Convolution을 사용하면 3차원 데이터의 공간적 정보를 유지하여 다음 Layer로 전달
      3. 핵심 아이디어는 feature selection이 performance에 가장 크게 영향을 미치므로 가장 좋은 feature map을 뽑아주는 convolution filter를 learning하는 모델을 만드는 것
3. CNN 기본 구조 개념
   1. CNN 기본 구조 개념
      1. 전통적인 MLP에 여러 개의 Convolutional layer 붙인 구조
      2. Convolutional layer를 통해 입력 받은 이미지에 대한 Feature 추출
      3. 추출된 Feature를 기반으로 MLP를 이용해 분류
   2. 일반적인 CNN 구조



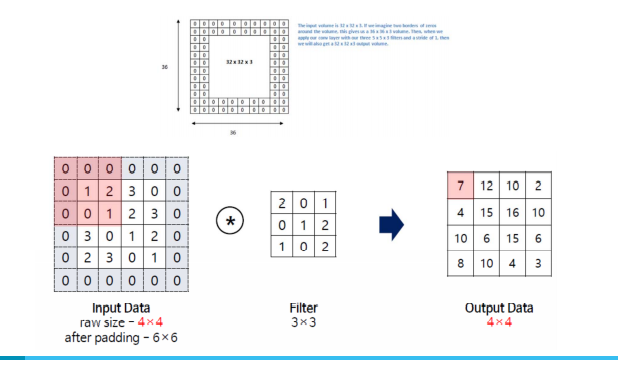
1. Convolution layer
   1. Convolution layer
      1. 다양한 위치에서 동일한 특징 추출(탐색)
      2. 특징을 추출하는 Filter와 Filter 값을 비선형 값으로 변환해 주는 Activation Function으로 구성
   2. 합성곱(Convolution)
      1. 특정(높이, 너비)를 갖은 필터(Filter, Kernel)를 일정간격(Stride)으로 이동해가며 입력데이터에 적용
      2. 이미지 전체를 동일한 필터로 Shift 시키면서 합성곱을 계산함



* 1. 합성곱(Convolution) 연산에서의 매개변수(Parameter)
     1. 필터를 적용한 후, 모든 원소에 편향(bias)이 더해짐
     2. 필터(가중치), 편향이 학습을 시킬 매개변수가 됨

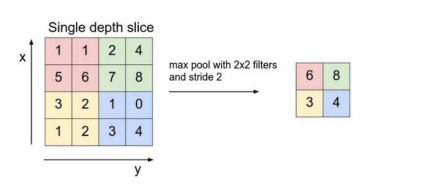


* 1. Convolution layer-Filter
     1. Filter는 그 특징이 Data에 있는지 없는지 검출해 주는 함수 개념
  2. 패딩(Padding)
     1. 합성곱 연산을 수행하기 전에 입력데이터 주변을 특정값 (주로 0–Zero padding)을 채우는 단계로, 입력데이터와 출력데이터의 크기를 맞추기 위해 사용

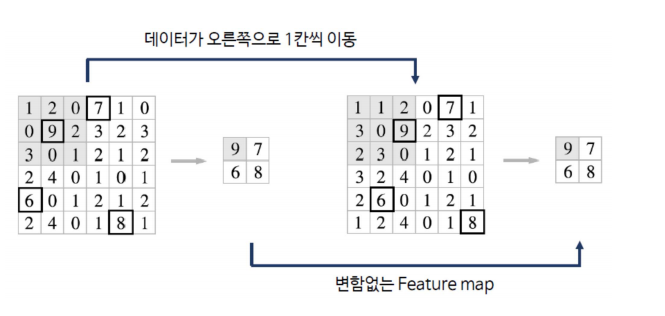


* 1. Sparse weights
     1. Full connection 대비 훨씬 적은 표현형을 learning하게 되는 단점이 있지만, 모델 complexity가 낮아진다는 장점이 큼
     2. 픽셀값이 많고 feature의 공간 상관성이 높은 이미지 데이터에서는 유용함
  2. Shared(tied) weights
     1. 하나의 필터와 연결되는 모든 뉴런은 동일한 weight로 학습됨(bias는 제외)
     2. 필터의 모든 뉴런이 다른 위치에 있는 동일한 feature를 detect할 수 있음
     3. 학습시켜야 할 parameter 수를 줄여서 Learning 속도가 빠름

1. Pooling
   1. 풀링 과정
      1. Kernel 해상도를 줄여줌 -> 모델의 파라미터 개수를 줄임
      2. 최대/평균 풀링을 주로 사용
   2. 풀링(sampling or pooling)
      1. Convolutional layer를 거쳐 추출된 특징들은 풀링 과정을 거침
      2. 추출된 Activation map을 인위적으로 줄이는 기법(max pooling, average pooling 등)
      3. 풀링은 대상영역에서 최댓값(Max Pooling)이나 평균(Average Pooling)만을 취하는 명확한 처리이므로 학습해야 할 매개변수가 없음
      4. Max pooling



* 1. Pooling Layer의 특징 : 입력의 변화에 영향을 적게 받음
     1. 입력데이터가 조금 변해도 풀링의 결과는 잘 변하지 않음



1. Full connected layer
   1. Fully connected layer – Softmax
      1. Softmax 또한 sigmoid나 ReLu와 같은 activation function
      2. Softmax 는 여러 개의 분류를 가질 수 있는 함수 > 확률을 표현해 주는 개념