# 딥러닝

## 개념

* 일반 신경망 모델 : 원시 데이터에서 ‘직접 특징’을 추출해서 만든 특징벡터를 입력으로 사용
* 딥러닝 신경망 : 특징 추출과 학습을 함께 수행, ‘효과적인 특징’을 학습을 통해 추출 가능

## 3가지 문제

### 기울기 소멸 문제

* + 은닉층이 많은 다층 퍼셉트론에서, 출력층에서 아래층으로 갈 수 록 전달되는 오차가 크게 줄어 학습이 되지 않는 현상
  + 완화 : 시그모이드나 쌍곡 탄젠트 대신 ReLU 함수 사용

### 가중치 초기화 문제

* + 경망 성능에 큰 영향을 주는 요소로 보통 초기값으로 0에 가까운 무작위 값 사용
  + 균등분포, 제이비어 초기화, 허 초기화 등 사용
  + 개선 : 입력 데이터를 제한적 볼츠만 머신을 학습시킨 결과의 가중치 사용
  + 개선 : 인접 층간 가중치를 직교 행렬로 초기화

### 과적합 문제

* + 모델이 학습 데이터에 지나치게 맞추어진 상태
  + 학습되지 않은 데이터에 대한 성능 저하
  + 완화 : 오차 함수를 오차 항과 모델 복잡도 항으로 정의하는 '규제화' 기법
  + 완화 : 일정 확률로 노드들을 무작위로 선택하여, 선택된 노드 앞뒤로 연결된 가중치 연결선이 없는 걸로 간주하고 학습하는 '드롭아웃 '기법
  + 완화 : 전체 학습 데이터를 일정크기로 나눈 '미니 배치' 단위로 학습

# 컨볼루션 신경망

## 개념

* 동물의 계층적 특징 추출과 시간 인식 체계를 참조하여 만들어진 딥러닝 신경망 모델
* 영상 분류, 문자 인식 등 인식 문제에 높은 성능

## 처리과정

* 전반부 : 컨볼루션 연산을 수행하여 특징 추출
  + 컨볼루션 연산을 하는 Conv층
  + ReLU연산을 하는 ReLU층
  + 풀링 연산 Pool(선택)
* 후반부 : 특징을 이용하여 분류
  + 전방향으로 전체 연결된 FC층 반복
  + 분류의 경우 마지막 층에 소프트맥스를 하는 SM연산 추가

## 컨볼루션

* 컨볼루션 : 일정 영역의 값들에 대해 가중치를 적용하여 하나의 값을 만드는 연산
* 스트라이드 : 커널을 다음 컨볼루션 연산을 위해 이동시키는 칸 수
* 패딩 : 컨볼루션 결과의 크기를 조정하기 위해 입력 배열 둘레를 확장하고 0으로 채우는 연산

## 특징지도

* 컴볼루션 필터의 적용 결과로 만들어지는 2차원 행렬
* 특징지도의 원소 값 : 컨볼루션 필터에 표현된 특징을 대응되는 위치에 포함하고 있는 정도
* k개의 컨볼루션 필터를 적용하면 k의 2차원 특징지도 생성

## 풀링

* 일정 크기의 블럭을 통합하여 하나의 대표값으로 대체하는 연산
* 특징 지도 크기 감소로 메모리 크기와 계산량 감수
* 최대값 풀링 : 지정된 블록 내의 원소들 중에서 최대값을 대표값으로 선택
* 평균값 풀링 : 블록내의 원소들의 평균값을 대푯값으로 사용
* 확률적 풀링 : 블록 내의 각 원소가 원소값의 크기에 비례하는 선택적 확률을 갖도록 하고, 이 확률에 따라 원소 하나를 선택