

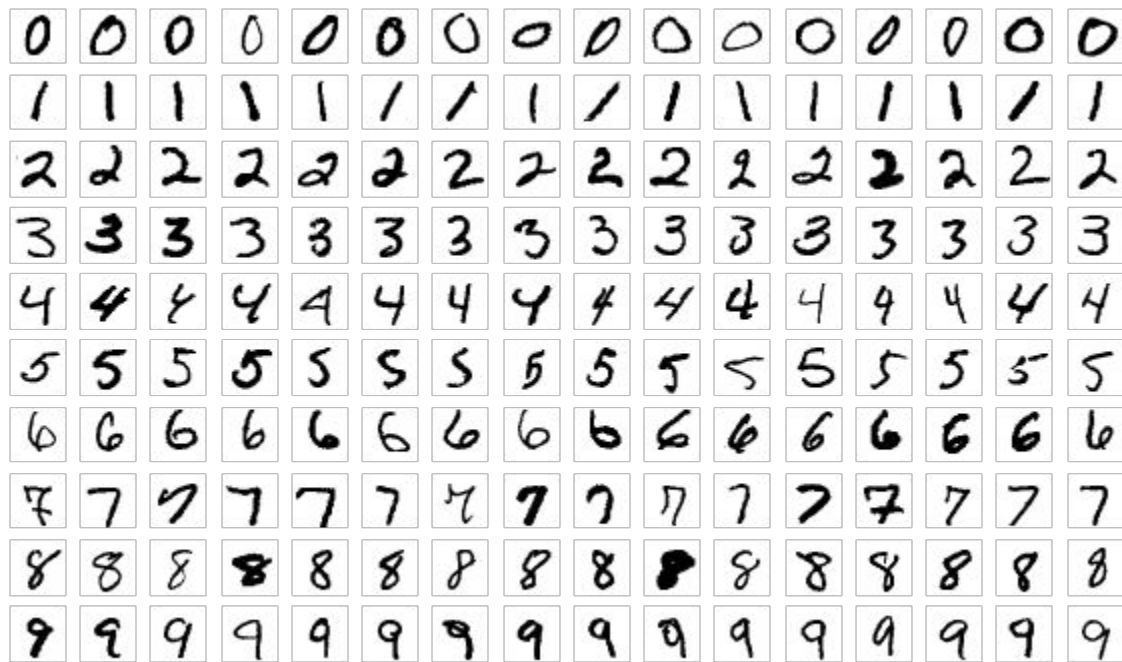
이미지 인식의 꽃, CNN

김용현, 강인영, 노지민, 유정수, 정유찬

목차

1. MLP로 이미지 분류하기(MNIST)
2. MLP의 한계 및 딥러닝의 도입
3. CNN이란?
4. 쉬는 Time~
5. 실습 문제

MLP로 MNIST 이미지 분류하기



MLP의 한계

■ 앞선 딥러닝 모델 →

하나의 은닉층을 가진 **MLP(다층 퍼셉트론)**

■ MLP의 장점이자 단점

=> '학습데이터'에서 완벽한 모델 생성

▶ **과적합!!**

■ Gradient Vanishing 문제

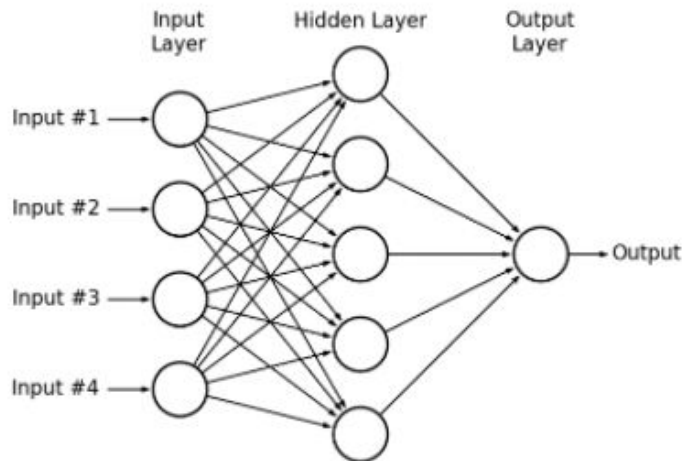
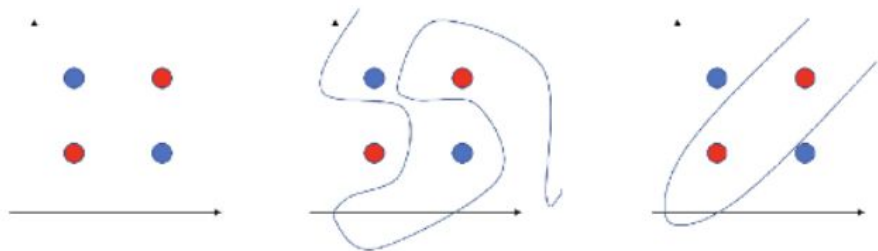


Fig. 1

Structure Multilayer Perceptron Neural Network

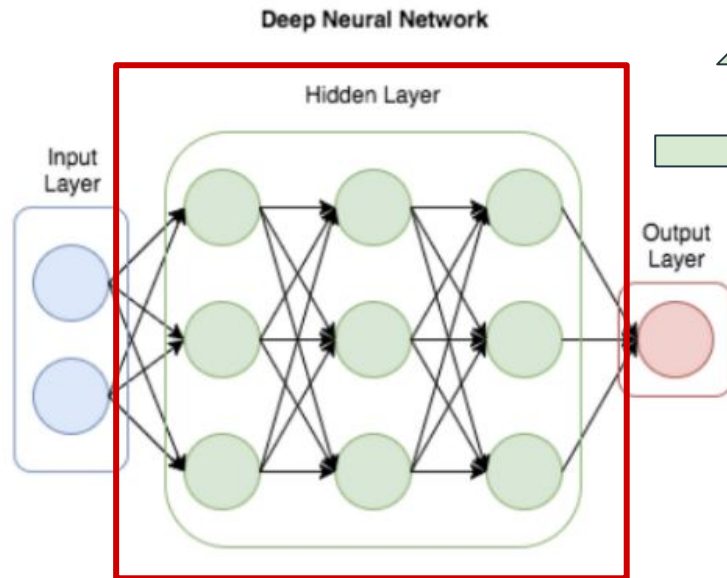
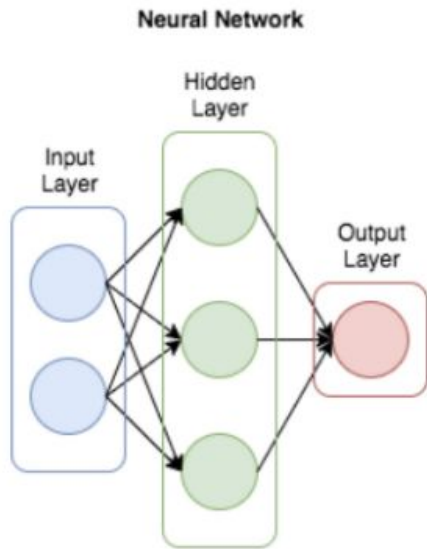


MLP ?

사전학습, RELU, GPU

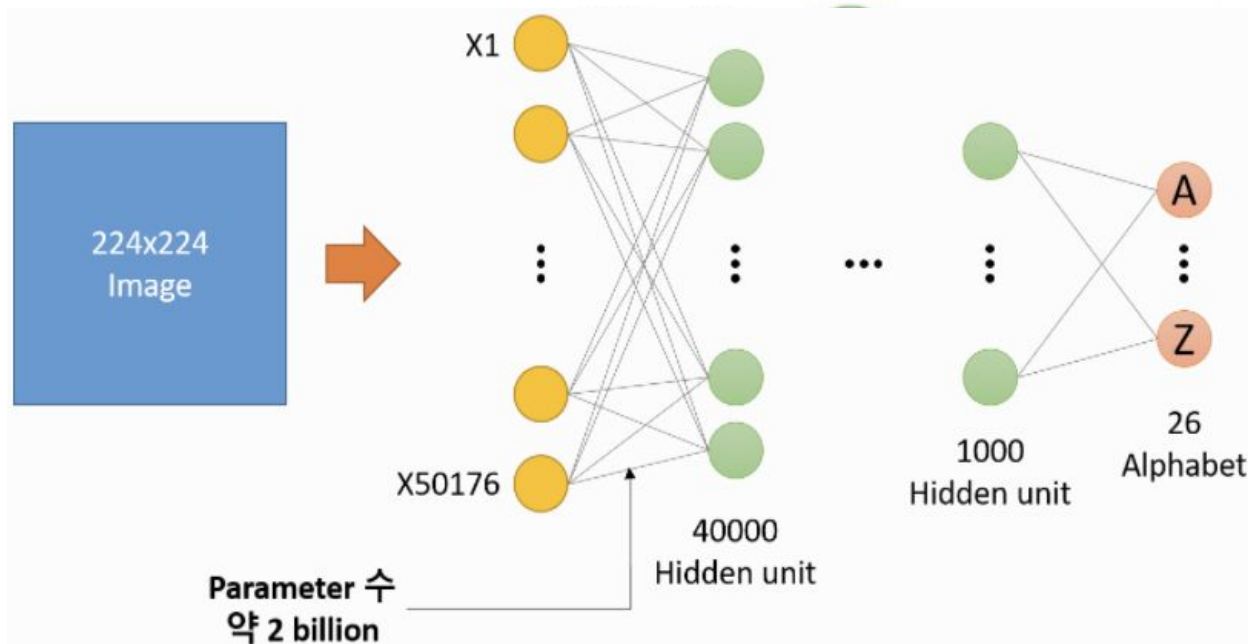
DNN !

더 깊은 모델

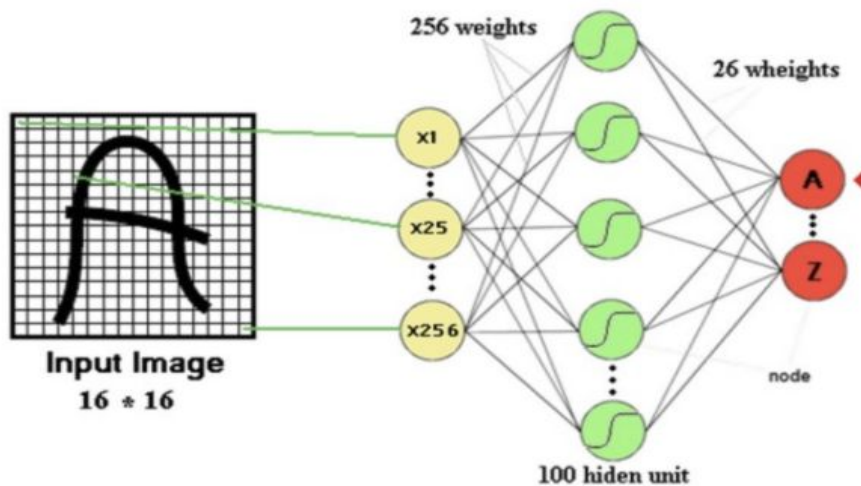


DNN : 2개 이상의 Hidden Layer를 가진 다층 신경망(Deep Neural Network)

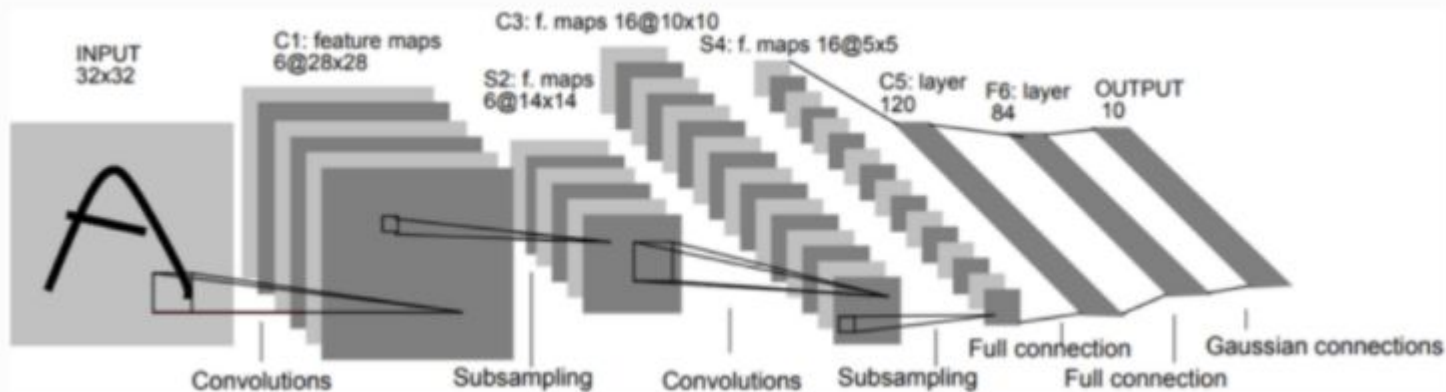
DNN의 한계 ▶ 데이터량이 많은 이미지와 영상



✓ 데이터 손실(평탄화) ✓ 매우 높은 시간 복잡도 ✓ Parameter 수 ↑



DNN



CNN

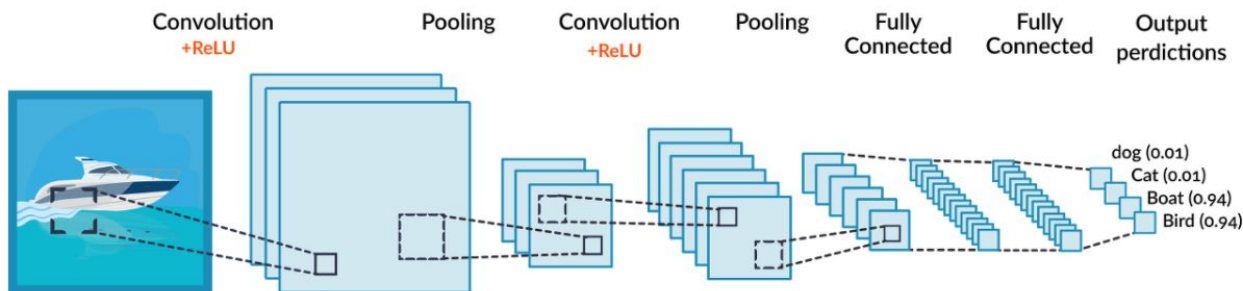


CNN이란?

CNN (Convolutional Neural Network) - 시각적 이미지를 분석하는데 사용되는 깊은 인공신경망의 한 종류

- CNN은 **변환 불변성 특성**에 기초, 이미지 및 비디오 인식, 추천 시스템, 이미지 분류, 의료 이미지 분석에 응용됨
- CNN은 다른 이미지 분류 알고리즘에 비해 상대적으로 **전처리를 거의 사용하지 않음.**

-> 네트워크가 기존 알고리즘에서 수작업으로 제작된 여러 필터 역할을 스스로 학습한다는 것을 의미



CNN의 원리

입력된 이미지에서 특징을 추출하기 위해 커널(kernel)을 도입함

커널? -> 공업수학 라플라스 변환에서 계산을 편하게 하기 위해 e^{-st} 를 커널로 사용함!

$$F(s) = \int_0^{\infty} k(s, t) f(t) dt$$

$$F(s) = L(f) = \int_0^{\infty} e^{-st} f(t) dt.$$

CNN의 원리

컨볼루션 (합성곱) - 하나의 함수와 또 다른 함수를 반전 이동한 값을 곱한 다음, 구간에 대해 적분하여 새로운 함수를 구하는 수학 연산자

공업수학 시간에서는 곱 형태의 라플라스 변환을 쉽게 하기 위해 컨볼루션 개념을 배웠음!!

$$h(t) = (f * g)(t) = \int_0^t f(\tau)g(t - \tau)d\tau.$$

-> 시간이 지나면서 타우 영역에 있는 하나의 함수가
같은 영역에 있는 **또 다른 함수** 위를 지나가게 됨!!

합성곱을 라플라스 변환에 적용 $L((f * g)(t)) = L(f)L(g)$

CNN의 원리

1 _{x1}	1 _{x0}	1 _{x1}	0	0
0 _{x0}	1 _{x1}	1 _{x0}	1	0
0 _{x1}	0 _{x0}	1 _{x1}	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved
Feature









$$h(t) = (f * g)(t) = \int_0^t f(\tau)g(t - \tau)d\tau.$$

-> 시간이 지나면서 타우 영역에 있는 하나의 함수가
같은 영역에 있는 **또 다른 함수** 위를 지나가게 됨!!

CNN의 원리

컨볼루션은 주로 filter연산에서 사용되며, 영상으로부터 특정 feature를 추출하고 싶을 때 사용함!

-> 어떤 커널을 쓰느냐에 따라 이미지 데이터가 다양하게 나올 (필터가 적용된 것처럼 나올!)

Details [edit]		
Depending on the element values, a kernel can cause a wide range of effects.		
Operation	Kernel	Image result
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur 3 × 3 (approximation)	$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	
Gaussian blur 5 × 5 (approximation)	$\frac{1}{256} \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}$	

제로 패딩(ZERO-Padding)

입력값: $d_1 \times d_2$
필터: $k_1 \times k_2$
결과값: $(d_1 - k_1 + 1) \times (d_2 - k_2 + 1)$

5x5 행렬 입력, 3x3 필터
output : $(5-3+1) \times (5-3+1) = 3 \times 3$

제로 패딩 후

5x5 행렬 → 7x7 행렬 사이즈 변경

3x3 필터

output : $(7-3+1) \times (7-3+1)$
= 5x5

1	2	3	4	5
2	1	0	1	2
3	0	1	1	0
1	4	1	1	2
2	1	1	0	0

패딩 전



0	0	0	0	0	0	0
0	1	2	3	4	5	0
0	2	1	0	1	2	0
0	3	0	1	1	0	0
0	1	4	1	1	2	0
0	2	1	1	0	0	0
0	0	0	0	0	0	0

패딩 후

풀링(Pooling)

풀링 (pooling) - 크고 복잡한 데이터를 축소하기 위해 풀링 과정을 거침

풀링은 입력이 작게 이동해도 근사적으로 불변!

-> 커널내부의 각 인접 픽셀들 몇 개만 대상으로 하기 때문!

-> 데이터 차원 감소로 인한 계산효율성 향상, 메모리 요구량의 감소

종류: max pooling, average pooling

Activation Map

12	20	30	0
8	12	2	0
34	70	37	7
112	100	22	12

Max Pooling

20	30
112	37

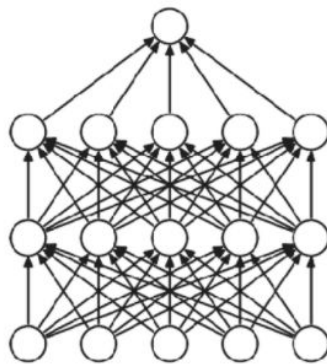
Average Pooling

13	8
79	18

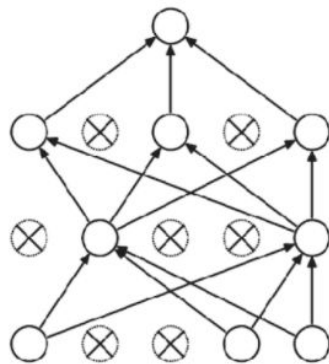
드롭아웃(Drop Out)

드롭아웃 (drop out) - 은닉층에 배치된 노드 중 일부를 임의로 **꺼주는 것**

- > 학습 과정에서 많은 학습량이 많아지게 되는 것을 막는다!
- > **과적합**을 효과적으로 피할 수 있음!!



(a) 일반 신경망



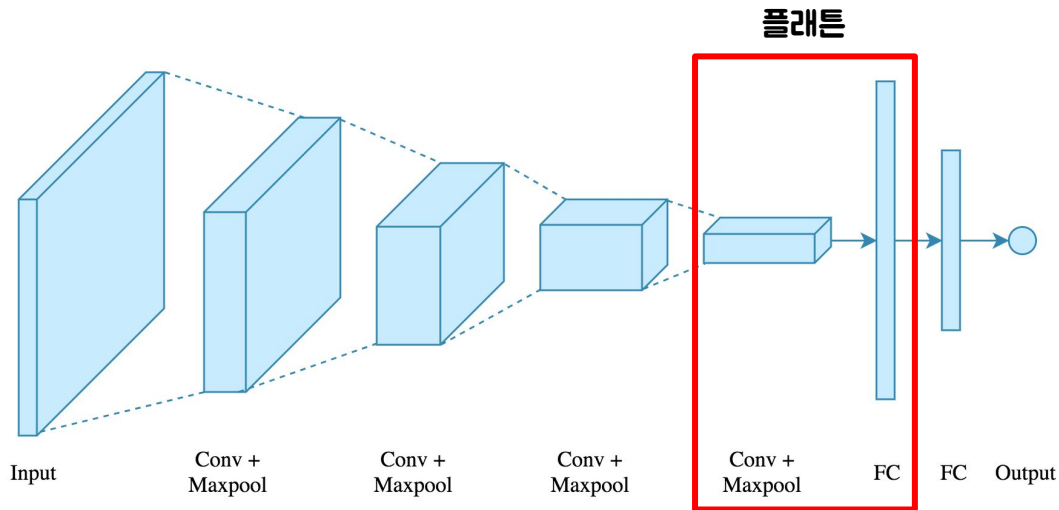
(b) 드롭아웃을 적용한 신경망

플래튼(Flatten)

Convolution과정과 Pooling과정이 끝난
데이터들을 한줄로 만드는 과정

How?

-> CNN과정을 거친 데이터는 사진 자료
자체가 아닌 그 사진에서 추출한 **특이점들**로
이 점들을 확인해 소프트맥스로 분류하는 것!



뭘봐?쉬는거 참봐?

쉬는 Time~~



예제 코드

<https://colab.research.google.com/drive/1cK7wkbCSsTfcepBNzCt4WQ3SSlyTHOY6?usp=sharing>



실습 : 옷 종류 분석

<https://www.kaggle.com/mirhyun0508/2022-smarcle-ai-fashion-mnist-cnn>

